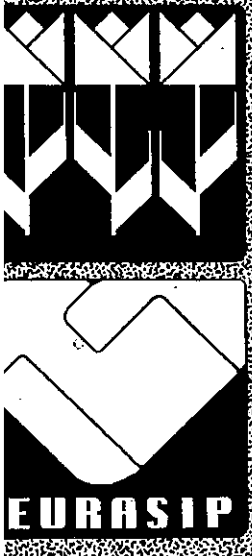


EUSIPCO-86

I.T. YOUNG
J. BIEMOND
R.P.W. DUIN
J.J. GERBRANDS
editors

Participants Edition
Part 2

THE HAGUE, THE NETHERLANDS
SEPTEMBER 2-5, 1986

North-Holland

Young, I.T.;
Biemond, J.; Duin, R.P.W.;
Gerbrands, J.J. (eds.):

3rd European Signal Processing Conference. The Hague, Sept.2-5,1986, Vol.2.

Amsterdam, New York, Oxford: North Holland Publishing Company, 1986.

EUSIPCO-86

EUSIPCO-86
SIGNAL PROCESSING III: THEORIES AND APPLICATIONS

Third European Signal Processing Conference

The Hague, The Netherlands
September 2-5, 1986

Edited by

I. T. YOUNG
and
R. P. W. DUIN

Pattern Recognition Group
Department of Applied Physics
Delft University of Technology
Delft, The Netherlands

J. BIEMOND
and
J. J. GERBRANDS

Laboratory for Information Theory
Department of Electrical Engineering
Delft University of Technology
Delft, The Netherlands

PARTICIPANTS EDITION
PART 2

N·H
P·C
1986

NORTH-HOLLAND — AMSTERDAM • NEW YORK • OXFORD • TOKYO

Post-conference edition (ISBN: 0 444 70085 4) will be published by:

ELSEVIER SCIENCE PUBLISHERS B.V.
P.O. Box 1991
1000 BZ Amsterdam
The Netherlands

Sole distributors for the U.S.A. and Canada:

ELSEVIER SCIENCE PUBLISHING COMPANY, INC.
52 Vanderbilt Avenue
New York, N.Y. 10017
U.S.A.

PRINTED IN THE NETHERLANDS

FOREWORD

The European Signal Processing Conference 1986, **EUSIPCO-86**, was the third in a sequence of international conferences promoted and organized by **EURASIP**, the European Association for Signal Processing. This book (in two volumes) represents the Proceedings of that conference. The conference was held from 2 to 5 September 1986 at the Netherlands Congress Centre in The Hague.

The conference provided a forum for the discussion of all aspects of signal processing theory and practice. Through the presentation of papers, posters, tutorials, technical exhibitions, and informal discussions, participants had the opportunity to exchange ideas with colleagues and stay abreast of current developments.

There were 42 separate sessions of the conference organized in seven parallel programs. The Scientific Program Committee reviewed over 450 submitted abstracts to select the 350 papers that were presented at **EUSIPCO-86**. Each submitted abstract was reviewed by at least two reviewers from two independent institutions. In addition nine tutorials were offered by experts in the field on topics ranging from "Compiling Silicon" to "Adaptive Detection." A special highlight of the meeting was the plenary talk by Dr. H. Peek on "The Compact Audio Disc - Digital Signal Processing in the Home."

Both the program sessions and these Proceedings were organized along the following lines:

Volume I:
- One-Dimensional Signal Theory
- One-Dimensional Signal Processing
- One-Dimensional Filtering
- Spectral Analysis
- Speech and Sound
- Audio and Speech Processing
- Speech Coding
- Speech Analysis and Recognition
- Applications

Volume II:
- Two-Dimensional Signal Processing
- Image Processing
- Digital Video
- Image Analysis
- Detection and Estimation
- Communications
- Radar
- Geophysics
- Chips
- Implementations
- Biomedical Applications

The diversity of these topics as well as the extraordinary tempo at which the work has progressed since the preceding **EUSIPCO** conferences in 1980 (Lausanne) and 1983 (Erlangen) attest to the vitality of this field.

We would like to thank all the participants in the conference and, in particular, those whose contributions made this Proceedings possible. In addition we would like to thank the Delft University of Technology for its support and the use of its facilities in the organization of **EUSIPCO-86**. Our thanks are also due to the co-sponsoring institutions for their financial support and to North-Holland Publishing Company for their cooperation, advice, and flexibility.

The organization of the conference could not have been carried out without the outstanding cooperation of the Scientific Program Committee, the Session Chairmen, the Tutorial Speakers, and the Local Organizing Committee. In particular we would like to single out Ms. José de Bruin for her heroic contribution to **EUSIPCO-86**.

Delft, The Netherlands
Summer, 1986.

Jan Biemond
Robert P.W. Duin
Jan J. Gerbrands
Ian T. Young

EUSIPCO-86

Conference Chairman

I. T. Young

Program Committee

E. Backer
M. Bellanger
A.J. Berkhout
R. Boite
D. Bosman
C. Braccini
H.J. Butterweck
T.A.C.M. Claassen
P. Dewilde
T.S. Durrani
S. Eiho
M. Ekstrom
B. Escudie
A. Fettweis
E. Gelsema
B. Gold
G. Granlund
C. Geuguen
O.E. Hermann
M. Kunt
M.A. Lagunas
W.F.G. Mecklenbräuker
R.M. Mersereau
H. Meyr
R. de Mori
M. Nagao
Y. Neuvo
H. Niemann
B. Picinbono
F. Rocca
H. Schüssler
J. Tribolet

Local Committee

J. Biemond
J. de Bruin
R.P.W. Duin
J. J. Gerbrands
A.C. de Ridder
J. Ridder
L. F. van der Wal

Organizers:

EURASIP
Delft University of Technology

CONTENTS

A. ONE-DIMENSIONAL SIGNAL THEORY

D. SPECTRAL ANALYSIS

E. SPEECH AND SOUND

F. AUDIO AND SPEECH PROCESSING

G. SPEECH CODING

H. SPEECH ANALYSIS AND RECOGNITION

I. APPLICATIONS

J. TWO-DIMENSIONAL SIGNAL PROCESSING

K. IMAGE PROCESSING

L. DIGITAL VIDEO

M. IMAGE ANALYSIS

N. DECTECTION AND ESTIMATION

O. COMMUNICATIONS

Q. GEOPHYSICS

R. CHIPS

S. IMPLEMENTATIONS

T. BIOMEDICAL APPLICATIONS

** not included in this volume.*

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

STABILIZED RECONSTRUCTION IN SIGNAL PROCESSING. A NEW PROPOSAL

M.J. Casanove[1], S. Roques[2] and A. Lannes[1]

[1]Laboratoire d'Optique Electronique 29, rue J. Marvig F-31055 Toulouse Cedex

[2]Observatoire du Pic du Midi et de Toulouse 14, Av. E. Belin F-31400 Toulouse

1. Theoretical Framework

One of the main problems encountered in Signal Restoration and Image Processing is the choice of the regularization principle ensuring the stability of the reconstruction procedures. Basically, we have to find a certain compromise between resolution and robustness. Moreover, before setting the reconstruction process in motion, it must be possible to have an idea of its relative stability. In this context, we have devised and developed a new deconvolution method, which finally shows what can and cannot be done in this field, whatever the particular technique implemented on the computer.

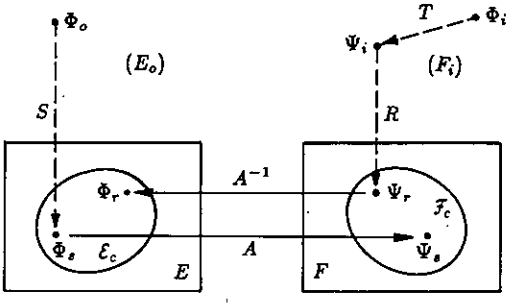


Figure 1

The diagram presented on Fig. 1 specifies the guidelines of our approach. The object space E_o and the image space F_i are Hilbert spaces (cf. [1] and appendix 1 in [2]); $\Phi_o(x)$ denotes the original object function defined by the observer, and $\Phi_i(x)$ its experimental image. This image is an approximation to the convolution product of Φ_o with some point-spread function. The accuracy of this approximation, which depends on the size of systematic and noise-type errors, is characterized by some pointwise error term σ_i:

$$|\hat{\Phi}_i(u) - h(u)\hat{\Phi}_o(u)| \leq \sigma_i(u) \quad (u \in \mathbf{R}^p).$$

Here $\hat{\Phi}_i(u) \stackrel{\text{def}}{=} U\Phi_i$ denotes the Fourier transform of Φ_i, and h a given transfer function.

In general, in view of the indeterminacies that may occur, it is preferable to give up trying to determine Φ_o at its highest level of resolution. It is then natural to define the object function to be reconstructed, Φ_s, as a smoothed version of Φ_o: $\Phi_s = S\Phi_o$; Φ_s lies in some Hilbert subspace E of E_o, which is referred to as the object-reconstruction space. In this paper, E is the Hilbert space of square-integrable complex-valued functions ϕ with support in some bounded region $V \subset \mathbf{R}^p$. The smoothing operator S corresponds to a continuous transformation in general irreversible.

The operation T, which is also irreversible, corresponds to the preliminary manipulations leading to the definition of the input image $\Psi_i \stackrel{\text{def}}{=} T\Phi_i$. This function can be regarded as a blurred representation of $\Psi_s \stackrel{\text{def}}{=} A\Phi_s$, where A is a continuous mapping from E into F_i. In this approach, the operators T and A are defined by taking account of the transfer function h and the Signal-to-Noise Ratio: $\mathrm{SNR}(u) \stackrel{\text{def}}{=} |\hat{\Phi}_i(u)|/\sigma_i(u)$.

By definition, $F \stackrel{\text{def}}{=} AE$ denotes the range of A. The reconstructed image Ψ_r is obtained from Ψ_i via the reconstruction operator R: $\Psi_r \stackrel{\text{def}}{=} R\Psi_i$. Provided that A is a one-to-one mapping from E onto F, the reconstructed object Φ_r can be defined by the relation $\Phi_r \stackrel{\text{def}}{=} A^{-1}\Psi_r$. Thus, for example (cf. [3]), when R is the orthogonal projection of F_i onto the closure of F, and when Ψ_r lies in F, Φ_r is the least-squares solution Φ of the equation $A\phi = \Psi_i$, i.e., the solution of the normal equation:

$$A^*A\Phi = A^*\Phi_i. \tag{1}$$

For stable reconstruction, A must have a continuous inverse. If this is not the case, the problem is ill posed, and a small image-reconstruction error $\Psi_r - \Psi_s$

may lead to a very large object-reconstruction error $\Phi_r - \Phi_s$.

It may often happen that Φ_s is confined to a given subset $\mathcal{E}_c$ of E, for example, when the nonnegativity constraint is to be applied; Ψ_r is then chosen in the subset $\mathcal{F}_c \overset{\text{def}}{=} A\mathcal{E}_c$.

2. Resolution and robustness

The guiding idea is that Band-Limited Interpolation is possible to some extent, whereas Band-Limited Extrapolation is strictly forbidden [4, 5]. The most natural way of regularizing the deconvolution problem is therefore to proceed in such a way that the high-frequency components of the object function to be reconstructed Φ_s are approximately known, and therefore *a priori* practically equal to 0. To this end, we define Φ_s such that $\widehat{\Phi}_s$ is small, in the mean square sense, outside a certain region H_r regularizing the "effective support" of the transfer function h. Clearly, Φ_s corresponds to the notion of "diffraction-limited image" w.r.t. the "synthetic aperture" H_r. For simplicity we shall assume in the following that H_r is centrosymmetric. We are thus led to set $\widehat{\Phi}_s(u) = s(u)\widehat{\Phi}_o(u)$, where s is a centrosymmetric function satisfying the following properties:

(i) *The energy of s is concentrated in H_r.* To this end, given χ of the order of 98%, we impose the following condition: the fraction of energy of s in H_r must be equal to χ^2 (cf. appendix 2 in [2]).

(ii) *The support D_r of the corresponding point-spread function $\hat{s}$ is as small as possible w.r.t. the choices of H_r and χ.* The idea is of course to have the best possible resolution.

(iii) *The normalization condition* $s(0) = 1$.

The support of $\hat{s}$, D_r, proves to be a certain p–dimensional ellipsoid: the *resolution ellipsoid*. The size $\delta\tau$ of D_r defines the resolution limit of the reconstruction process; $\delta\tau$ varies as $1/\Delta\nu$, where $\Delta\nu$ is the diameter of H_r (cf. Fig. 2). In most cases, $\hat{s}$ is nonnegative on D_r; thus, if $\Phi_o \geq 0$, so too is Φ_s.

When the signal-to-noise ratio is less than some threshold value α_t of the order of unity, the corresponding image information $\widehat{\Phi}_i$ can be discarded *a priori*. The function

$$\widehat{\Phi}_t \overset{\text{def}}{=} s_t\widehat{\Phi}_i \text{ where } s_t = \begin{cases} s/h & \text{if SNR} > \alpha_t \\ 0 & \text{otherwise,} \end{cases}$$

then gives the principal features of $\widehat{\Phi}_s$. Depending on the values of SNR, $\widehat{\Phi}_t(u)$ is more or less reliable. In this context, it is natural to define the reconstructed object Φ_r as the function that minimizes the functional on E,

$$q(\phi) \overset{\text{def}}{=} \left\| g(\widehat{\Phi}_t - \widehat{\phi}) \right\|^2, \tag{2a}$$

where g is a certain weight function with least upper bound unity. This function must be a nondecreasing function of SNR; g is of course equal to 0 in the parts of H_r where SNR$< \alpha_t$. According to our regularization principle, g is set equal to 1 outside H_r.

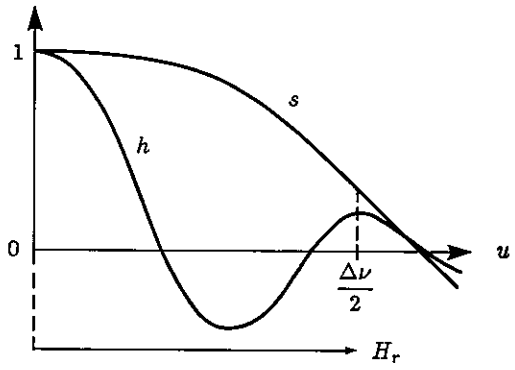


Figure 2

Equation (2a) can also be written in the form

$$q(\phi) = \|\Psi_i - A\phi\|^2, \tag{2b}$$

where $\Psi_i \overset{\text{def}}{=} T\Phi_i = U^*gs_tU\Phi_i$ and $A\phi \overset{\text{def}}{=} U^*gU\phi$. Thus Φ_r is the solution of the normal equation (1); A^*A proves then to be of the form $I - B$ where B is the "finite-transfer operator" on E:

$$B\phi \overset{\text{def}}{=} vU^*wU\phi \text{ with } w \overset{\text{def}}{=} 1 - g^2. \tag{3}$$

Here v is the characteristic function of V; clearly w can be regarded as the "weighted-characteristic function" of the region $W \subset H_r$ on which a certain interpolation is to be performed. *The operator B is positive definite, compact and its largest eigenvalue λ is strictly less than 1.*

As a result the inverse of A^*A is continuous, and the *number of degrees of freedom* of the reconstruction process N is the number of the eigenvalues of B

essentially different from 0. Moreover, λ, the largest eigenvalue of B, and therefore $\mu = 1 - \lambda$ (the smallest eigenvalue of A^*A) can be estimated analytically, and this by simple inspection of v and w (cf. [6]). If μ is close to 0, the problem is in practice ill conditioned, and Φ_s must be redefined at a lower level of resolution by modifying the choice of H_r.

The main parameter (involved in the corresponding analysis) characterizes the amount of weighted interpolation to be performed:

$$\eta \stackrel{\text{def}}{=} \left[\int v(x)\,dx\right]^{1/p} \left[\int w(u)\,du\right]^{1/p}. \qquad (4)$$

In practice, this parameter must be less than or of the order of 3 (unless the error level is very low).

Before setting the reconstruction process in motion, it is preferable to have estimated the relative object-reconstruction error $\theta \stackrel{\text{def}}{=} \|\Phi_r - \Phi_s\| / \|\Phi_r\|$. An upper bound of θ is given by

$$\Theta \stackrel{\text{def}}{=} \frac{1}{\sqrt{\mu}} \frac{\|\Psi_r - \Psi_s\|}{\|\Phi_r\|},$$

where μ can be estimated from the behaviour of λ for small η (cf. Fig. 5 below). Clearly

$$\|\Psi_r - \Psi_s\| \leq \|\Psi_i - \Psi_s\|.$$

Let us now denote by k_r the characteristic function of the region K_r located outside H_r on which SNR$< \alpha_t$, and by $\sigma_o(u)$ an upper bound of $|\widehat{\Phi}_o(u)|$. It is then easy to show that

$$\|\Psi_i - \Psi_s\|^2 \leq \|g s_t \sigma_i\|^2 + \|k_r s \sigma_o\|^2.$$

By suitably choosing H_r and the weight function g, the size of the input error can be made acceptably small.

When Θ has a reasonable value, the reconstruction process is set in motion. The method of conjugate gradients is then very well suited to solving the normal equation: *the convergence is superlinear*, and in fact, the solution is attained in a number of iterates less than or of the order of N (the number of degrees of freedom). Moreover, if N is not too large, an appropriate implementation of this technique allows us to localize the eigenvalues of A^*A. At the end of the reconstruction process, the relative object-reconstruction error can then be estimated in a more precise way.

3. Numerical implementation

To illustrate our analysis in a concrete manner, we now present the simulated study of an academic one-dimensional deconvolution problem. The phantom object Φ_o and its experimental image Φ_i are represented on Fig. 3; Φ_i is a blurred version of the convolution of Φ_o by a rectangle function: Poisson noise depending on the pointwise values of this convolution was added.

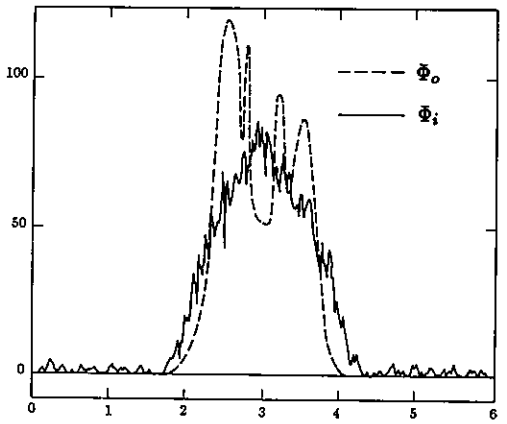


Figure 3

Here, the size of the support of the rectangle function corresponds to the unit of length. In order to show the interest of the notion of degrees of freedom, this simulation was carried out on an oversampled set of 256 equally-spaced points. The general appearence of Φ_i shows that the high-resolution structures of the original object Φ_o are completely washed out.

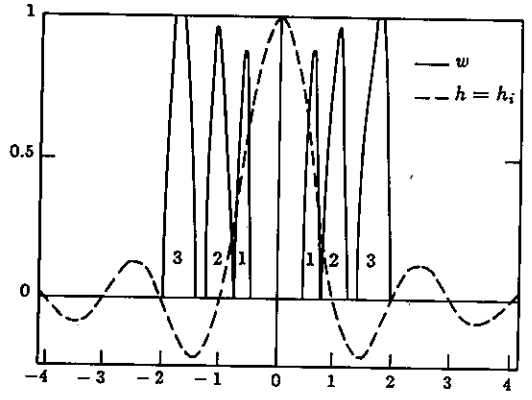


Figure 4

The corresponding transfer function h is represented on Fig. 4, as well as the weighted-characteristic

function w relative to the choice $\Delta\nu = 4$. We recall that $\Delta\nu$ denotes the diameter of the synthetic aperture H_r; here $\Delta\nu$ has to be compared to the size of the effective support of h, say 3 or 3.5. Clearly, w is a nonincreasing function of the pointwise values of SNR. The second and third peaks are due to the zeros of the transfer function h inside H_r, whereas the first reflects a situation that may occur accidentally. The value of the interpolation parameter η is of the order of $\eta_e = 2.7$.

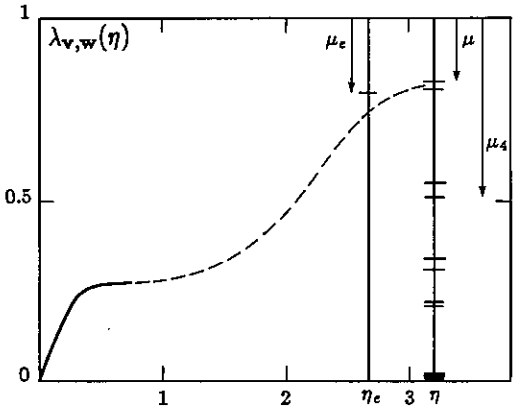


Figure 5

As outlined on Fig. 5 (where is shown the general behaviour of λ as a function of η), it is then possible to have an idea of μ; here the estimated value of μ, μ_e was found to be 0.2. The upper bound Θ was then about 15%. As the situation seemed to be relatively stable, the reconstruction process was set in motion. In order to show the part played by the nonnegativity constraint in the definition of the actual support of Φ_s, V was deliberately chosen too large. By using $v\Phi_t$ as starting point, the conjugate-gradient iteration converges very rapidly (about 3 iterations). At the cost of a few extra iterations (here 8), the same algorithm yields the localization of the eigenvalues (cf. Fig. 5). Note that the number of degrees of freedom is equal to 8.

The results of the reconstruction are presented on Fig 6 where Φ and Φ_c denote the least-squares solutions without and with nonnegativity constraint respectively. These functions have of course to be compared with Φ_s, the definition of which corresponds to a gain in resolution slightly greater than 2 ($\delta\tau \simeq 0.41$). Although, in practice, Φ_s is of course unknown, this method allows us to have an idea of the relative object-reconstruction error θ; this by referring to the singular-value decomposition of A, and by taking account of the localization of the eigenvalues. In the special case under consideration, we found an estimate of about 10%, whereas the actual error θ was equal to 8%. Finally, on comparing the general appearence of the eigenfunctions of A^*A (corresponding to the smallest eigenvalues) with that of the reconstructed object, it was possible to conclude that the high-resolution structure of Φ was not an artefact.

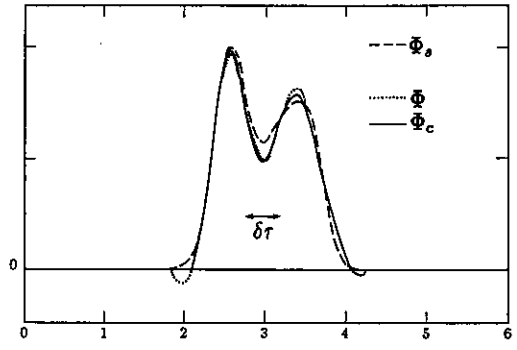


Figure 6

References

[1] Reed M. and Simon B., Methods of modern mathematical Physics, I: Functional analysis, Academic Press New York and London (1972).

[2] Lannes A. et Pérez J.-Ph., Optique de Fourier en Microscopie Electronique, Masson, Paris (1983).

[3] Sanz J.L.C. and Huang T.S., Unified Hilbert space approach to iterative least-squares linear signal restoration, J. Opt. Soc. Am., 73 (1983), pp. 1455-1465.

[4] Lannes A., Abstract holography, J. Math. Anal. Appl., 74 (1980), pp. 530–559.

[5] Youla D.C., Generalized image restoration by the method of alternating orthogonal projections, IEEE Trans. Circuits and Systems, CAS–25 (1978), pp. 694–702.

[6] A. Lannes, M.J. Casanove and S. Roques, Stabilized reconstruction in signal and image processing, series of papers submitted to Optica Acta.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A SIGNAL PROCESSING APPROACH TO THE DIGITAL SIMULATION OF MULTI-DIMENSIONAL CONTINUOUS SYSTEMS

Rudolf Rabenstein

Lehrstuhl für Nachrichtentechnik
University Erlangen-Nürnberg
Cauerst. 7, D-8520 Erlangen, West-Germany

The digital simulation of 1D continuous systems described by ordinary differential equations with constant coefficients is a well studied subject. Considerably less is known about the corresponding multi-dimensional problem, except for the methods for the numerical solution of partial differential equations (PDE) provided by numerical analysis. This paper approaches the problem from a signal processing point of view and extends the 1D simulation techniques to multidimensional systems.

1. INTRODUCTION

A multi-dimensional system S^c described by a PDE with constant coefficients may be characterized by its Green's function $g^c(\mathbf{x},t)$. $\mathbf{x}$ denotes the vector of space variables and t the time variable. The Green's function can be obtained from the PDE by suitable functional transformations. The relation between the input signal $v^c(\mathbf{x},t)$ and the output signal $y^c(\mathbf{x},t)$ is given by a multi-dimensional convolution with respect to space and time variables:

$$y^c(\mathbf{x},t) = v^c(\mathbf{x},t) \underset{\mathbf{x}}{*} \underset{t}{*} g^c(\mathbf{x},t). \qquad (1)$$

The digital simulation of S^c is accomplished by a discrete system S^d characterized by a discrete Green's function $g^d(\mathbf{n},k)$. The corresponding relation to (1) is given by

$$y^d(\mathbf{n},k) = v^d(\mathbf{n},k) \underset{\mathbf{n}}{*} \underset{k}{*} g^d(\mathbf{n},k) \qquad (2)$$

where $\underset{\mathbf{n}}{*} \underset{k}{*}$ denotes the multi-dimensional convolution with respect to the vector of discrete space variables $\mathbf{n}$ and time variable k. The usual requirement for S^d is that for some space increment h and time increment T:

$$y^d(\mathbf{n},k) \approx y^c(\mathbf{n}h,kT) \text{ for } v^d(\mathbf{n},k) = v^c(\mathbf{n}h,kT) \qquad (3)$$

Two problems arise at this point:

For 1D systems, there exists a variety of methods to determine S^d from S^c, such as impulse and step invariant transformations, bilinear transformation and others. However, none of these methods is readily extendable to higher dimensions, since initial values for the time variable and boundary values for the space variables have to be considered.

The second problem is the calculation of the convolution in (2), due to the large number of arithmetic operations even for moderate dimensions.

How these difficulties can be overcome will be shown briefly for a heat flow problem in two space and one time dimensions. A more detailed description of the approach is given in /8/. At first, the continuous problem will be treated by choosing appropriate functional transformations for the time and space coordinates. Initial and boundary values are thus automatically included into the description. Along the same lines, a simulating discrete system is derived under the reasonable assumption that the continuous input function is bandlimited in the spatial coordinates. Finally the reduction of the numerical expense will be addressed.

2. CONTINUOUS SYSTEM

In this section, we show for the linear heat flow equation how the Green's function can be derived elegantly by means of functional transformations.

Let $v(\mathbf{x},t)$ and $y(\mathbf{x},t)$ be functions of the space variables $\mathbf{x} = (x_1, x_2)'$ (' denotes transposition) and the time variable t where $\mathbf{x} \in G$ and $t > 0$. The set G consists of the square $G = \{\mathbf{x} | 0 < x_i < A,\ i = 1,2\}$. Further $y_0(\mathbf{x})$ is a constant function with respect to time. v and y are the input and output functions respectively of a linear system S^c described by the PDE (4)

$$y_t(\mathbf{x},t) - \Delta y(\mathbf{x},t) = v(\mathbf{x},t) \quad \mathbf{x} \in G \qquad (4a)$$
$$y(\mathbf{x},0) = y_0(\mathbf{x}) \quad \mathbf{x} \in G \qquad (4b)$$
$$y(\mathbf{x},t) = 0 \quad \mathbf{x} \in \partial G \qquad (4c)$$

The subscript t denotes derivation with respect to time and Δ denotes the Laplace operator

$$\Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} \qquad (5)$$

(4b) specifies the initial value $y_0(\mathbf{x})$ for t = 0 and (4c) describes the boundary conditions.

Homogeneous Dirichlet boundary conditions are assumed for simplicity.
The PDE (4) can now be transformed into an algebraic equation through application of suitable functional transformations. For a proper treatment of initial and boundary conditions, the domains of integration of the transformations have to match the domains on which the independent variables of the PDE are defined. To achieve this, we will apply a method dating back to /1/. The problem is considered on the extended domain $G' = \{\mathbf{x} | -A < x_i < A, \ i=1,2\}$. $v(\mathbf{x},t)$ and $y(\mathbf{x},t)$ are defined on G' by odd continuation

$$v(-x_1,x_2,t) = v(x_1,-x_2,t) = -v(x_1,x_2,t), \quad (6)$$

$y(x_1,x_2,t)$ alike. Then we introduce the 2D limited Fourier transformation with respect to the space variables

$$Y(\boldsymbol{\nu},t) = \mathcal{F}_A\{y(\mathbf{x},t)\} = \int_{G'} y(\mathbf{x},t)\exp(-j\boldsymbol{\nu}'\mathbf{x}\pi/A)d\mathbf{x} \quad (7)$$

with the vector of spatial frequencies $\boldsymbol{\nu} = (\nu_1,\nu_2)'$. Application of $\mathcal{F}_A$ to (4) turns the PDE into an ordinary differential equation in t

$$Y_t(\boldsymbol{\nu},t) + \boldsymbol{\nu}'\boldsymbol{\nu}(\pi/A)^2 Y(\boldsymbol{\nu},t) = V(\boldsymbol{\nu},t) \quad (8a)$$
$$Y(\boldsymbol{\nu},0) = Y_0(\boldsymbol{\nu}), \quad (8b)$$

that can be easily solved by Laplace transformation with respect to time. The solution is

$$Y(\boldsymbol{\nu},t) = Y_0(\boldsymbol{\nu})G(\boldsymbol{\nu},t) + V(\boldsymbol{\nu},t) \underset{t}{*} G(\boldsymbol{\nu},t), \quad (9)$$

where $G(\boldsymbol{\nu},t) = \exp(-\boldsymbol{\nu}'\boldsymbol{\nu}(\pi/A)^2 t)\,\delta_{-1}(t)$. (10)

Inverse 2D limited Fourier transformation

$$y(\mathbf{x},t) = (2A)^{-2} \sum_{\boldsymbol{\nu}} Y(\boldsymbol{\nu},t)\exp(j\boldsymbol{\nu}'\mathbf{x}\pi/A) \quad (11)$$

yields

$$y(\mathbf{x},t) = y_0(\mathbf{x}) \underset{t}{*} g(\mathbf{x},t) + v(\mathbf{x},t) \underset{\mathbf{x}t}{**} g(\mathbf{x},t) \quad (12)$$

with $$g(\mathbf{x},t) = \mathcal{F}_A^{-1}\{G(\boldsymbol{\nu},t)\}, \quad (13)$$

where we recognize the Green's function of the continuous problem as $g^c(\mathbf{x},t) = g(\mathbf{x},t)$.

Equ. (12) is the solution of the PDE (4). It consists of two terms: the response to the initial value $y_0(\mathbf{x})$ and the response to the input signal $v(\mathbf{x},t)$. The first term will be omitted in the sequel, since it is equal to the response to an input signal of the form $v(\mathbf{x},t) = y_0(\mathbf{x})\delta_0(t)$ and can therefore be treated as a special case of the second term. Only here we will very briefly touch upon the case of non homogeneous boundary conditions (4c). The boundary values are specified by four functions of one space coordinate only, defined on the four sides of the square G. Each of those functions would be included in (8a) through the transformation (7), just like the initial value $y_0(\mathbf{x})$ is included in (9) through Laplace transformation. The total solution would then contain four more terms in addition to those given in (12).

Of special interest is the response to a bandlimited input signal, that is $V(\boldsymbol{\nu},t) = 0$ except for $|\nu_i| < N$, i=1,2 with some integer N. In this case $G(\boldsymbol{\nu},t)$ in (9) may be replaced by its bandlimited version

$$G_b(\boldsymbol{\nu},t) = \begin{cases} G(\boldsymbol{\nu},t) & |\nu_i| < N \ i=1,2 \\ 0 & \text{else} \end{cases}, \quad (14)$$

such that the Green's function is given by

$$g_b(\mathbf{x},t) = \sum_{\boldsymbol{\nu}}^{N} \exp(-\boldsymbol{\nu}'\boldsymbol{\nu}(\pi/A)^2 t + j\boldsymbol{\nu}'\mathbf{x}\pi/A) \quad (15)$$

where $\sum_{\boldsymbol{\nu}}^{N}$ denotes summation over all $|\nu_i| < N$ i=1,2.

3. DISCRETE SYSTEM

It was shown in the last section, how the output signal of the continuous system can be calculated from the input signal by continuous convolution with the Green's function. In this section we will show, how samples of the output signal can be calculated from samples of the input signal by discrete convolution with a sequence that will be called the Green's function of the discrete problem. At first, we look at the spatial discretisation and then at the discretisation with respect to time.

3.1 Discretisation of the Spatial Variables

If input signal $v(\mathbf{x},t)$ and output signal $y(\mathbf{x},t)$ are bandlimited functions with respect to the spatial variables, then we can represent these functions without loss of information on a spatial grid with the grid points $\mathbf{x} = \mathbf{n}h$.
$\mathbf{n} = (n_1,n_2)'$ is a vector with the integer elements n_1 and n_2. The step size h depends on the highest frequency component to be represented on the grid. In the sequel we will assume, that $V(\boldsymbol{\nu},t)$ (and consequently $Y(\boldsymbol{\nu},t)$) is nonzero only for $|\nu_i| < N$ i=1,2. The step size h necessary to represent all those spatial frequencies is then given by $h = A/N$.

We start with specializing (9) for homogeneous initial and boundary conditions to the bandlimited input signal $v(\mathbf{x},t)$:

$$Y(\boldsymbol{\nu},t) = V(\boldsymbol{\nu},t) \underset{t}{*} G_b(\boldsymbol{\nu},t). \quad (16)$$

From (11) follows directly that the values of $y(\mathbf{x},t)$ at the grid points are given by the 2D inverse discrete Fourier transformation (2D-IDFT) of lenght 2N in both directions

$$y(\mathbf{n}h,t) = \text{2D-IDFT}_{2N}\{Y^*(\boldsymbol{\nu},t)\} \quad (17)$$

where

$$Y^*(\boldsymbol{\nu},t) = h^{-2} \sum_{\mathbf{i}} Y(\boldsymbol{\nu}-2N\mathbf{i},t) \quad (18)$$

is the periodic extension of $Y(\boldsymbol{\nu},t)$. Due to the assumed bandlimitation there is no aliasing and thus

$$Y^*(\boldsymbol{\nu},t) = h^{-2} Y(\boldsymbol{\nu},t) \quad |\nu_i| < N, \ i = 1,2. \quad (19)$$

The function $V^*(\boldsymbol{\nu},t)$ is defined in the same way.

Inserting into (16) and applying 2D-IDFT yields the relation

$$y(\mathbf{n}h,t) = v(\mathbf{n}h,t) \;\textcircled{2N}\; \underset{t}{*}\; g_N(\mathbf{n}h,t) \qquad (20)$$

with

$$g_N(\mathbf{n}h,t) = \text{2D-IDFT}_{2N}\{G_b(\mathbf{v},t)\} = h^2 g_b(\mathbf{n}h,t). \qquad (21)$$

$\textcircled{2N}$ denotes 2D discrete cyclic convolution with respect to the discrete space variable n with period 2N in either direction.
(20) describes how the exact samples of the output signal can be calculated from the sampled input signal by a discrete operation for the spatial variables.

3.2 Discretisation of the Time Variable

A different approach has to be taken for the discretisation of the time variable, since the the assumption of bandlimitation is not appropriate here. However, we can use the known simulation techniques for 1D systems instead (e.g. /2/). For instance, let $v(\mathbf{n}h,t) = v_0(\mathbf{n}h)\delta_0(t)$. The exact output signal at the equally spaced discrete times $t = kT$ (step size T) according to (20) is then given by

$$y(\mathbf{n}h,kT) = v_0(\mathbf{n}h) \;\textcircled{2N}\; g_N(\mathbf{n}h,kT). \qquad (22)$$

Note that for $v_0(\mathbf{n}h) = y_0(\mathbf{n}h)$ we have the response to the initial value (see (9)). Using $g_N(\mathbf{n}h,kT)$ to calculate the approximate response to arbitrary input signals sampled at $t = kT$ in time results in a simulation according to the well known impulse invariant transformation

$$y^d(\mathbf{n},k) = v^d(\mathbf{n},k) \;\textcircled{2N}\; \underset{k}{*}\; g_N(\mathbf{n}h,kT). \qquad (23)$$

$*_k$ denotes discrete convolution. Step and ramp invariant transformations can be defined in the same way. Also bilinear transformation is possible by starting from the Laplace transformation of (20). However, the impulse invariant transformation lends itself to an easy implementation. With the relation

$$G_b(\mathbf{v},(k+1)T) = G_b(\mathbf{v},kT)G_b(\mathbf{v},T) \qquad (24)$$

(see (10) and (14)), we can give the following recursion for (23)

$$y^d(\mathbf{n},k+1) = y^d(\mathbf{n},k) \;\textcircled{2N}\; g_N(\mathbf{n}h,T) + v^d(\mathbf{n},k+1). \qquad (25)$$

A matrix notation reveals that this recursion corresponds to a state space description. An implementation of the simulating discrete system S^d can be given as shown in figure 1 (see /3/). S^d is designed by determining $g_N(\mathbf{n}h,T)$ from (21).

Now the derivation of the discrete system S^d is complete. The discrete output signal $y^d(\mathbf{n},k)$ is calculated from samples of the input signal $v(\mathbf{n}h,kT)$ by convolution with the discrete Green's function of the discrete system $g^d(\mathbf{n},k) = g_N(\mathbf{n}h,kT)$. The simulation is exact for the spatial variables as long as the input signal is bandlimited, while it is an approximation for the time variable in the sense of the impulse invariant transformation.

However, exactness in space direction has its price. The sequence $g^d(\mathbf{n},1) = g_N(\mathbf{n}h,T)$ as required for an implementation according to fig. 1 will in general be nonzero for most of the grid points $\mathbf{n}h$, resulting in a large number of operations to perform the convolution $\textcircled{2N}$. This problem will be adressed in the next section.

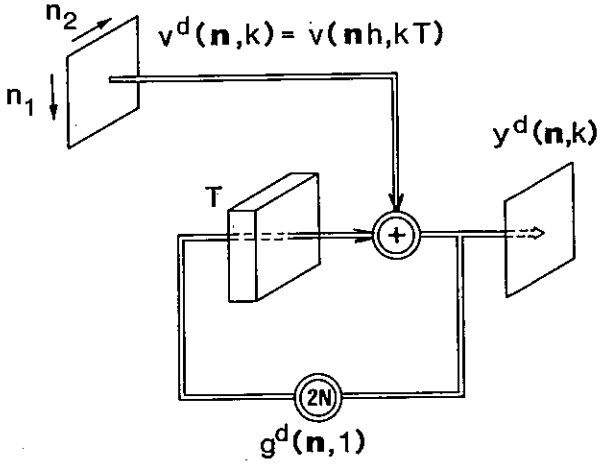


figure 1: Implementation of the simulating discrete system S^d

4. REDUCTION OF THE NUMERICAL EXPENSE

Two methods will be proposed to reduce the number of arithmetic operations required for the convolution with the discrete Green's function. They are based on the properties of the Green's function in the spatial and in the frequency domain respectively.

4.1 Approximation of the Discrete Green's Function

The discrete Green's function according to (21) will have values that are not equal to zero but very small for most of the spatial grid points, if T is not chosen too large. We may therefore approximate its spectrum $G_b(\mathbf{v},T)$ by

$$\tilde{G}_b(\mathbf{v},T) = \text{2D-DFT}\{\tilde{g}_N(\mathbf{n}h,T)\}, \qquad (26)$$

where $\tilde{g}_N(\mathbf{n}h,T)$ is nonzero only for $|n_i| \le m$ $i=1,2$ with $m \ll N$. The discrete Green's function $g^d(\mathbf{n},1)$ in fig. 1 is then replaced by the approximation $g^d(\mathbf{n},1) = \tilde{g}_N(\mathbf{n}h,T)$. We briefly discuss three methods for the choice of the $(2m+1)^2$ nonzero coefficients of $\tilde{g}^d(\mathbf{n},1)$:

The simplest way to obtain an approximate Green's function with a small spatial region of support is to truncate the exact discrete Green's function. This corresponds to an L_2-

approximation of $G_b(\nu,T)$. The main disadvantage is, that in general $\tilde{G}_b(0,T) \neq G_b(0,T)$. That means, that a steady state solution is not approximated well.

Another possibility that avoids the disadvantage mentioned above is to consider ν as a continuous variable and to determine $\tilde{g}_N(\mathbf{n}h,T)$ such that $\tilde{G}_b(\nu,T)$ and $G_b(\nu,T)$ and their first m derivatives with respect to ν match for zero frequency. This corresponds to the usual explicit discretisations used in the finite difference methods for the numerical solution of PDEs. The approximation is quite good for small spatial frequencies, but these discretisations are known to be unstable unless T is chosen very small.

The third way is a combination of both. $\tilde{G}_b(\nu,T)$ is determined to match $G_b(\nu,T)$ and possibly some of its derivatives at zero frequency. Remaining degrees of freedom are used for an L_2-approximation over all spatial frequencies. This method is an extension of the 1D approach given in /4/ to the 2D case. The stability constraints are greatly reduced as can be shown by the methods in /5/, allowing a tradeoff between time step size T and quality of approximation, i. e. between speed and accuracy.

4.2 Division into Subbands

Inspection of (10) shows that different frequency components of the input signal appear with quite different weights at the output. Low spatial frequencies have a gain of almost unity, while higher frequencies are increasingly damped. Depending on the actual frequency content of the input, the output signal may not always contain the full frequency range and consequently the fine grid spacing h may not always be required. To exploit this fact, the input signal is split up into r^2 subband signals with L^2 frequency components each (rL=N):

$$V_{\boldsymbol{\rho}}(\nu,t) = \begin{cases} V(\nu,t) & \rho_i L \leq \nu_i < (\rho_i+1)L \quad i=1,2 \\ 0 & \text{else,} \end{cases} \tag{27}$$

$$\text{such that } V(\nu,t) = \sum_{\boldsymbol{\rho}} V_{\boldsymbol{\rho}}(\nu,t), \quad \boldsymbol{\rho} = (\rho_1,\rho_2)' \quad 0 \leq \rho_i < r \quad i=1,2. \tag{28}$$

Thus we obtain r^2 partial problems

$$Y_{\boldsymbol{\rho}}(\nu,t) = V_{\boldsymbol{\rho}}(\nu,t) \overset{*}{t} G_{b\boldsymbol{\rho}}(\nu,t), \tag{29}$$

each with a different Green's function depending on the frequency components specified by . Each problem can now be treated as before, however, with an increased spatial step size rh corresponding to the decreased frequency range. Only those subbands have to be used, which actually contribute nonneglectable quantities to the output signal. Thus we can match the number of arithmetic operations required for the calculation of the output signal to its current frequency content. Of course, each partial Green's function defined by its spectrum $G_{b\boldsymbol{\rho}}(\nu,T)$ can be approximated by the methods described in section 4.1.

5. CONCLUSION

We have dealt with the question how to find a discrete system for the digital simulation of a continuous system given by a PDE. The heat flow equation was chosen as an example.

The result is a flexible method for the digital simulation of multi-dimensional continuous systems that can be tuned to the problem, the desired accuracy and the given computing facility by appropriate choice of the algorithm parameters, as time step size T, number of subbands r and region of support for $\tilde{g}_d(\mathbf{n},1)$. The required computations are formulated as signal processing operations, so that the use of existing signal processing hard- and software is facilitated.

The presented method has been described under several simplifying assumptions, which do not restrict its use for more general cases. At first it is not confined to the heat flow equation. A discrete system can be assigned to any continuous system given by a linear PDE in the same way as described in sections **2** and **3**. Also more complicated domains and sampling schemes can be treated using a generalized definition of the limited Fourier transformation and of the sampling process.

ACKNOWLEDGEMENT

The author is indebted to Prof. Schüßler for his advice in preparing this manuscript.

REFERENCES

/1/ G. Doetsch: Integration von Differentialgleichungen vermittels der endlichen Fouriertransformation, Math. Annalen, 112 (1936), pp. 52 - 68

/2/ H.W.Schüßler: A Signal Processing Approach to Simulation, FREQUENZ 35 (1981) 7, pp.174 - 184

/3/ H.W.Schüßler: A 1D Approach to 2D Signal Processing, in: P.Stucki (ed.), Advances in Digital Image Processing (Plenum Publishing Corp.,1980) pp. 33 - 59

/4/ H.W.Schüßler, P.Steffen: An Approach for Designing Systems with Prescribed Behaviour at Distinct Frequencies Regarding Additional Constraints, in: Proc. ICASSP 85

/5/ P.Steffen, R.Rabenstein: Stability of Image Sequence Processing Systems, AEÜ 37 (1983) 7/8, pp. 261 - 266

/6/ R.Rabenstein: A Signal Processing Approach to the Numerical Solution of Partial Differential Equations, in: NTG-Fachbericht 84, (VDE-Verlag GmbH, Berlin, 1983)

/7/ R.Rabenstein: A Signal Processing Approach to the Numerical Solution of Parabolic Differential Equations, GMD-Study, in print

/8/ R.Rabenstein: Discrete Simulation of Multi-Dimensional Continuous Systems, to be publ.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON THE DISCRETIZATION ERROR OF FOURIER DESCRIPTORS OF PLANAR CLOSED CURVES

F. van der Heyden

Twente University of Technology,
BSC, department of Electrical Engineering,
P.O. Box 217, 7500 AE Enschede, The Netherlands.

ABSTRACT: Fourier descriptors (FDs) have interesting properties in respect to recognition and measurements of boundary curves. Digitalization of a curve introduces uncertainties in the FDs. One source of uncertainty is due to nonuninform sampling, which is inherent in a 8-connected chain code representation of the curve. The effects of nonuniform sampling on the FDs is analysed and an upperbound of the corresponding error is given.

1. Introduction

The recognition of planar closed curves, and measurements on this curves regarding the geometry and shape arises in many applications such as machine vision, aircraft recognition, medical diagnosis, character reading, etc. Fourier descriptors (FDs) have been successfully used by many investigators [1,2]. Deviations of the measured region boundary from the real boundary of the object affects the uncertainty in the measured FDs. These deviations depend on the imaging conditions. However, when the conditions are optimized, one source of uncertainty still remains: the digitizing error. The digitalization of the curve possibly affects the FDs threefold:

- aliasing
- quantization error
- nonuniform sampling

In this paper the attention is focused on the error due to nonuniform sampling. This error can be reduced by preprocessing the curve [3], but this impairs the computational effectiveness. Therefore it is favourable to have a model of the variations of FDs due to the nonuniform sampling, so as to deliberately decide whether preprocessing is needed.

Amongst the various definitions of Fourier descriptors of closed contours the one used here is from Persoon and Fu [1]. It involves the Fourier series expansion of the function, mapping the arclength of the contour into its co-ordinates. It is defined as follows:

Let $z(l)=x(l)+jy(l)$ be the complex function ($j=\sqrt{-1}$) generated by tracing a planar closed curve 'C' along its length in a counterclockwise direction. 'l' is the running arclength (figure 1). This function is periodic with a period 'L', the perimeter of the contour. The complex Fourier series coefficients Z_k of the function $z(l)$ are defined to be the FDs of the curve 'C' and are given by:

$$Z_k = \frac{1}{L}\int_{l=0}^{L} z(l)\exp(\frac{-2\pi jkl}{L})dl \qquad (1)$$

$$z(l) = \sum_k Z_k \exp(\frac{2\pi jkl}{L}) \qquad (2)$$

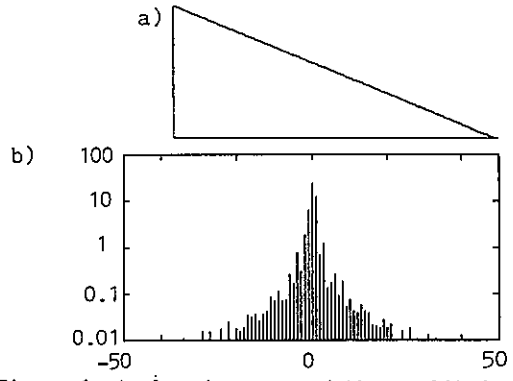


Figure 1. A closed curve and the amplitude spectrum of its FDs.

A property, which will be exploited in this paper, is that the FDs are at most of order $O(1/k^2)$, by which is meant that

$$\sum_k k^2 |Z_k|^2 \qquad (3a)$$

possesses a limit. In fact (3a) is proportional to the squared perimeter of the contour. On the other hand it can be proven that if the contour exhibits sharp edges the limit

$$\lim_{k\to\infty} k^2 Z_k \qquad (3b)$$

does not exist. These properties stem from the choice of the 'running arclength' as the independent variable describing the contour.

2. Effects of discretization of the contour

Computing the FDs of a continuous contour by digital means inevitably implies the use of some digitizing scheme. We take the sampling grid on the (x,y) plane to be a square grid, and assume a grid intersection quantization method, so that the resulting discrete contour is unbiased with respect to its area. The discrete contour is assumed to be represented by a 8-neighbourhood connected chain.

The effects of chain code representation on distances in the FD space may be better understood with the aid of figure 2, which shows two triangles. They are identical, except that figure 2a is a uniform sampled continuous contour, while figure 2b is one of its 8-n connected chain code representations.

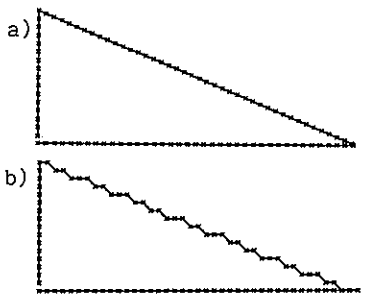


Figure 2. Uniform sampled triangle (a) and an 8-n chain code representation (b).

The basic shape difference between the triangles of figure 2 consists of an irregular 'ripple'-component present in the hypothenuse of figure 2b. Assume that we are comparing the normalised FDs, derived from these two triangles by the euclidean distance measure. Using Parsevals theorem, the distance between the normalised FDs can be decomposed into contributions from each pair of corresponding points in the space domain (Note the presumed one-to-one correspondence between points of the uniformly sampled contour and the points of the chain code).

The resulting distance can be split into two parts. One contribution is made up by the 'ripple' component. Each point of the uniformly sampled contour is rounded to a gridpoint, giving rise to random fluctuations proportional to the grid period. This type of error, known as quantization error, is inherent in the uncertainty introduced by discretization. It can be reduced to a arbitrary low level by choosing a finer grid period.

The chain code representation implies an increased density of points on the hypothenuse at the cost of a decreased density on the legs. This causes the points in the chain coded triangle to pull away from the points in the uniform sampled triangle. The 'average pull away distance' in this example is about 0.01 times the perimeter, so that this type of error dominates the quantization error if the perimeter is larger than about 20 times the grid period. Moreover the density error dominates in the middle frequency range of the FDs, which happens to be the range which describes the more detailed information of the contour.

3. Influence of sampling density error

Let W_k be the set of FDs associated with the nonuniform sampled contour. The aim in this section is to express W_k in terms of the set Z_k.

If the discrete contour is traced a systematical error occurs in measuring the running arclength l. Let p denote the measured running arclength. The tangent angle of the contour in l is $\phi(l)=\arg(z'(l))$. Then by given l an infinitesimal increment dl is overestimated proportional to a factor $c_z(l)$, which only depends on $\phi(l)$. In the case of an 8-n chain we have:

$$dp = c_z(l)dl \tag{4}$$

$$c_z(l) = \cos(\phi(l))+(\sqrt{2}-1)\sin(\phi(l)) \tag{5}$$

From (4):

$$p(l) = \int_{\xi=0}^{l} c_z(\xi)d\xi \tag{6}$$

so that the (average) measured perimeter will be $P=p(L)$. The inverse, $l(p)$ can be expressed as:

$$l(p) = \int_{\eta=0}^{p} c_w(\eta)d\eta \tag{7}$$

$$c_w(p) = \frac{1}{c_z(l(p))} \tag{8}$$

Let $s=l/L$ and $t=p/P$ be the normalized running arclength of resp. the true running arclength and the apparent running arclength. Furthermore define $v(s)=z(sL)$, so that $z(l(p))=v(s(t))$ and:

$$Z_k = \int_{s=0}^{1} v(s)\exp(-2\pi jks)ds \tag{9}$$

The measured FDs, then, can be expressed as:

$$W_k = \frac{1}{P}\int_{p=0}^{P} z(l(p))\exp(\frac{-2\pi jkp}{P})dp \tag{10}$$

$$= \int_{t=0}^{1} v(s(t))\exp(-2\pi jkt)dt$$

Furthermore from (9):

$$v(s(t)) = \sum_k Z_k\exp(2\pi ks(t)) \tag{11}$$

which finally yields:

$$W_n = \int_{t=0}^{1} \sum_k Z_k \exp(2\pi jks(t))\exp(-2\pi jnt)dt \qquad (12)$$

$$= \sum_k Z_k \int_{t=0}^{1} \exp(2\pi j(ks(t)-nt))dt$$

Expression (12) gives the dependency of W_k on Z_k. In order to arrive at an estimate of the upper bound of $|W_k-Z_k|$ define:

$$\Delta s(t) = t-s(t) = \sum_m \Delta_m \exp(2\pi jmt) \qquad (13)$$

in which Δ_m are the coefficients of the Fourier expansion of $\Delta s(t)$. Furthermore define $\tilde{\Delta}s(t)=\Delta s(t)-\Delta_0$ as the fluctuating part of $\Delta s(t)$. Then:

$$W_n = \sum_k Z_k \exp(2\pi jk\Delta_o) \int_{t=0}^{1} \exp(2\pi j(k-n)t) \qquad (14)$$

$$\exp(2\pi jk\tilde{\Delta}s(t))dt$$

$$= Z_n \exp(2\pi jn\Delta_o) + \sum_{k\neq n} Z_k \exp(2\pi jk\Delta_o) 2\pi jk\Delta_{n-k}$$

where it is presumed, that $|2\pi k\tilde{\Delta}s(t)|<<1$.

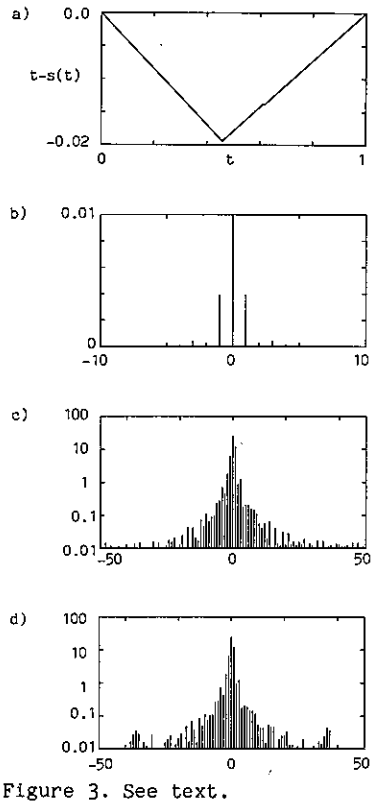


Figure 3. See text.

Figures 3a and 3b depict the functions $\Delta s(t)$ resp. Δ_m belonging to the triangle of figure 1a. Figures 3c and 3d show FDs predicted by (14) and FDs belonging to the 8-n discrete triangle of figure 2b. As can be seen the predicted FDs match the measured FDs in the lower and middle frequency range. The deviations in the higher frequency range are due to the quantization noise, which is not covered by expression (14).

4. Bounds on sampling density error

The function $\Delta s(t)$, introduced in (13), can be regarded as a frequency modulation function in (12). An upperbound of $\Delta s(t)$ and the corresponding Fourier series Δ_m can be found by writing (refer to (7) and (8)):

$$\Delta s(t) = t-s(t) = t-\frac{P}{L}\int_{\eta=0}^{t} c_w(\eta P)d\eta \qquad (15)$$

$$= \frac{P}{L}\int_{\eta=0}^{t} \Delta c_w(\eta P)d\eta$$

where $\Delta c_w(\eta P)$ is defined by:

$$\Delta c_w(\eta P) = \frac{L}{P}-c_w(\eta P) \qquad (16)$$

Using Parsevals theorem the following identity of the Fourier series Δ_m in terms of $\Delta c_w(\eta P)$ can be stated:

$$4\pi^2 \sum_m m^2|\Delta_m|^2 = \frac{P^2}{L^2}\int_{\eta=0}^{1} |\Delta_w(\eta P)|^2 d\eta \qquad (17)$$

The right hand term reaches its upper value if the contour is such that only maximum and minimum values of $c_w(\eta P)$ occurs. This is the case only when the tangent angle of the contour equals $n\pi/4+n/8\pi$ during one (noninterrupted) half of the period P and $n\pi/4$ during the other half. The maximum resp. minimum values of $c_w(\eta P)$, according (8) and (5), are 1 resp. 0.92, so that then: L=0.96P. Accordingly the minimum resp. maximum values of $\Delta c_w(\eta P)$ are: -0.04 resp. 0.04:

$$\sum_m m^2|\Delta_m|^2 \leq \frac{(1.04)^2(0.04)^2}{4\pi^2} \approx (0.007)^2 \qquad (18)$$

In the worst case situation, described above, the first harmonics Δ_1 and Δ_{-1} will dominate (note that Δ_m is of the order $O(1/m^2)$). This gives:

$$|\Delta_1| = |\Delta_{-1}| \approx \frac{1.04x0.04}{2\pi\sqrt{2}} \approx 0.005 \qquad (19)$$

It must be notified however that if other harmonics dominate, the sampling density error would be much smaller. For instance the discrete circle is 8-fold rotational symmetric so that the dominating harmonic of $\Delta s(t)$ is bounded by: $|\Delta_8| \leq 0.005/8=0.0006$ (experiments have shown that this bound is reached almost completely). Furthermore a reduction of the 'worst case' sampling density error also occurs, when the contour is rotated, or when the function $\Delta c_w(\eta P)$ is strongly nonperiodical.

The sampling density error might give rise to an erroneous starting point shift (14), given by Δ_0. This starting point is bounded by: $|\Delta_o|<0.01$, as can be seen from (15). However,

generally speaking one is not interested in the starting point of a contour. Therefore this error will kept from the next discussion.

5. A model of the sampling density error

It is of general importance to have a model, which covers the sampling density error, even in the worst case, without having a priori knowledge about the shape and orientation of the contour at hand. Absorbing the starting point shift error in Z_n, (14) can be rewritten as:

$$W_n - Z_n = \sum_{m \neq 0} Z_{n-m} 2\pi j(n-m)\Delta_m \qquad (20)$$

If it is assumed, that the first harmonics of $\Delta s(t)$ dominate (the worst case), (20) becomes:

$$W_n - Z_n \simeq Z_{n-1} 2\pi j(n-1)\Delta_{-1} + Z_{n+1} 2\pi j(n+1)\Delta_1 \qquad (21)$$

Because Z_n is of order $O(1/n^2)$ it is reasonable to propose a signal model according:

$$E\{Z_n\} = 0 \qquad (22a)$$

$$\mathrm{Var}\{\mathrm{Re}(Z_n)\} = \mathrm{Var}\{\mathrm{IM}(Z_n)\} = \frac{|Z_1|^2}{2n^4} \qquad (22b)$$

Then after some mathematical manipulations it follows:

$$\mathrm{Var}\{|W_n - Z_n|\} = \frac{|\Delta_1|^2 |Z_1|^2 8\pi^2}{n^2} \quad |n|>1 \qquad (23)$$

In general an upperbound of the sampling density error is given by (19) and (23):

$$\mathrm{Var}\{|W_n - Z_n|\} \simeq \frac{a^2 |Z_1|^2}{n^2} \quad a \leq 0.04 \qquad (24)$$

Figure 4, which shows the amplitude spectrum of the differences between the original FDs (figure 1b) and the predicted FDs (figure 3c), confirms this result.

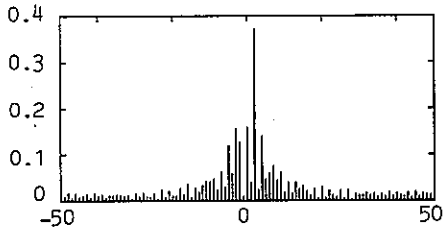


Figure 4. Sampling density error of figure 2b.

6. Verification

From (24) optimal weighting factors A_n can be derived, in order to minimize the distance between the nonuniform sampled contour and the original contour:

$$d(W,Z) = \sum_n |Z_n - A_n W_n|^2 \qquad (25)$$

Optimizing the weighting factors A_n by minimalization of $\mathrm{Var}\{d(W,Z)\}$, using (22) and (24), yields:

$$A_n = \frac{1}{1+a^2 n^2} \qquad (26)$$

Note, that A_n acts as a low pass filter with bandwidth $1/a$. Figure 5 shows the measured distance $d(W,Z)$ of the triangle example, in dependency of the cut-off frequency $1/a$ (disconnected line). In addition the measured distance is shown when using an rectangular low pass filter (connected line). It can be seen that filtering according (26) gives a minimum distance, which is slightly better than the rectangular filter. Moreover this minimum is reached when $a \simeq 0.04$. This corresponds well with the experiments of Wallace and Wintz [2], who applied a low pass filter to the contour. Their experiments suggest that a FIR filter with a nominal width of about 0.04 times the perimeter would be optimal.

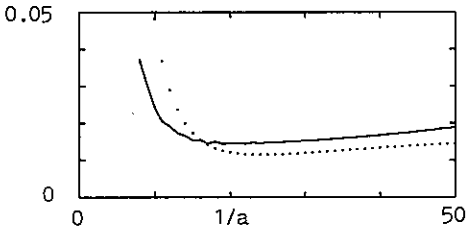


Figure 5. See text.

7. Conclusion

The chain code representation of a contour introduces some errors in the corresponding FDs. The contribution of nonuniform sampling to these errors has been examined and a statistical model, describing this contribution, is proposed.

References

[1] E. Persoon and K.S. Fu, Shape discrimination using Fourier descriptors, IEEE Tr. SMC-7 170-179 (1977).

[2] T.P. Wallace and P.A. Wintz, An efficient three-dimensional aircraft recognition algorithm using normalized Fourier descriptors, Comp. Graphics and Image Processing, 13, 99-126 (1980).

[3] B. Shahraray and D.J. Anderson, Uniform resampling of digitized contours, IEEE Tr PAMI-7 674-681 (1985).

SIGNAL PROCESSING/III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMAGE CODING BASED ON 2-D LINEAR PREDICTION AND 2-D MULTIPULSE EXCITATION

C. HORNE, K. JAINANDUNSING, and ED. F. DEPRETTERE

Delft University of Technology
Department of Electrical Engineering
Mekelweg 4, 2628 CD Delft, The Netherlands

ABSTRACT

In this paper it is shown that images can be effectively modeled as the output of a field-adaptive 2-D linear prediction spectral shaping filter input with a certain low-capacity 2-D excitation field. From the proposed class of models, we present a prototype coder that is essentially a 2-D version of the so-called multipulse excited linear prediction model that has been shown to allow effective medium bit rate encoding of speech signals. As in the 1-D case, we model the non-stationary spectral characteristics of the image to be encoded by a space-varying field-adaptive linear prediction filter. The prediction residual that contains the information regarding the image edges is then approximated by a low capacity excitation field that contains a limited number of non-uniformly spaced pulses with varying amplitudes. Both the pulse positions and the pulse amplitudes are determined using an analysis-by-synthesis procedure that minimizes the accumulated squared error between the original and the decoded image fields. Several commonly used 8 bit PCM quantized gray-level test images have been encoded using the 2-D multipulse excited linear prediction model, and preliminary results indicate that the proposed approach yields surprisingly high fidelity decoded images.

1. INTRODUCTION

In speech coding problems, a common and powerful approach is to use a block-adaptive inverse linear prediction (LP) filter to model the short-time spectral envelope of the speech signal. The prediction residual, containing information about the fine structure of the signal spectrum, is then quantized and used as filter excitation to retrieve the speech signal. There are many ways in which the residual can be quantized, but it is difficult to design a quantizer that is optimal with respect to some meaningful objective or subjective performance criterion. In one class of coders, the residual is approximated by a (constrained) low capacity excitation signal that is obtained by an analysis-by-synthesis procedure that minimizes a certain distance between the original and the decoded signal. Examples of such coders can be found in the literature. See e.g. [1, 2, 3]. The aim of this paper is to report on some preliminary results in 2-D linear prediction modeling of images and 2-D approximation of the prediction residual by an analysis-by-synthesis optimization procedure. A conceptual block diagram is depicted in figure 1.
In this figure, the residual r(n,m) is the output of a p-th order field-adaptive linear prediction filter

$$A(z,w) = \sum_{k=0}^{p} \sum_{l=0}^{p} a_{kl} z^{-k} w^{-l} \quad , \quad a_{00} = 1 \qquad (1)$$

input with the image field $s(n,m)$. The difference between $s(n,m)$ and the output $\hat{s}(n,m)$ of the inverse filter $\frac{1}{A(z,w)}$ input with an appropriate approximation $v(n,m)$ of $r(n,m)$ is fed through an error-weighting filter $W(z,w)$. The resulting weighted difference $e(n,m)$ is squared and accumulated, and is used as a small-field measure for judging the effectiveness of the presumed approximation $v(n,m)$ of $r(n,m)$.
One way to approximate $r(n,m)$ is by selecting $v(n,m)$ from a code book of representative excitation fields, obtained from a sufficient large collection of test images. The vector v_i to be selected from the code book is the one that minimizes the distortion measure $\|s-\hat{s}(v_i)\|^2$. Here we shall not pursue this approach, since vector quantization is well established in the image coding community. Notice, however, that the usage of a weighted distortion measure as suggested above is not so common and may enhance the

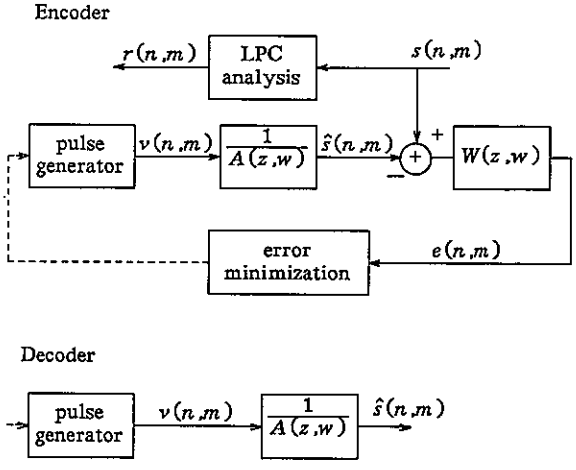


Figure 1. Block diagram for the analysis-by-synthesis procedure.

* This research has been partially sponsored by the Dutch National Applied Science Foundation under Contract STW DEL 44.0643.

quality of the decoded images, as has been amply demonstrated in 1-D speech coding experiments [4]. Thus, instead of elaborating on VQ approximation techniques we shall concentrate on an approximation method that is known in the 1-D literature as the *multipulse excitation linear prediction coding [1]*. This prototype analysis-by-synthesis coding technique has been shown to be very successful in high quality low bit rate coding of speech signals. In short, the multipulse approximation $v(n,m)$ of $r(n,m)$ consists of a limited number of non-uniformly spaced pulses on each optimization field.

In section 2 we briefly review the idea of auto regressive image modeling and the usage of 2-D linear prediction techniques to extract model parameters. In section 3 we describe procedures to determine the multipulse excitation field. Finally, in section 4, we give some experimental results revealing that the proposed image coding technique is worth continuing research efforts in this direction.

2. 2-D LINEAR PREDICTION MODELING OF IMAGES

The technique of linear prediction has been shown to be very successful in many 1-D signal modeling and spectral estimation problems [5]. Linear prediction is widely used to extract fundamental features in recognition and low bit rate encoding of speech signals. Several approaches to extend the 1-D linear prediction modeling of signals to the domain of 2-D random fields have been presented in the recent literature. See e.g. [6, 7, 8, 9, 10]. 2-D linear prediction is not a straight forward extension of 1-D linear prediction, at least in practice. This is mainly due to the fact that the support of 2-D prediction filters is discontinuous [6]. Therefore, almost all practical 2-D linear prediction models are non-optimal and differ in the way the model support is chosen and is related to the number of independent prediction coefficients, and also in the extend to which the orthogonality principle is violated. However, it should be noted that , as far as image coding is concerned, most of the proposed 2-D auto regressive models behave almost similarly so that the choice of a particular model is of quantitative, rather than of qualitative concern. Here we shall use the conceptual direct form (exact) prediction filter with full quarter plane support. Thus, if $S=[s(n,m)]_{n,m=0,1,..,N-1}$ is a small-field image, then this field is modeled as

$$s(n,m) = -\sum_{(k,l)\neq(0,0)} a_{k,l} z^{-k} w^{-l} s(n,m) + r(n,m) \quad (2)$$

where the $a_{k,l}$ are obtained by minimizing

$$\underline{E} = \sum_n \sum_m \underline{e}^2(n,m) \quad (3)$$

$\underline{e}(n,m)$ being the difference between $s(n,m)$ and $\underline{s}(n,m)$ with

$$\underline{s}(n,m) = \sum_{(k,l)\neq(0,0)} a_{k,l}\, z^{-k} w^{-l}\, s(n,m) \quad (4)$$

The prediction coefficients are field adaptive and are computed on fields $S=[s(n,m)]_{n,m=0,2,..,N-1}$ where N is an appropriate analysis field dimension. If the analysis fields overlap, then the support of the filter parameters in the decoder will be less than N.

For image coding purposes, a second order model, that is $k,l = 1,2$, has enough modeling capability. Once the prediction coefficients $a_{k,l}$ have been determined, the prediction residual $r(n,m)$ is to be approximated by a multipulse field $v(n,m)$, which is such that the output $\hat{s}(n,m)$ of the filter $\frac{1}{A(z,w)}$ input with $v(n,m)$ is sufficiently close to $s(n,m)$, which itself is the output of the same filter input with $r(n,m)$. In the next section we show how the excitation signal $v(n,m)$ has to be computed.

3. MULTIPULSE EXCITATION CODING

The multipulse excitation coding technique was introduced in [1] for 1-D signals and is now widely used in high fidelity medium bit rate coding of speech signals. Multipulse coding is but one example [1, 2, 3] of how a prediction residual can be effectively modeled by an analysis-by-synthesis optimization method based on some appropriate fidelity criterion. Here, our objective is to show that such a procedure can also be used to encode images, and we shall restrict ourselves to the multipulse excitation approach as a prototype example from this class of coders.

Basic Algorithm

Let $S=[s(n,m)]_{n,m=0,1,..,L-1}$ be an image field of size L^2 pels. $L \leqslant N$, where N is the dimension of the prediction analysis field. Similarly, let $\hat{S}$, V and E be LxL field representations of the signals $\hat{s}(n,m)$, $v(n,m)$ and $e(n,m)$ of figure 1. Also let $\sigma,\hat{\sigma},\nu$ and ϵ be 1-D (row oriented) representations of these fields. That is, denoting

$$e_f = [\underbrace{00..}_{f}0100..0] \qquad f=0,1,..,L-1 \quad (5)$$

we have

$$\sigma = [e_0 S \cdots e_f S \cdots e_{L-1} S] \quad (6)$$

and similarly for $\hat{\sigma},\nu$ and ϵ.

Put

$$1.\ H(z,w) = \frac{1}{A(z,w)} W(z,w) = \sum_{i,j=0}^{\infty} h_{i,j} z^{-i} w^{-j} \quad (7)$$

$$2.\ H = [h_{i,j}]_{i,j=0,1,2,..} \quad (8)$$

$$3.\ H_T = [h_{i,j}]_{i,j=0,1,..,L-1} \quad (9)$$

$$4.\ z^{-p} w^{-q} H_T = [z^{-p} w^{-q} h_{i,j}]_{i,j=0,..,L-1} = [h_{i-p,j-q}]_{i,j=0,..,L-1} \quad (10)$$

with $h_{kl}=0$ if $k<0$ or $l<0$,

$$5.\ \mu_{pq} = [e_0 z^{-p} w^{-q} H_T .. e_f z^{-p} w^{-q} H_T .. e_{L-1} z^{-p} w^{-q} H_T] \quad (11)$$

and let H be an ordered stack of the L^2 row vectors μ_{pq} with μ_{00} on top and $\mu_{L-1,L-1}$ at the bottom:

$$\mathrm{H} = [\mu_{pq}]_{p,q=0,1,..,L-1} \quad (12)$$

Finally, let E_0 (or equivalently ϵ_0) be the memory hangover of the filter H(z,w) from the previous fields, and let $E^{(0)}$ (or equivalently $\epsilon^{(0)}$) be the initial approximation error for the current L*L field, that is:

$$\epsilon^{(0)} = \epsilon_0 + \sigma \mathrm{H} \quad (13)$$

Then

$$\epsilon = \epsilon^{(0)} - \nu \mathrm{H} \quad (14)$$

where the entries v(n,m) of ν are of the form

$$v(n,m) = \sum_{k=1}^{K} b_k(i_k,j_k)\delta(n-i_k,m-j_k) \quad K<L. \quad (15)$$

Both the positions (i_k,j_k) and the amplitudes b_k are unknown and are to be chosen such that the l_2 norm of the error vector ϵ is minimal. This problem - as it stands - is difficult to solve in a tracktable way and, therefore, some simple procedure yielding an acceptable suboptimal solution has to be found.

In 1-D speech coding problems [1], the positions i_k and the amplitudes b_k can be computed one at a time without

impairing subjective optimality. Unfortunately, this rather simple procedure does not provide acceptable solutions in 2-D image coding problems. Here, the decoded images turn out to be poor approximations of the reference images in the sense that smooth backgrounds are perceptible distorted and that sharp edges are annoyingly aliased. Therefore, multipulse image coding requires a more involved search procedure to provide suboptimal approximations, yet close enough to the optimal solution. Two possible suboptimal approaches to the pulse search problem have been investigated in [11] for the 1-D case. The methods of [11], when adopted to the image coding problem, dramatically improve the quality of the decoded images, both in terms of objective distortion measures and in terms of subjective judgements.

For lack of space, we cannot drift into all of the technical details of the search procedures derived from [11]. Roughly stated, however, the positions (i_k, j_k) are found one at a time , while the amplitudes b_l, l=1,2,..,k ; k=1,2,..,K are jointly computed.

Thus let

$$\epsilon^{(k-1)} = \epsilon^{(0)} - \nu^{(k-1)}\mathrm{H} \tag{16}$$

where $\nu^{(k-1)}$ is a vector with entries

$$\sum_{l=1}^{k-1} b_l(i_l^*, j_l^*)\delta(n - i_l^*, m - j_l^*) \tag{17}$$

Then the k-th position (i_k^*, j_k^*) is such that the l_2 norm of the vector

$$\epsilon^{(k)} = \epsilon^{(k-1)} - b_k(i_k, j_k)\mu_{i_k, j_k} \tag{18}$$

is minimal, that is

$$\min_{(i_k, j_k)} \epsilon^{(k-1)}[I - \mu_{i_k, j_k}^t \|\mu_{i_k, j_k}\|^{-2}\mu_{i_k, j_k}]\epsilon^{(k-1)t} \tag{19}$$

where ^{t} denotes vector transposition.

Once (i_k^*, j_k^*) has been found, the complete partial excitation field

$$\nu^{(k)} = [\sum_{l=1}^{k} b_l(i_l^*, j_l^*)\delta(n - i_l^*, m - j_l^*)]_{n,m=0,1,..,L-1} \tag{20}$$

is recomputed by solving the following linear least-squares problem

$$\min_{[b_l]_{l=1,2,..,k}} \|\epsilon^{(0)} - \nu^{(k)}\mathrm{H}\|^2 \tag{21}$$

Although this procedure is computational involved, it seems to be necessary and sufficient to obtain high fidelity decoded images. In [11], it is further shown that the inversion of a matrix (of increasing dimension) each time a new pulse position has been found, can be avoided by decomposing the projection spaces using orthogonal spanning vectors instead of span $[\mu_{i_l^*, j_l^*}]_{l=1,2,..,k}$. Although this approach does not reduce the computational load, it allows a straight forward and robust VLSI implementation. Moreover, this orthogonal procedure yields an approximation that is closer to the optimal one than in the approach outlined above. See [12] for more details. Other approximation procedures based on different approximation constraints and leading to more structured algorithms for VLSI are currently under investigation.

4. SOME EXPERIMENTAL RESULTS

Several test images were analyzed with the proposed coding method using different values of the parameters N, L, and K. Here we show some typical images generated by the coder, input with the "girl" test image shown in figure 2. The analysis parameter setting is shown in table 1.

Table 1. Parameters MPE analysis.	
image dimension	256
LP analysis procedure	exact full quarter plane
filter order	2
gray levels	256
analysis field dim. N	32, non overlapping
minimization field dim. L	8
weighting filter W	Identity
# of pulses per min. field K	10

Neither the filter coefficients, nor the excitation field have been further quantized. The other images shown in figures 3 - 5 are the decoded image $\hat{S}$ (figure 3), the prediction error field $\underline{E}$ (figure 4), MPE excitation field V (figure 5), and the error field E (figure 6). In figures 4 - 6 the absolute values of the signals $\underline{e}(n,m)$, $\nu(n,m)$, and $e(n,m)$ are plotted. As an objective measure of the fidelity, we used

Figure 2. Original 8 bit/pel image.

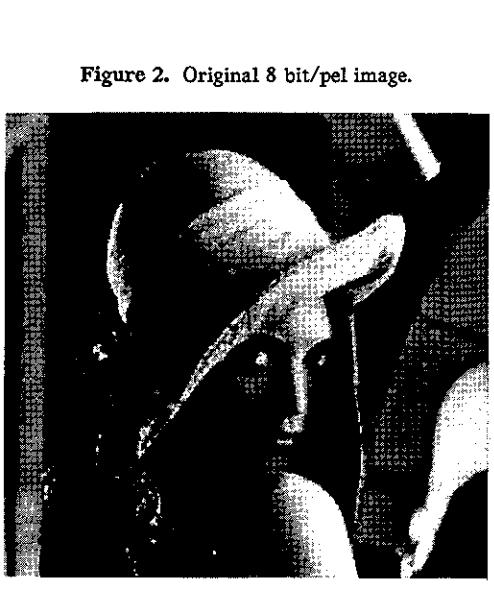


Figure 3. Decoded image using MPE input.

the signal-to-noise ratio, defined by [13]

$$SNR = 10 \log_{10} \frac{(Peak-to-peak\ value\ of\ original\ data)^2}{\frac{1}{I^2} \sum_{m=1}^{I} \sum_{n=1}^{I} [s(n,m) - \hat{s}(n,m)]^2}$$

where I^2 represents the number of pels of the original image. For the "girl" image, we obtained an SNR of 29.08 dB. From the images shown here it can be seen that the MPE excitation signal accurately models the prediction error signal, and provides a high-fidelity sharp-edged decoded image.

Acknowledgment

The authors wish to recognize the assistance and support from the Information Theory Section of the Department of Electrical Engineering. They are particularly indebted to Prof.D.Boekee , Prof.J.Biemond and Dr.R.Plompen for their continuous encouragement.

References

1. B.S. Atal and J.R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," *Proc. IEEE Int. Conf. Acoust., Speech and Signal Processing*, pp. 614-617 (April 1982).
2. Ed. F. Deprettere and P. Kroon, "Regular excitation reduction for effective and efficient LP-coding of speech," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 25.8.1-25.8.4 (March 1985).
3. M.R. Schroeder and B.S. Atal, "Code-Excited Linear Prediction (CELP): high quality speech at very low bit rates," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 937-940 (1985).
4. P. Kroon and E. Deprettere, "Quantization Procedures for Regular-Pulse Excited Coders," *Proceedings European Signal Processing Conference*, (Sep. 1986).
5. J.D. Markel and A.H. Gray, *Linear Prediction of speech*, Springer Verlag, Berlin (1976).
6. T.L. Marzetta, "Two-Dimensional Linear Prediction:Autocorrelation Arrays, Minimum-Phase Prediction Error Filters and Reflection Coefficient Arrays," *IEEE Trans. Acoust.,Speech,Signal Processing* ASSP-28(6) pp. 725-733 (Dec. 1980).
7. S.R. Parker and A.H. Kayran, "Lattice Parameter Autoregressive Modeling of Two-Dimensional Fields—Part I: The Quarter-Plane Case," *IEEE Trans. Acoust.,Speech,Signal Processing* ASSP-32(4) pp. 872-885 (August 1984).
8. S. Ranganath and A.K. Jain, "Two-Dimensional Linear Prediction Models—Part I: Spectral Factorization and Realization," *IEEE Trans. Acoust.,Speech,Signal Processing* ASSP-33(1) pp. 280-299 (Feb. 1985).
9. H. Lev-Ari and S.R. Parker, "Lattice-Filter Modeling of Two-Dimensional Fields," *Proceedings Int.Conf.Acoust.,Speech,Signal Processing* 3 pp. 1317-1320 (1985).
10. P.A. Maragos, R.W. Schafer, and R.M. Mersereau, "Two-Dimensional Linear Prediction and Its Application to Adaptive Coding of Images," *IEEE Trans. Acoust.,Speech,Signal Processing* ASSP-32(6) pp. 1213-1229 (Dec. 1984).
11. P. Kroon and E.F. Deprettere, "Experimental evaluation of different approaches to the multi-pulse coder," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 10.4.1-10.4.4 (March 1984).
12. C. Horne, "High Fidelity Coding of Images by means of Two-Dimensional Linear Prediction Analysis-by-Synthesis," Technical Report # 86xx, (EE) Delft University of Technology, Delft,The Netherlands (1986).
13. A.K. Jain, "Image Data Compression: A Review," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 349-389 (March 1981).

Figure 4. Prediction error field.

Figure 5. MPE excitation field.

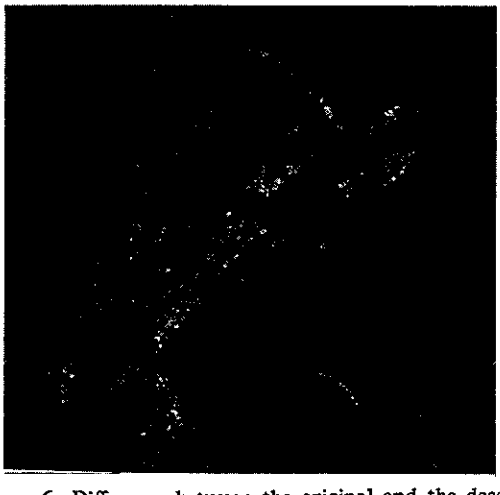


Figure 6. Difference between the original and the decoded image.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EXTENSION OF THE NOTION OF ANALYTIC SIGNAL FOR MULTIDIMENSIONAL SIGNALS. APPLICATION TO IMAGES

Françoise PEYRIN, Yue Min ZHU, Robert GOUTTE

Laboratoire de Traitement du Signal et Ultrasons, UA CNRS 1216, INSA B 502, 69621 Villeurbanne cedex, France.

The concept of analytic signal introduced in the field of telecommunication by J. Ville has many applications in signal processing. Until now, this notion has been limited to the case of 1D signals.
The purpose of this paper is to try to generalize this notion to multidimensional signals. The underlying idea is the redundancy of the Fourier Transform of real signals. A class of generalized multidimensional Hilbert Transform is defined and some of their particularities are discussed. Examples for 1D signals and 2D images are given.

1. INTRODUCTION

The concept of analytic signal was introduced by J. VILLE [1] in the definition of the instantaneous spectrum and of the instantaneous frequency. It has received considerable attention in many domains such as telecommunication, radar, sonar, speech... This notion allows to assign to real time-domain signals a magnitude and phase representation similar to the one used in the Fourier domain. The notion of complex envelope of a real signal, which plays an important role in the study of bandpass signals, also uses the concept of analytic signal [2]. Furthermore it is a way of transposing the principle of causality to the Fourier Domain. Recently, it has been shown that some signal processing techniques (homorphic filtering, complex cepstrum techniques [3], Wigner-Ville Transform [4] [5]...) give better results when applied to analytic signals rather than to real signals.
Generally the notion of analytic signal is limited to monodimensional signal. In this paper we study the extension of this concept to multidimensional signals and its consequences, especially for its application to 2D or 3D image processing.

In the first section the conventional definition of analytic signal associated to a 1D real signal and its main properties are recalled. It is particularly noted that the redundancy of the real signal spectra is eliminated without loss of information when considering the analytic signal. This idea can be exploited to define a class of multidimensional analytic signals. Their definitions are given in the second section where some of their properties are discussed. The transposition of these notions to discrete mono or multidimensional signals is outlined in the last section. Finally numerical simulations of 1D or 2D analytic signals are presented.

2. ANALYTIC SIGNAL [1] [6]

An analytic signal is a complex signal which can be defined by one of the two equivalent properties : i) the imaginary part of the analytic signal is the Hilbert Transform of its real part, ii) the Fourier Transform of the analytic signal is zero for negative frequencies and is twice the Fourier Transform of its real part for positive frequencies.
Let f(t) be a real temporal signal. The analytic signal z(t) associated to f(t) is a complex signal as defined by property (i) or (ii), the real part of which is f(t). It can be written :

$$z(t) = f(t) + i\,Hf(t) \qquad (1)$$

H denoting the Hilbert Transform (HT) defined by :

$$Hf(t) = \frac{1}{\pi}\,v.p \int_{-\infty}^{\infty} \frac{f(\tau)\,d\tau}{t-\tau} \qquad (2)$$

where v.p means that the integral must be considered in a cauchy principal value sense.

For illustration, let us give the HT of two simple signals :

$$f(t)=\cos t \qquad Hf(t)=\sin t \text{ and } z(t)=e^{it} \qquad (3)$$

$$f(t)=\mathrm{rect}_T(t) = \begin{cases} 1 \text{ if } |t|<T/2 \\ 0 \text{ else} \end{cases}$$

$$Hf(t) = \frac{1}{\pi}\,\ln\left|\frac{t+T/2}{t-T/2}\right| \qquad (4)$$

As seen on these examples, the HT is a linear transform which associates to an even (resp. odd) function and odd (resp. even) function.
The complex nature of the analytic signal allows to assign the real signal a magnitude

and a phase representation :

$$z(t) = |z(t)|\, e^{i\,\rho(t)} \qquad (5)$$

$|z(t)|$ can be considered as a model of the signal envelope and $\rho(t)$ is the instantaneous phase of the signal. The notion of instantaneous frequency defined by :

$$\nu(t) = \frac{1}{2\pi}\,\frac{d\rho(t)}{dt} \qquad (6)$$

is currently used in signal processing.

In the Fourier domain definition (1) becomes :

$$Fz(u) = 2\,E(u)\,Ff(u) \qquad (7)$$

where F denotes the Fourier Transform, and E is the Heaviside unity function :

$$E(u) = \begin{cases} 0 & \text{if } u < 0 \\ 1 & \text{if } u > 0 \end{cases} \qquad (8)$$

As the Fourier Transform of an analytic signal has no negative frequencies, the principle of causality has been transposed to the Fourier domain. It is known that the real and imaginary parts of the Fourier Transform of a real causal signal form an Hilbert Transform pair. By duality the inverse Fourier Transform of a causal spectrum, as defined by relation (7) is the analytic signal defined by relation (1), showing the equivalence of these two definitions.
Another application of the notion of analytic signal is in sampling. Indeed as the frequency bandwith of the analytic signal is half the bandwith of the real signal the sampling frequency can be taken two times smaller when working on the analytic signal rather than on the real signal. This economy in sampling rate is obtained to the expense of an additional storage of the imaginary part which is nothing else than a signal in quadrature with f(t). This technic has been applied in some practical situations [7].
An important point to consider is that no information has been lost when using the analytic signal instead of the real signal. Indeed, the Fourier Transform of a real signal having an even real part and an odd imaginary part, all the information about the signal is contained in half the spectra. Then the use of the analytic signal is a way of suppressing the redundancy of the spectra. This consideration will be helpfull to extend this concept to multidimensional signals.

3. EXTENSION TO MULTIDIMENSIONAL SIGNALS

3.1. Definition

Let $f(\vec{x})$, $\vec{x} \in R^n$, be a real multidimensional signal. The Fourier Transform of such a signal is always an hermitian function :

$$Ff(-\vec{u}) = \overline{Ff(\vec{u})} \qquad \vec{u} \in R^n \qquad (9)$$

Then the information contained within this spectra is redundant. The idea we develop is to associate to $f(\vec{x})$ a complex signal $z(\vec{x})$, the real part of which is $f(\vec{x})$, and such as frequencial redundancy is eliminated.
These requirements can be written :

$$z(\vec{x}) = f(\vec{x}) + i\,g(\vec{x}) \qquad (10)$$

and

$$Fz(\vec{u}) = 2\,m(\vec{u})\,Ff(\vec{u}) \qquad (11)$$

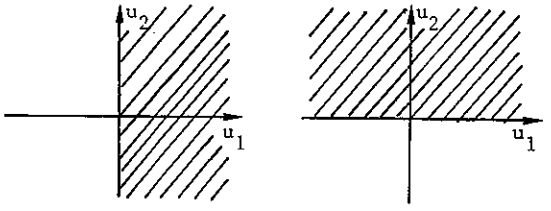
where the filter $m(\vec{u})$ is the characteristic function of a domain D such as- Id ie the application defined by $\vec{u} \rightarrow -\vec{u}$ is a bijection from D to $R^n - D$.
Although there are many solutions to choose $m(\vec{u})$, we shall limit ourselves to the case where the space R^n is divided in two by an hyperplane. In this case we have :

$$m(\vec{u}) = \begin{cases} 1 & \text{if } h(\vec{u}) > 0 \\ 0 & \text{if } h(\vec{u}) < 0 \end{cases} \qquad (12)$$

where $h(\vec{u})$ is a linear form from R^n to R. Explicetely, $h(\vec{u})$ can be written :

$$h(\vec{u}) = \vec{a}.\vec{u} \quad \text{with } \vec{a} \in S^n \qquad (13)$$

where S^n is the unit sphere of R^n.
As an illustration we give examples of possible choices for $m(\vec{u})$ in the case of 2D images in Figure 1.



$$m_1(\vec{u}) = \begin{cases} 1 & \text{if } u_1 > 0 \\ 0 & \text{else} \end{cases} \qquad m_2(\vec{u}) = \begin{cases} 1 & \text{if } u_2 > 0 \\ 0 & \text{else} \end{cases}$$

Figure 1

We call canonical solutions the ones in which the domain is separated by one of the axes. If $(\vec{e}_1, \vec{e}_2, \ldots \vec{e}_n)$ is the canonical base of R^n

they correspond to :

$$\vec{a} = \vec{e}_i \qquad h_i(\vec{u}) = u_i \text{ if } \vec{u} = \sum_{i=1}^{n} u_i\,\vec{e}i$$

and the filters $m(\vec{u})$ will be respectively noted $m_i(\vec{u})$. The two examples of Figure 1 are the canonical solutions for n = 2.

Let us now characterize the analytic signal in the spacial domain. Using (10) and (11) we obtain

$$Fg(\vec{u}) = (1-2m(\vec{u}))\; i\; Ff(\vec{u}) \qquad (14)$$

and including (12) and (13) it leads to :

$$Fg(\vec{u}) = -i \, sign(\vec{a}.\vec{u}) \; Ff(\vec{u}) \qquad (15)$$

This relation can be taken as the frequential definition of a multidimensional Hilbert Transform (MHT) of direction $\vec{a}$. In the spatial domain this operator can be viewed as a convolution with a kernel $k_{\vec{a}}$ satisfying :

$$g = Hf = f * k_{\vec{a}}$$
$$Fk_{\vec{a}}(\vec{u}) = -i \, sign(\vec{a} \cdot \vec{u}) \qquad (16)$$

For instance, for the canonical MHTs (denoted H_i) the kernels $k_i(\vec{x})$ ($\vec{a} = \vec{e}_i$) are given by :

$$k_i(\vec{x}) = \frac{1}{\pi} \; vp \left(\frac{1}{x_i}\right) \prod_{\substack{j=1 \\ j \neq i}}^{n} \delta(x_j) \qquad (17)$$

where δ is the Dirac delta function.
This relation shows that the canonical MHT H_i of a signal $f(\vec{x})$ is the monodimensional HT of the partial function $f(x_1, \ldots x_n)$ considered as a function of x_i.

3.2. Properties

It can be shown that most of the properties of the conventional HT are still valid for the MHT. They will be studied extensively elsewhere. In this sub-section we shall only discuss a few particularities of the MHT.
The first remark is about non unicity. Indeed we have defined a class of MHT depending of the direction $\vec{a}$. Subsequently several choices are possible to define the analytic signal.
A second remark is that in 1D, the introduction of analytic signal is strongly related to the principle of causality for temporal signals. For multidimensional signals which generally are not temporal signals, this notion does not seen to be as evident. In fact the MHT as it has been introduced here, allows to define a kind of "pseudo-causality" for multidimensional signals. This notion which is still non uniquely defined is related to the choice of $\vec{a}$ and can be adapted to the type of processed signals.
Finally, let us note that, in paticular for images, the concepts developed here allow to generalize to image notions such as envelopes spatial phase, and spatial frequency. Furthermore it can be used in image sampling to reduce aliasing effects.

4. CONTINUOUS AND DISCRETE EXAMPLES

For application it is of interest to transpose the notions of analytic signal and Hilbert Transform to discret signals. Even in the monodimensional case this aspect has not widely been considered.
It is not the purpose of this paper to detail this point. We shall only point out that discretisation can be easily performed using the frequential definitions of these operators (relations (8) and (15)). Furthermore this gives a computational procedure to evaluate analytic signals. Let us note that some care has to be taken to the problem of periodisation introduced by sampling.
In the next sub sections in the case of 1D signals and 2D images.

4.1. Monodimensional signals

We have computed the Discrete Hilbert Transform (DHT) of the two test signals $f(t) = \cos t$ and $f(t) = rect_T(t)$. Their continuous Hilbert Transform are given in section 2. Their DHT computed on 256 points are respectively represented on Figure 2 and 3. The results are in agreement with theory.

4.2. 2D images

To illustrate the concept developped in this paper we give the MHT of the two test images.

The first one is the 2D cosine function $f(x,y) = \cos 2\pi (u_o x + v_o y)$. It can be shown that the MHT of the cosine signal is independant of the choice of $\vec{a}$ and is given by :

$$Hf(x,y) = \sin 2\pi (u_o x + v_o y) \qquad (18)$$

It then prolongs the well known 1D HT of the cosine signal. The analytic signal associated to the 2D cosine signal is then always :

$$z(x,y) = \exp(2i\pi(u_o x + v_o y)) \qquad (19).$$

This is of importance for applications. A numerical computation of the MHT is represented on Figure 4. The computation has been performed for a 64 x 64 image.

The second example is the 2D rectangular function $f(x,y) = rect_X(x) \, rect_Y(y)$. Its MHT depends of the direction $\vec{a}$. For instance :

$$H_1 f(x,y) = \frac{1}{\pi X} \ln \left| \frac{x - X/2}{x + X/2} \right| rect_Y(y) \qquad (20)$$

$$H_2 f(x,y) = \frac{1}{\pi Y} \ln \left| \frac{y - Y/2}{y + Y/2} \right| rect_X(x) \qquad (21)$$

In fact these two transforms although they are different, have the same shape up to a rotation. A numerical computation of H_1 is represented on Figure 5 for a 64 x 64 image.

5. CONCLUSION

The notion of analytic signal is strongly related to the causality principle of physical temporal signals. Although this notion has not the same significance for images or more generally multidimensional signals, the concept of analytic signal can also be usefull for multidimensional signals.
In this paper we have tried to generalize this concept to multidimensional signals. The main

difference is that there is no longer unicity. A class of generalized mutidimensional Hilbert Transform have been defined and some of its properties have been discussed. Continuous and numerical examples have been given. They seem to have good properties to be used in some practical situations.

REFERENCES

[1] Ville, J., Câbles et Transmissions, 2°A, n°1, 1948, 61-74.
[2] Roubine, E., Introduction à la théorie de la communication (Masson & Cie, 1970).
[3] Kunt, M., Traitement numérique des signaux, Traité d'électricité, Vol XX, Editions Georgi, 1980).
[4] Claasen, T.A.C.M., and Meckenbraüker, W.F. G., Philips Journ. of Res., Vol 35, 1980, Part I, 217-250, Part II, 276-300.
[5] Flandrin, P., and Escudié, B., Proc. 8th Coll., GRETSI, Nice, 1981, 66-74.
[6] De Coulon, Théorie et Traitement des Signaux, Traité d'électricité, Vol VI, (Editions Georgi, 1984).
[7] Leblanc, L.R., IEEE Trans. on Comm. Tech., Vol COM-17, n°4, 1969, 481-488.

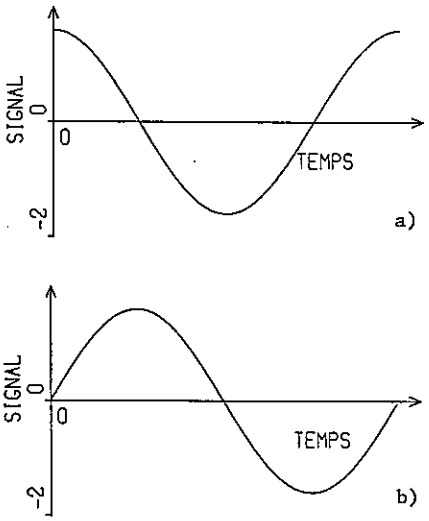


Figure 2
1D cosine function a) and its DHT b)

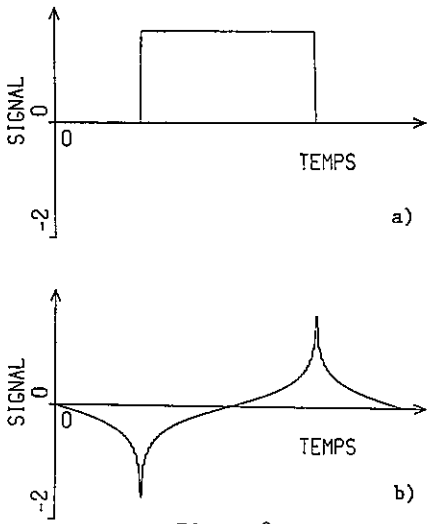


Figure 3
1D rectangular function a) and its DHT b)

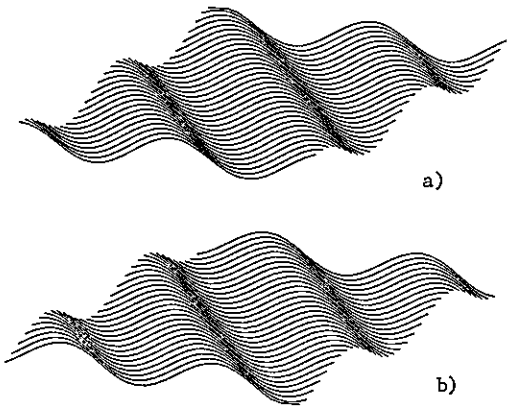


Figure 4
2D cosine function a) and its DHT b)

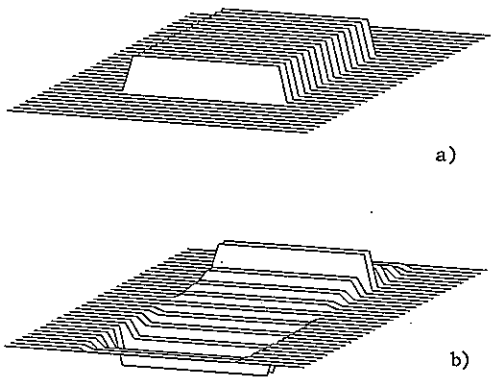


Figure 5
2D rectangular function a) and its DHT b)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

TWO-DIMENSIONAL DISCRETE STOCHASTIC CONGRUENCES IN COMMUNICATIONS†

Salvatore D. MORGERA

McGill University, Department of Electrical Engineering,
McConnell Engineering Bldg., 3480 University Street,
Montréal, Québec H3A 2A7, Canada.

In this work we attempt to take an initial step in demonstrating how simple number theoretic concepts may be applied to digital communications theory applications. The constructions used are two-dimensional and employ the Chinese remainder theorem in order to deal with simultaneous congruences. The setting here is quite different from that found in previous literature. The integers operated on using congruential arithmetic are stochastic; thus, we deal with the congruences as a transformation from one probability distribution to another. One advantage in using a transformation of this type in certain digital communication systems is that a large input dynamic range is mapped into a smaller dynamic range associated with the moduli of the transformation. Subsequent processing on the integers in the smaller dynamic range can usually be carried out more efficiently with little or no increase in the system probability of error. The same sort of number representation that results here, the Sino-representation, has also led to the development of a number of fast digital signal processing algorithms in a deterministic setting.

1. INTRODUCTION

Recently, an increasing number of applications of number theory to communication theory have become evident. One that immediately comes to mind is the application of congruential arithmetic to public-key data encryption in which messages represented by integers are raised to a given power and only the residue or remainder, modulo a prescribed encryption modulus, is transmitted. Also in many cases, the security of data systems and files is assured through congruence relationships - and the operations of error detection and correction for encoded data are carried out using congruential arithmetic in accord with a prescribed modulus. At present, however, examples of knowledge-sharing between these fields are limited. This is probably due to the fact that the two areas have very different historical origins; one field developed largely by pure mathematicians and the other largely by communications engineers. Another, not necessarily mutually exclusive, group of engineers trained in digital signal processing and algorithm computational complexity seem to have been able to draw more out of number theory in their development of efficient transform and convolution algorithms. In large part, however, any crossovers of number theory with engineering have dealt with what we shall call deterministic problems.

In taking our departure from deterministic to stochastic applications of number theory for multi-dimensional problems, we are led into the interesting world of homomorphic images. In the two-dimensional case, a received stochastic sequence is mapped into a generally non-binary image. Although our work on this problem has just begun, we forsee the opportunity to apply a number of techniques from pattern recognition and image processing to the detection of the received symbol sequence in a digital communications system. This represents a considerable departure from conventional methods.

2. TWO-DIMENSIONAL DISCRETE STOCHASTIC CONGRUENCES

The problem of interest is posed as follows. We have a real, discrete time continuous amplitude stochastic variable X, i.e., to every experimental outcome ζ, we assign a number $X(\zeta)$. The domain of X is the set of outcomes, or space S, and its range is the set (ensemble) R of the real numbers $X(\zeta)$. In many engineering problems, particularly those involving signal detection or estimation, X represents a discrete time sample of a stochastic process $X(t,\zeta)$ representing either signal plus noise or noise only.

In order to digitally process the sequence of samples, so as to form, for example, the

† Research supported by Canada NSERC Grant A0912 and Québec FCAC Grant EQ-0350.

statistic suggested by the likelihood ratio, we must also know the so-called dynamic range of the sequence and quantize the amplitudes accordingly. Quantization is equivalent to a discretization in amplitude or value. A simple, linear quantization is obtained by following transformation. Define a discrete stochastic variable Y such that,

$$Y = g(X) = y \qquad y - \tfrac{1}{2} \leq X < y + \tfrac{1}{2} \tag{1}$$

where the ensemble for Y is the countably infinite set Z of integers. In practice, we only process a finite subset of the integers, the size of the subset dependent on the problem at hand. It is clear from (1) that the distribution of Y is given by,

$$P\{Y = y\} = \int_{y-\frac{1}{2}}^{y+\frac{1}{2}} f_X(x)dx \qquad y \in Z \tag{2}$$

where $X \sim f_X(x)$.

It is many times very advantageous from the standpoint of reducing computational complexity in the decision-making process to further transform the stochastic variable Y. One transformation which can reduce the integer set considered from countably infinite to finite (or from finite to a subset thereof), is what we term a two-dimensional discrete stochastic congruence, viz.,

$$Z_1 \equiv Y \bmod n_1 \quad , \quad Z_2 \equiv Y \bmod n_2 \tag{3}$$

where $(n_1,n_2) = 1$, $y \in Z$, and $z_i \in Z_{n_i} := (0,1,2,\ldots,n_i-1)$, $i = 1,2$. The notion of congruence is attributed to Gauss and had an import beyond expectation; however, little is known of the behavior of such a transformation in a stochastic setting, as presented here. The transformation, or mapping $Z \to Z_{n_1} \times Z_{n_2}$ of (3) allows the conversion of a discrete stochastic variable Y into a two-dimensional binary image whose nonzero pixel occurs at the coordinate (Z_1,Z_2). If we consider a sequence of discrete stochastic variables $\{Y_i\}$ each element of which is mapped onto a binary image plane of dimension $n_1 \times n_2$ and the image planes summed and normalized, the resulting image plane will, in general, have a variety of gray levels and not be binary. A transformation of the type given by (3) thus allows us to interpret a sampled, quantized stochastic process as an image on which signal processing operations may be carried out using residue arithmetic. Advantages of the mapping (3) over that of a one-dimensional discrete stochastic congruence, $Z \equiv Y \bmod n$, include reduction of complexity of calculation; accommodation of an increased dynamic range for relatively small values of n_1, n_2; and interpretation of the signal detection problem as a multidimensional pattern recognition/image processing problem. Multi-dimensional maps, or homomorphic images, of this type permit the use of a computationally efficient Sino-representation for the maps [1,2].

Using Euler's theorem [2,3], we can construct a number satisfying (3) as,

$$y = qm + z \tag{4a}$$

where,

$$z = [n_2^{\phi(n_1)} \cdot z_1 + n_1^{\phi(n_2)} \cdot z_2] \bmod m \tag{4b}$$

In (4) Euler's totient function, $\phi(\ell)$, is equal to the number of positive integers less than or equal to the positive integer ℓ and relatively prime to ℓ; the integer $m = n_1 n_2$ and $q \in Z$. To show that (4) satisfies (3), note that $n_j^{\phi(n_i)} \equiv \delta_{ij} \bmod n_j$ $i,j = 1,2$ from Euler's theorem and the assumption that $(n_1,n_2) = 1$. Although the form of (4) is somewhat clumsy in that the constants multiplying z_1,z_2 may be large numbers raised to high powers, this is of secondary concern to us here.

From (3) and (4), the joint probability distribution of (Z_1,Z_2) is given by,

$$P\{Z_1 = z_1,\ Z_2 = z_2\} = \sum_{q=-\infty}^{+\infty} \int_{z-\frac{1}{2}+qm}^{z+\frac{1}{2}+qm} f_X(x)dx \quad z \in Z_m \tag{5}$$

We see from (4b) that computation of z in (5) requires the image pixel position (z_1,z_2). If we wish to reconstruct the value of y from the values (z_1,z_2) uniquely, we must restrict y to the range $m_0 \leq y \leq m_0 + m - 1$, where m_0 is a constant. This follows as a condition of the Chinese remainder theorem (CRT) [3,4]. To place y in the above range and satisfy the condition, we set $q = 0$ and add an appropriate multiple of m to z in (4a); this multiple is given by $a = \lceil (z + m_0)/m \rceil$. The integer $y = z + am$ will satisfy the congruences (3) and fall within the desired range.

In the special case when $X \sim N(\mu,\sigma^2)$, the joint probability distribution (5) may be cast in the form,

$$p\{Z_1 = z_1, Z_2 = z_2\} = \sum_{q=-\infty}^{+\infty} \{\Phi[v_2(z,q)] - \Phi[v_1(z,q)]\} \tag{6a}$$

where,

$$v_1(z,q) = (z - \tfrac{1}{2} + qm - \mu)/\sigma \ , \ v_2(z,q) = (z + \tfrac{1}{2} + qm - \mu)/\sigma \tag{6b}$$

and $\Phi(\cdot)$ is the commonly used notation for the df of a normally distributed stochastic variable. A problem of considerable practical importance involves discrimination using a statistic based on (Z_1,Z_2) for the two underlying hypotheses,

$$X \sim \begin{cases} N(\mu_0,\sigma^2) & H_0 \\ N(\mu_1,\sigma^2) & H_1 \end{cases} \quad \mu_0 < \mu_1 \tag{7}$$

3. SIMULATION RESULTS

In order to accurately compute (6) great care must be taken. We resort to a numerical approach due to the fact that most readily available tables of the Gaussian df either do not contain the argument range or the resolution of interest to us in this work. First, the range of q must be restricted; a reasonable approach is to restrict $q \in [-L\sigma, L\sigma]$, where L is a sufficiently large integer ($L \geq 10$). We then take a sufficiently small increment Δv and perform the indicated numerical integration using the composite corrected trapezoidal rule. Calculations are carried out using double precision on a CDC Cyber 170 system. Representative results for $P\{Z_1 = z_1, Z_2 = z_2\}$ are obtained by selecting $n_1 = 7$, $n_2 = 9$, and $\sigma \in [0.1, 3.2]$.

Figures 1 and 2 illustrate the shape of the log probability distributions for $\sigma = 0.8$ and $\sigma = 3.2$, respectively. These figures reveal the fact that the most probable pixels fall on 45-degree diagonals across the image plane. We have arbitrarily chosen $\mu_0 = 31$ (and $m_0 = 0$) in order to "center" the conditional distribution under H_0 in the images. The conditional distribution under H_1 is easily derived from that under H_0 using the correspondence $z_i(\mu_1) = [z_i(\mu_0) + \mu_1] \bmod n_i$ $i = 1,2$. For H_1, we choose $\mu_1 = 34$; thus, the difference between means $|\mu_1 - \mu_0| = 3$. The one-dimensional marginal distribution for modulus $n = 7$ or $n = 9$ is easily obtained by summing over rows or columns, respectively, of the two-dimensional distribution.

The critical region for rejecting H_0 lies along diagonals bounded by the two thresholds $\{([\mu_1 + \frac{1}{2}] \pm 1) \bmod n_1, ([\mu_1 + \frac{1}{2}] \pm 1) \bmod n_2\}$. Table 1 shows the Bayesian error, P_{ec}, for equiprobable hypotheses numerically computed from $P\{Z_1 = z_1, Z_2 = z_2 | H_i\}$ $i = 1,2$. In the conventional case, a single threshold is placed at $|\mu_1 - \mu_0|/2$ and the resulting error is given by,

$$P_e = \text{Erfc}\,(|\mu_1 - \mu_0|/2\sigma). \tag{8}$$

For large values of the complementary error function argument, the conventional error is computed using the excellent approximation [5],

$$\text{Erfc}(x) \cong [(1-a)x + a(x^2+b)^{\frac{1}{2}}]^{-1}(2\pi)^{-\frac{1}{2}}e^{-x^2/2} \tag{9}$$

where $a = 1/\pi$ and $b = 2\pi$. Also shown in Table 1 is the average signal-to-noise ratio (SNR).

We note that Table 1 Bayesian error values P_{ec} down to an SNR of 5.5dB are extremely close to the exact P_e values, and, incidentally, are lower than those obtained using a one-dimensional discrete stochastic congruence with $n = 7$. At -6.5 db (in fact, for SNR values less than approximately -1.0dB), there appears to be a definite advantage to using a two- or higher-dimensional mapping. This is due to the fact that $2\sigma/m < 1$, even for $\sigma = 3.20$, and that the decision regions are better separated in a geometrical sense than in the conventional case.

σ	SNR(dB)	P_{ec}	P_e
0.10	23.5	3.65×10^{-51}	3.67×10^{-51}
0.20	17.5	3.19×10^{-14}	3.19×10^{-14}
0.40	11.5	8.84×10^{-5}	8.84×10^{-5}
0.80	5.5	3.04×10^{-2}	3.02×10^{-2}
1.60	-0.5	1.72×10^{-1}	1.74×10^{-1}
3.20	-6.5	2.40×10^{-1}	3.19×10^{-1}

TABLE 1. Bayesian Error for Two-Dimensional Stochastic Congruence. Means $\mu_0 = 31$, $\mu_1 = 34$; modulo $m = n_1 n_2 = 63$.

The likelihood ratio for hypothesis testing is given by,

$$\ell(z_1,z_2) = \frac{P\{Z_1 = z_1,\ Z_2 = z_2 | H_1\}P_1}{P\{Z_1 = z_1,\ Z_2 = z_2 | H_0\}P_0} \tag{10}$$

where P_i is the a priori probability of H_i $i = 0,1$. In the event that a statistically independent discrete time sequence $\{X_i | 1 \leq i \leq N\}$ is available over an observation interval (0,T), it is completely evident that the zero-memory nonlinearity of the congruence operation renders the corresponding sequence $\{(Z_{1i}, Z_{2i}) | 1 \leq i \leq N\}$ statistically independent. The likelihood ratio for the sequence over (0,T) is then given by,

$$\ell(\underline{z}) = \frac{\prod_{i=1}^{N} P\{Z_{1i} = z_{1i},\ Z_{2i} = z_{2i} | H_1\}P_1}{\prod_{i=1}^{N} P\{Z_{1i} = z_{1i},\ Z_{2i} = z_{2i} | H_0\}P_0} \tag{11}$$

The log-likelihood ratio $\ell n[\ell(z_1,z_2)]$ is shown in Figure 3 for $\sigma = 0.8$, H_0 and H_1 as described above, and $P_0 = P_1$. In calculating these results, single precision computation and a "floor" (lowest value) of 1.81×10^{-33} is used for the probability distributions.

4. DISCUSSION: TWO-DIMENSIONAL PROBLEM

Interesting applications of the two-dimensional approach have been suggested, one of which is viewing digital MASK (or MPSK) detection as a pattern recognition or image processing problem. The proposed scheme is shown in Figure 4. In the MASK case, M symbols are transmitted by M equally likely pulses $\pm A_i p(t)$ $i = 1,2,\ldots,M/2$, where A_i denotes the amplitude of the ith pulse and M is assumed even. An M-ary symbol conveys the information of $k = \log_2 M$ bits. The noise is assumed additive white Gaussian and the detector is preceder by a matched filter, i.e., $H(\omega) = P^*(\omega)e^{-j\omega T}$ to optimize the SNR [7]. If we choose $n_1 = 7$, $n_2 = 9$, we require that

$\max_i\{A_i\} < \lfloor m/2 \rfloor = 31$ to avoid folding and allow for a multiple of the noise standard deviation (recall that we assume $\mu_0 = \lfloor m/2 \rfloor$). If we also keep pulse amplitudes apart by two levels to avoid overlapping decision regions in the image plane, we may confortably accommodate $M = 16$ alternatives or $k = 4$ bits in the above map (actually, 21 alternatives are possible). In the conventional case, MASK detection is carried out pulse-by-pulse using a set of amplitude thresholds; the difference here is that we allow a number, N, of MASK pulses to be demodulated and mapped onto the same plane prior to detection. This so-called composite map then potentially contains a total of Nk bits and is generally non-binary, due to the fact that similar pulse amplitude levels can reoccur within a sequence of N symbols. The intensity level of each pixel reflects the number of times the corresponding pulse amplitude level occurs within the sequence of N symbols. Such an interpretation permits the concurrent detection of a number of symbols with an associated SNR gain dependent on the pulse-to-pulse noise statistics.

We now present a detection procedure for the composite map; although the approach is not the most computationally efficient technique possible, it does permit a most interesting visualization from the point of view of multi-dimensional signal processing. Let $h(z_1,z_2)$ be the reference image template and $x(z_1,z_2)$ be the received image. In the presence of white noise, an optimum detector forms the 2-D convolution.

$$y(z_1,z_2) = x(z_1,z_2) \;**\; h(z_1,z_2) \tag{11}$$

It can be easily shown that the frequency domain counterpart of (11) is,

$$Y(k,\ell) = X(k,\ell)\, H(k,\ell) \tag{12}$$

where X, H, and Y are the 2-D DFT's of x, h, y, respectively. If $x(z_1,z_2)$ are $n_1 \times n_2$ arrays so that $y(z_1,z_2)$ is a $p_1 \times p_2$ array where $p_1 = 2n_1-1$ and $p_2 = 2n_2-1$, then the 2-D DFT's must all contain at least $p_1 \times p_2$ points. The form of the 2-D DFT $V(k,\ell)$ of the sequence $v(z_1,z_2)$ is given by,

$$V(k,\ell) = \sum_{z_1=0}^{p_1-1} \sum_{z_2=0}^{p_2-1} v(z_1,z_2) W_{p_1}^{kz_1} W_{p_2}^{\ell z_2} \tag{13a}$$

where,

$$W_{p_i} = e^{-j2\pi/p_i} \qquad i = 1,2 \tag{13b}$$

Using $n_1 = 7$, $n_2 = 9$, Figure 5a shows a typical $n_1 \times n_2$ reference image template $h(z_1,z_2)$ for $N = 32$ symbols padded with zeroes to size $p_1 \times p_2$. Figure 5b shows the associated $n_1 \times n_2$ receive image $x(z_1,z_2)$ also padded with zeroes to size $p_1 \times p_2$. The standard deviation of the noise is selected as $\sigma = .707$. The inverse DFT of the product of the DFT's of the zero padded arrays corresponds to the linear convolution (11) of $x(z_1,z_2)$ and $h(z_1,z_2)$ [7]. For a fixed reference image template, $H(k,\ell)$ can be precomputed and stored. In this case, the number of real multiplications or real additions using the FFT to implement the DFT's is given by $O[4n_1n_2\log_2 n_1n_2 + 10n_1n_2]$. If the MASK pulse time duration is T s, there are NT s in which to carry out the above processing. Number theoretic transforms offer a more computationally attractive alternative to the above use of the Fourier transform [8]. Figure 6 is a plot of the linear convolution array $y(z_1,z_2)$ obtained by taking the inverse DFT of $Y(k,\ell)$. It is easy to show that the convolution array $y(z_1,z_2)$ also exhibits energy dispersion along 45-degree diagonals as induced by the mapping (3).

To detect the N symbol sequence, we sum the pixel intensities in an interval lying on the 45-degree ridge passing through the zero lag point (0,0). The interval length should be equal to approximately twice the pulse amplitude decision region; in this case, two pixels on either side of (0,0) are summed with the zero lag point and thresholded. This operation could be carried out by altering the transfer function of the reference image to $H'(k,\ell) = H(k,\ell)\, F(k,\ell)$, where $F(k,\ell)$ is a 45-degree rotated "velocity selecting" filter as used in seismic analysis [9]. The cascade combination of H and F may now be precomputed and used in place of H in (12). Assuming that the joint probability distribution of the composite map for $N = 32$ may be expressed at the sum of the joint distributions for each symbol, we estimate $P_{ec,M}$, the error probability of an M-ray symbol, is 1.67×10^{-4}. This value is extremely close to the error expected for a conventional MASK system having $M = 16$ and much lower than that for $M = 32$.

5. CONCLUSIONS

In this work we have examined the use of one- and two-dimensional stochastic congruences in certain digital communications systems. Such transformations considerably alter the input probability distribution, but are found to only minimally affect the Bayesian error of the system.

Treatment of the two-dimensional problem for MASK systems used the Chinese remainder theorem followed by multi-dimensional signal processing techniques. The approach is different than the conventional pulse-by-pulse detection in that detection is carried out on a sequence of pulses represented as a pattern on a composite map, or image. We take a frequency domain approach to this pattern recognition problem. Other, more computationally efficient, approaches are possible.

Our hope is that this work aids the understanding of the congruence as a transformation in a

stochastic setting. Through the use of simultaneous congruences or homomorphic mapping, we have also attempted to expose the alternative of using pattern recognition and image processing techniques in communication systems. Many interesting problems are open in the more than two-dimensional case, although there is a certain appeal to being able to see a one-dimensional sequence displayed as an image.

ACKNOWLEDGEMENT

Many thanks are due to Ms. Marie St-Germain for doing a highly professional job in preparing this manuscript.

REFERENCES

[1] M. Lauer, "Computing by Homomorphic Images", in Computer Algebra - Symbolic and Algebraic Computation, B. Buchberger, et al, Eds. Vienna: Springer-Verlag, 1983, pp. 139-168.

[2] M.R. Schroeder, Number Theory in Science and Communication. Berlin: Springer-Verlag, 1984.

[3] H.J. Nussbaumer, Fast Fourier Transform and Convolution Algorithms. Berlin: Springer-Verlag, 1981.

[4] J.H. McClellan and C.M. Rader, Number Theory in Digital Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[5] P.O. Börjesson and C-E.W. Sundberg, "Simple Approximations of the Error Function Q(x) for Communications Applications", IEEE Trans. Comm., Vol. COM-27, pp. 639-643, March 1979.

[6] A. Papoulis, Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.

[7] R.M. Mersereau and D.E. Dudgeon, "Two-Dimensional Digital Filtering", Proc. IEEE, Vol. 63, pp. 610-623, April 1975.

[8] R.C. Agarwal and C.S. Burrus, "Number Theoretic Convolutions to Implement Fast Digital Convolution", Proc. IEEE, Vol. 63, pp. 550-560, April 1975.

[9] B. Sako and K. Hirano, "Design of Recursive Digital Filters for the Selection or Rejection of a Particular Velocity in Seismic Signals", Circuits, Systems, and Signal Processing, Vol. 3, pp. 177-191, 1984.

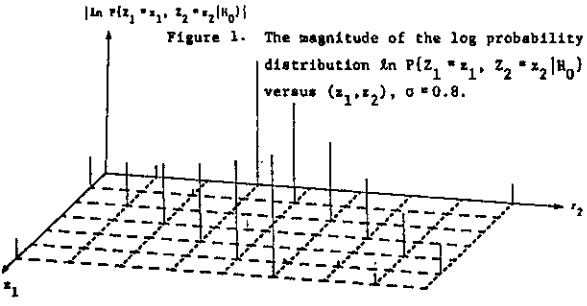


Figure 1. The magnitude of the log probability distribution $\ln P(Z_1=z_1, Z_2=z_2|H_0)$ versus (z_1,z_2), $\sigma=0.8$.

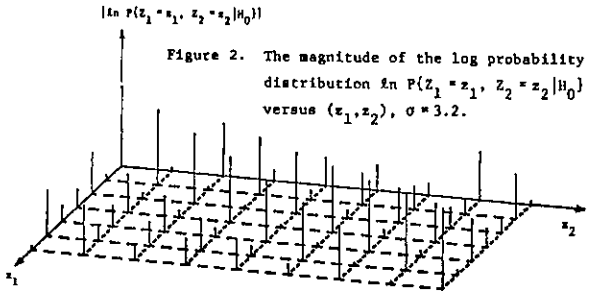


Figure 2. The magnitude of the log probability distribution $\ln P(Z_1=z_1, Z_2=z_2|H_0)$ versus (z_1,z_2), $\sigma=3.2$.

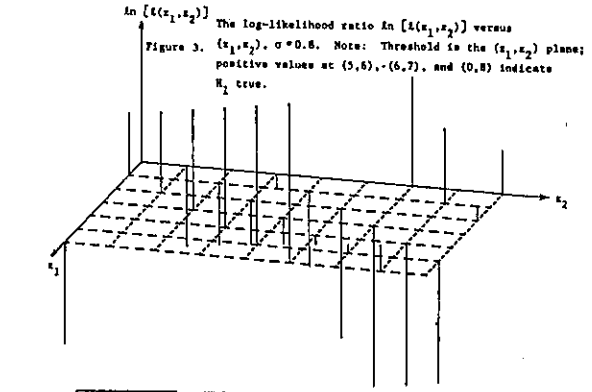


Figure 3. The log-likelihood ratio $\ln[\ell(z_1,z_2)]$ versus (z_1,z_2), $\sigma=0.8$. Note: Threshold is the (z_1,z_2) plane; positive values at (5,6),-(6,7), and (0,8) indicate H_1 true.

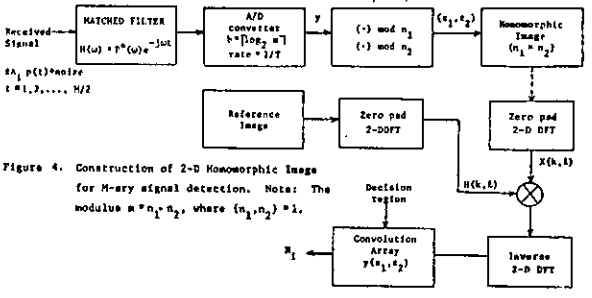


Figure 4. Construction of 2-D Homomorphic Image for M-ary signal detection. Note: The modulus $n=n_1=n_2$, where $(n_1,n_2)=1$.

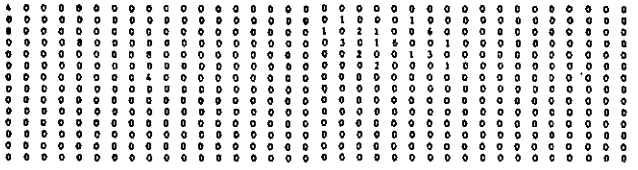


Figure 5a. The reference image, N = 32. The point (0,0) is at upper left-hand corner.

Figure 5b. The received image, N = 32. The point (0,0) is at upper left-hand corner.

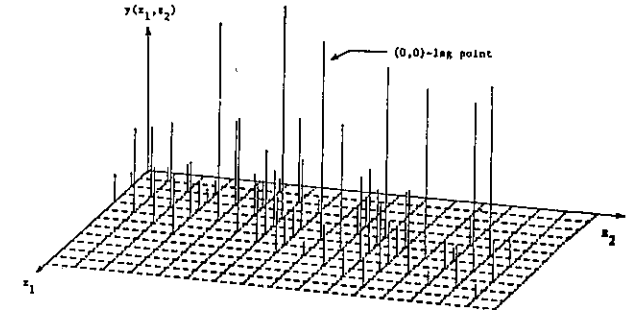


Figure 6. The linear convolution array $y(z_1,z_2)$ versus (z_1,z_2) for the N symbol MASK sequence of Figure 5. Note: The (0,0)-lag point of the convolution is indicated.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REAL-TIME SYSTOLIC ARRAY PROCESSOR FOR 2-D SPATIAL FILTERING

T. Aboulnasr and W. Steenaart

Department of Electrical Engineering
University of Ottawa
Ottawa, Ontario
Canada K1N 6N5

ABSTRACT. In this paper, the use of systolic arrays to implement 2-D Local State-Space digital filters for real-time image processing is presented. The array used is composed of ROM/adder cells and is shown to be 100% efficient. The size of the array is equal to the overall system matrix which is quite smaller than arrays where each processor corresponds to a pixel in the image. The use of ROM's along with Local State-Space implementation provides improved performance under finite-length register restrictions while keeping the price reasonably low.

1. INTRODUCTION

Advances in VLSI technology indicate that further increases in speed will have to be obtained through parallel operation and/or pipelining arrays. Ideal candidates for VLSI implementation have to be regular, modular and require only local communication.

Systolic arrays are the simplest kind of multiprocessor configurations [1] meeting these requirements. Each processor is allowed to communicate only with its nearest neighbors Systolic arrays based on a multiply/add processor cell are naturally suited for efficient matrix computations [1-2]. Since matrix manipulation is widely used in many D.S.P. algorithms, e.g., DFT and FIR filtering, systolic arrays are especially attractive in this field.

The first step is to formulate the algorithm in matrix form. It follows that one can find a systolic array implementation for an algorithm that is already in matrix form, namely, state space realizations of digital filters.

In this paper, a 100% efficient systolic array implementation for 2-D LSS state-space digital filters is proposed for real-time image processing.

2. SYSTOLIC ARRAY IMPLEMENTATION OF 1-D STATE-SPACE DIGITAL FILTERS

The array considered here was introduced in [2] and is shown in Fig. 1. The array is used to multiply an MxR matrix A by an RxN matrix B or equivalently by a series of vectors $\underline{B}(1)$, $\underline{B}(2)$, There is a delay of R clocks for an MxR array before the first output component $C_1(i)$ appears. This is followed by the second component $C_2(i)$ one clock later in the next column and so on. For a constant flow of input vectors $\underline{B}(i)$, all processors in the array will be busy all the time (following an initial set up period) and the array is 100% efficient. Since coefficients are fixed in the cells, the multipliers can be replaced by PROM's thus eliminating the need to initially load the coefficients in the cells as well as improving accuracy by eliminating the error due to coefficient quantization. This, along with the cost reduction possible, makes the array of Fig. 1 using ROM/adder cells very attractive. Consider using the array to implement a 1-D zero-input state-space digital filter described by

$$\underline{x}(n+1) = A\underline{x}(n) \qquad (1)$$

where A is a MxM coefficient matrix, $\underline{x}(n)$ is Mx1 state vector. Computation of (1) using the array in Fig. 1 is straightforward. The problem is that it takes M clocks for the output $\underline{x}(n+1)$ corresponding to an input $\underline{x}(n)$ to be available. During that delay most of the array remains idle. Thus, the efficiency is drastically reduced.

3. SYSTOLIC ARRAY IMPLEMENTATION OF 2-D LSS FILTERS

The Local State-Space model used here to implement 2-D filters was introduced in [3]:

$$\underline{x}_{11}(m,n) = A\underline{x}(m,n) + \underline{b}u(m,n) \quad (2)$$

$$y(m,n) = \underline{c}^t\underline{x}(m,n) + du(m,n) \quad (3)$$

where

$$\underline{x}(m,n) = \begin{bmatrix} \underline{x}^h(m,n) \\ \underline{x}^v(m,n) \end{bmatrix}$$

= Mx1 Local state vector at point (m,n)

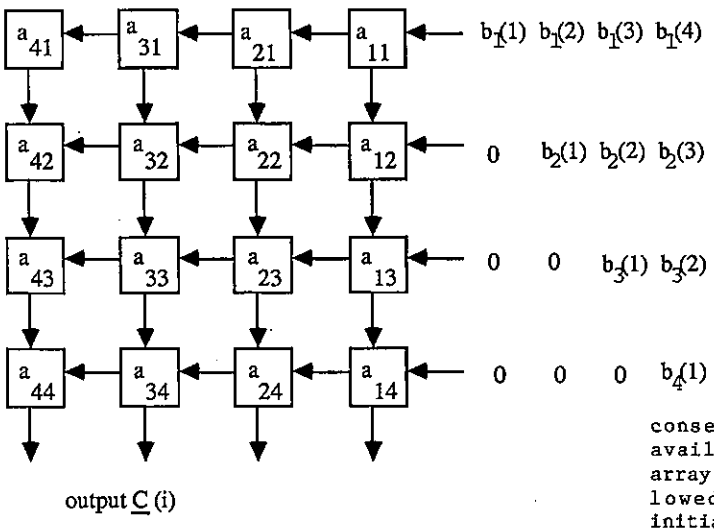


Figure 1(a): Square Systolic Array , $\underline{C}(i) = A * \underline{B}(i)$, M=R=4

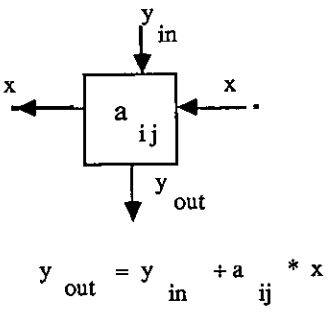


Figure 1(b) : Cell operation

$$\underline{x}_{11}(m,n) = \begin{bmatrix} \underline{x}^h(m+1,n) \\ \underline{x}^v(m,n+1) \end{bmatrix}$$

and u(m,n), y(m,n) are the input and output images respectively of N×N dimension each; A, $\underline{b}$, $\underline{c}^t$,d are the filter coefficients of appropriate dimensions. $\underline{x}^h$is the horizontal state vector and $\underline{x}^v$ is the vertical state vector. Thus, given $\underline{x}(m,n)$ and u(m,n), we can find $\underline{x}^h(m+1,n)$ and $\underline{x}^v(m,n+1)$ by evaluating (2). First, let us consider the zero-input case for an NxN picture shown in Fig. 2 with initial conditions for $\underline{x}^h(1,n)$, $1 \leq n \leq N$ and $\underline{x}^v(m,1)$, $1 \leq m \leq N$. The systolic array to be used here is the same one given in Fig. 1. However, the same array can be made 100% efficient because of the inherent nature of 2-D systems.

Consider the 4 x 4 array in Fig. 1, $\underline{x}(1,1)$ enters the array and after a delay of 4 clocks (1 cycle), components of $\underline{x}^h(2,1)$ start appearing at the array output followed by components of $\underline{x}^v(1,2)$ at successive columns.

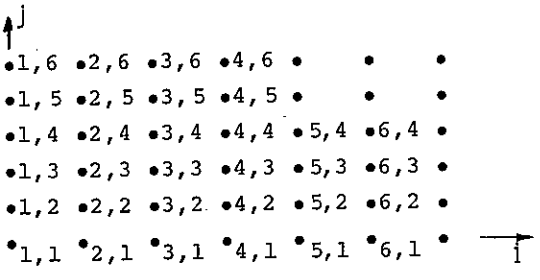


Figure 2 : 2-D Image Notation.

During that delay, no new input could be entered to the array and we have to enter three consecutive zero vectors until $\underline{x}^h(2,1)$ becomes available. As soon as $\underline{x}^h(2,1)$ appears at the array output, it is fed back to the input followed by $\underline{x}^v(2,1)$ which is already given as initial conditions. It would take another cycle (4 clocks) before $\underline{x}_{11}(2,1) = [\underline{x}^h(3,1):-\underline{x}^v(2,2)]$ starts appearing at the array output. However, the array can take $\underline{x}(1,2)$ as input while waiting since it is already available ($\underline{x}^h(1,2)$ as initial condition and $\underline{x}^v(1,2)$ as previous array output). Thus, this time we will only need to enter two consecutive zero vectors before $[\underline{x}^h(3,1):\underline{x}^v(2,2)]$ starts appearing at output followed by $\underline{x}_{11}(1,2)=[\underline{x}^h(2,2):-\underline{x}^v(1,3)]$. Similarly, we will need to enter one zero vector in the third cycle following $\underline{x}(3,1)$, $\underline{x}(2,2)$ and $\underline{x}(1,3)$.

Finally, starting the 4th cycle, the array takes in $\underline{x}(4,1)$ followed by $\underline{x}(3,2)$ then $\underline{x}(2,3)$ and finally $\underline{x}(1,4)$. By then the output $\underline{x}_{11}(4,1)$ is available and we can start working on the next cycle beginning with $\underline{x}(5,1)$ in the same manner until all points in this horizontal 4×N strip are processed. The whole process is repeated for subsequent strips until the whole image is processed.

Thus, recursion is done on the diagonal so that while waiting for the output due to a particular point $\underline{x}(m,1)$, successive points in the "past" of this point can be processed, i.e. $\underline{x}(m-1,2)$, $\underline{x}(m-2,3)$,...$\underline{x}(m-M+1,M)$. By then, the output due to $\underline{x}(m,1)$ will be ready and processing of the next diagonal can start. Fig. 3 shows the sequence of recursion. It is easily seen that after an initial setup period of (M-1) cycles (1 cycle = M clocks), all cells of the array are busy doing useful (nonzero) computations all the time. Also note that while $\underline{x}^h(i,j)$ appears exactly when needed, $\underline{x}^v(i,j)$ appears one clock early and has to be delayed before being entered to the array.

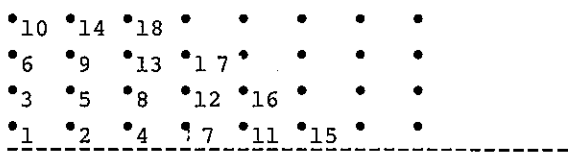


Figure 3 : Sequence of Computation.

4. THE COMPLETE 2-D LSS SYSTOLIC ARRAY PROCESSOR

The last cycle where the array is 100% busy is the cycle starting with the processing of $\underline{x}(N,1)$. The next cycle should have started with $\underline{x}(N+1,1)$ followed by $\underline{x}(N,2)$,$\underline{x}(N-1,3)$ and $\underline{x}(N-2,4)$. However, $\underline{x}(N+1,1)$ is irrelevant since it is beyond the boundaries of the NxN image. At the same time, when processing the next 4xN strip, we only need one useful clock in the first cycle (the remaining three have zero inputs). So instead of wasting the one clock period allotted to $\underline{x}(N+1,1)$ we process $\underline{x}(1,5)$ in that period, then proceed with $\underline{x}(N,2)$, $\underline{x}(N-1,3)$, $\underline{x}(N-2,4)$ as they become available. For the next cycle, we have two vacant clock periods in the current strip (corresponding to $\underline{x}(N+2,1)$, $\underline{x}(N+1,2)$) while we need exactly two clock periods for the second cycle of the following strip to process $\underline{x}(2,5)$ then $\underline{x}(1,6)$. It follows that we can overlap two consecutive strips so that any unused clock periods in one strip are used by the following strip to keep the array active all the time with the exception of an initial and a final period of (M-1) cycles each.

Initial Conditions

To process an NxN image, any LSS filter would require initial values for state vector components that cannot be computed otherwise, i.e.

$$\underline{x}^v(i,1),\ 1 \leq i \leq N;\ \underline{x}^h(1,j),\ 1 \leq j \leq N$$

with this information, we can start recursion from (1,1), for which the state vector $\underline{x}(1,1)$ is known. Every time a pixel on the boundary of the image is to be processed, the component of the state vector representing past information, has to be provided. For the recursion sequence in Fig. 3, we need one $\underline{x}^h(1,j)$ during each of the first M cycles. This can be achieved by fairly slow memories even for real-time oeration.

The storage requirements for initial conditions for the horizontal bottom boundary are different. Consider processing the first strip (rows 1,2,3,4 for M = 4) as shown in Fig. 4. The output due to processing at any point $\underline{x}(i,4)$ will include a vertical component $\underline{x}^v(i,5)$ which though it is not needed when processing this first strip, represents the initial conditions for the following strip Thus, these $\underline{x}^v(i,5)$ need to be stored for future use. Following the recusion sequence in Figure (3), every time we need to access one initial condition for row 1, we have to store one initial condition for row 5. This can be implemented by a FIFO memory. Since this operation needs to be performed once every cycle not every clock, slow memories can be used. Multiplexers are also needed to determine whether the output of the array or the initial condition is to be used as input for the next clock.

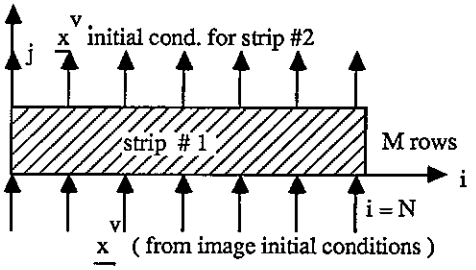


Figure 4 : Vertical initial conditions

Input/Output Considerations

To implement the LSS filter completely, we need to implement (2) and (3) for $u(m,n) \neq 0$. Equations (2), (3) can be combined together as:

$$\begin{bmatrix} y(m,n) \\ \hline \underline{x}^h(m+1,n) \\ \underline{x}^v(m,n+1) \end{bmatrix} = \left[\begin{array}{cc} c^t & d \\ \hline A & \underline{b} \end{array}\right] \left[\begin{array}{c} \underline{x}^h(n,n) \\ \underline{x}^v(m,m) \\ \hline u(m,n) \end{array}\right] \quad (4)$$

Obviously, (4) is still vector = matrix x vector and exactly the same discussion applies except for the fact that the array of Fig. (1) will have to be augmented by one extra column (to compute y(m,n)) and one extra row to account for the nonzero input term as shown in Fig. (5). Fig. (6) gives the details of one array cell.

5. TIMING CONSIDERATIONS/ACTUAL COMPONENTS

Let the basic system clock be of period P. This is the period required for any given cell to produce its output To process the whole NxN image, we need

$$T = P\{[N.M].[\frac{N}{M}]_R + T_e\} \quad (5)$$

where T = total time needed to process one NxN image in strips of size MxN each, M being the size of the A matrix in (2).

$[\frac{N}{M}]_R$ = the number of MxN strips to be processed=integer value of $[N/M+1]$

T_e = Number of clocks required for the last M-1 cycles of the last strip = M(M-1)

Thus, for real time applications, we must have

$$P < \frac{1/30}{N.M.\ [\frac{N}{M}]_R + M^2-M}$$

for a general second order filter (M=6) and a typical image of size 512×512, this gives P < 126 n.sec.

From Fig. 6, the minimum clock period P needed for completion of necessary operations is given by

$$P_{min} = t_1 + t_{PROM} + t_{Adder} + \max\{(t_{MUX} + t_{delay}), t_2\}$$

where

t_1, t_2 = setup plus propagation time for latch #1, #2 respectively
t_{PROM} = PROM access time
t_{adder} = adder propagation time
t_{MUX} = propagation time for multiplexer
t_{delay} = setup for delay

Table 1 gives a list of possible components for implementing the array for real-time operation with P = 90 n.s. The largest picture size (=power of two) for real time processing using the 7 x 7 systolic array proposed here is 512 x 512. To process 1024 x1024 image, we can split the image into four sub-images of size 1024 x256 each and use four arrays. It can also be shown that filters of orders exceeding 40 (for P = 100 n.s.) implemented as cascade of second order sections can process a 512×512 image in real-time.

TABLE 1

Possible Components for Real-Time Implementation of the Array

Function	Part Number	Device Delay
1.PROM	AMD 27520	45 n.sec.
2.Adder	TI 74LS181 (ALU) + TI 74LS182	19 n.s.
3.Latch	TI 74'S113	6 ~9 n.s.
4.MUX	TI 74 ALS158	5 n.s.
5.Delay	TI 74'S374	10 n.s.

Comparing this realization to the structure suggested in [4], it is clear that high speed column delays used in [4] pushed the price of the hardware to above $5,000. The cost of the array proposed here is closer to 20% of that price. In addition, the array has no coefficient quantization errors, improved finite wordlength performance along with the inherent properties of systolic array design.

CONCLUSION

In this paper, the use of systolic arrays for the implementation of 2-D LSS filters is suggested. The array is built from ROM/Adder cells. For a general second-order filter, an array of size not exceeding 7×7 can process 512×512 image in real-time. Several arrays can be used for real-time implementation of higher order filters and/or processing of larger images.

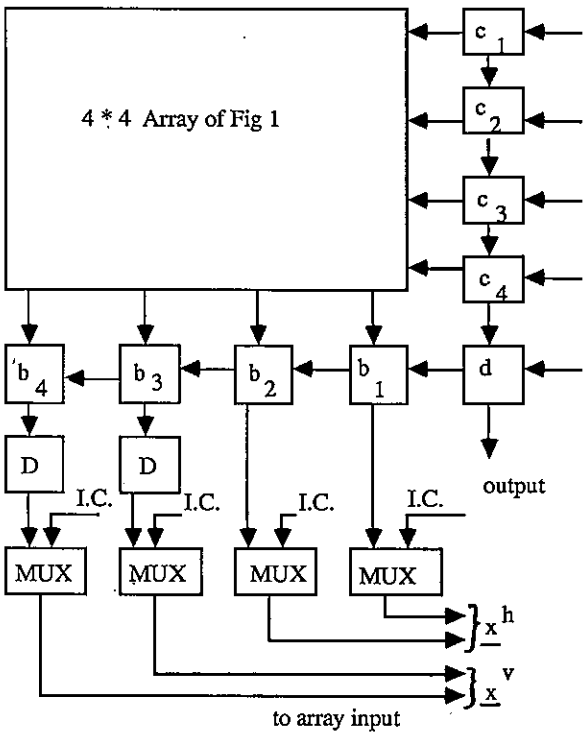


Figure 5: Overall Array

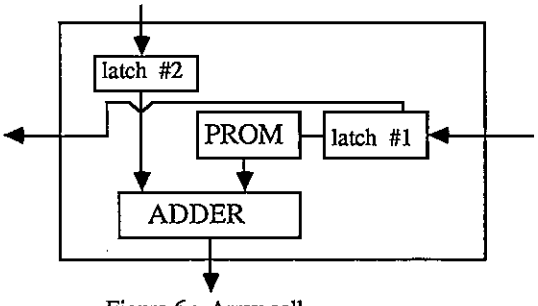


Figure 6: Array cell

REFERENCES

[1] Kung, H.T., "Why Systolic Architectures?", IEEE Trans. on Computer, Jan. 1982.

[2] Urquhart, R.B. and Wood, D., "Systolic Matrix and Vector Multiplication Methods for Signal Processing", IEE Proc., Vol. 131, Pt. F, No. 6, October 1984.

[3] Roesser, R.P., "A Discrete State-Space Model for Linear Image Processing", IEEE Trans. on Automatic Control, Vol. AC-20, pp. 1-10, February 1975.

[4] Ty, K.M. and Venetsanopoulos, A.N., "Two-Dimensional Digital Filters with Minimum Cycle Time", IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 1527-1530, Florida, 1985.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

BLOCK PARALLEL PROCESSING OF 2-D SIGNALS FILTERS BASED ON THE STATE-SPACE MODEL

Vassilis MERTZIOS

Department of Electrical Engineering
Democritus University of Thrace
67100 Xanthi, Greece

A new structure is presented for the block parallel processing of two-dimensional digital signals, based on a block-state space model characterized by high inherent parallelism. The block parallel processing model is derived from a 2-D block state-space model by applying the input vector decomposition idea. The optimal block dimensions are determined in order to minimize the critical number of notrivial multiplications per output sample. The data throughput-delay, estimated in the case of optimal block dimensions in the proposed model, is substantially reduced relatively to that one which has been estimated with the canonical state-space model. The presented structure is ideally suited for computer use and VLSI implementation.

1. INTRODUCTION

The implementation of linear shift-invariant, (LSI) digital filters is achieved on a digital computer or using special purpose hardware. In both cases the digital filter is described by a computational algorithm which gives the input-output relation. The state-space model is ideally served as an algorithm and provides an infinity of structures by using linear similarity transforms. The specific implementation, which is chosen each time, influences: i) the computational complexity (number of real multiplications and additions), ii) the memory requirements, iii) the number of dynamic elements, iv) the number of register coefficients, v) the effects of finite register length, vi) the data throughput-delay and vii) the cycle time. A number of structures have been studied to minimize the number of multiplications and the number of dynamic elements (i.e. minimal realizations). Furthermore, optimal structures with respect to the minimization of the roundoff noise have been proposed. However the last years, the dramatic development of VLSI has reduced the emphasis on minimizing the number of multiplications and dynamic elements of a signal device and has caused a shift toward considering structures using many parallel devices [1-3] . This is due to the fact that the design does not depend closely on the cost of the processing unit any more. Moreover to the above, the availability of modern processing techniques such as distributed arithmetic, multi-microprocessor systems and array processors permits fast computation of long-independent inner products. Thus, the direction of theoretical research has been changed and is being still changing.

Due to the above reasons a number of efforts has been directed to present block realization structures of one dimensional (1-D) infinite impulse response (IIR) digital filters. The most dominant is that of Burrus [4] , which is based on the difference equation description and results in a matrix representation of convolutions permitting the application of efficient FFT techniques. Later a number of papers on the block realization of two-dimensional (2-D) digital filters have been appeared [9-14] .

In this paper a block state-space realization model is presented in a compact algorithmic form suitable for computer use, which can be split in a number of parallel subfilters. The resulted benefits are high inherent parallelism and significant reduction in the data throughput delay, which are paid by an increase in the total number of nontrivial multiplications. This latter increase appears to happen also in the 1-D block processing structures implemented with subfilters in parallel [8] . Moreover in this paper, the block parallel processing model, which is based on the proposed block state realization model , is presented and an analysis is outlined to achieve a balanced distribution of the necessary nontrivial multiplications in the subfilters. Thus, the number of multiplications in the critical subfilter and therefore the computation time per output sample are reduced. In the sequel, the optimal dimensions of the block rectangle, which lead to the minimization of the multiplications in the critical subfilter, can be determined. Finally the data throughput-delay in the block parallel processing model, being estimated on the basis of the needed multiplications and additions in the critical subfilter is considered. The resulting data throughput-delay is increased almost linearly with the filter's order and is substantially smaller than that in the canonical form.

2. 2-D MODEL OF BLOCK PROCESSING IN THE STATE-SPACE

Consider the linear, time-invariant, multivariable, discrete-time, 2-D system described in state space as follows [15] :

$$\begin{bmatrix} \mathbf{x}^h(i+1,j) \\ \mathbf{x}^v(i,j+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(i,j) \\ \mathbf{x}^v(i,j) \end{bmatrix} + \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} u(i,j) \tag{1a}$$

$$y(i,j) = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(i,j) \\ \mathbf{x}^v(i,j) \end{bmatrix} \tag{1b}$$

or more compactly

$$\mathbf{x}'(i,j) = \mathbf{A}\,\mathbf{x}(i,j) + \mathbf{b}\,u(i,j) \tag{2a}$$

$$y(i,j) = \mathbf{c}\,\mathbf{x}(i,j) \tag{2b}$$

where

$$\mathbf{x}(i,j) = \begin{bmatrix} \mathbf{x}^h(i,j) \\ \mathbf{x}^v(i,j) \end{bmatrix}, \quad \mathbf{x}'(i,j) = \begin{bmatrix} \mathbf{x}^h(i+1,j) \\ \mathbf{x}^v(i,j+1) \end{bmatrix} \tag{3}$$

are the local state vectors, $\mathbf{x}^h \in \mathbb{R}^{n_1}$ is the horizontal state vector, $\mathbf{x}^v \in \mathbb{R}^{n_2}$ is the vertical state vector, i is the integer index of horizontal propagation, j is the integer index of vertical propagation, u is the scalar input, y is the scalar output and **A**,**b**,**c** are constant matrices of appropriate dimensions. Note that the matrices and vectors are denoted by boldface capitals and lowercase letters respectively. For a weak (local) initial condition we assume $\mathbf{x}(0,0) = \xi$ arbitrary and $\mathbf{x}^h(0,j) = \mathbf{x}^v(i,0) = \mathbf{0}, i,j \geq 0$; for a strong (global) initial condition we assume $\mathcal{X}(0,0) = \{\mathbf{x}^h(0,j), \mathbf{x}^v(i,0), i,j \in N\} \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} = \Xi$, where Ξ arbitrary vector [16].

The solution of (1) is the following general response formula [15,16]

$$y(i,j) = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \left[\sum_{k=0}^{j} \mathbf{A}^{i,j-k} \begin{bmatrix} \mathbf{x}^h(0,k) \\ 0 \end{bmatrix} + \sum_{h=0}^{i} \mathbf{A}^{i-h,j} \begin{bmatrix} 0 \\ \mathbf{x}^v(h,0) \end{bmatrix} + \sum_{(0,0)\leq(h,k)<(i,j)} \left[\mathbf{A}^{i-h-1,j-k}\tilde{\mathbf{b}} + \mathbf{A}^{i-h,j-k-1}\tilde{\tilde{\mathbf{b}}} \right] u(h,k) \right] \tag{4}$$

where

$$\tilde{\mathbf{b}}^T = \begin{bmatrix} \mathbf{b}_1^T & \mathbf{0} \end{bmatrix}, \quad \tilde{\tilde{\mathbf{b}}}^T = \begin{bmatrix} \mathbf{0} & \mathbf{b}_2^T \end{bmatrix}$$

and $\mathbf{A}^{i,j}$ is the transition matrix defined in [15] . Using the general responce formula (4), we arrive at the following relation:

$$\mathbf{x}(i+m,j+n) = \sum_{t=1}^{n} \mathbf{A}^{m-1,n-t}\mathbf{A}^{10}\mathbf{x}(i,j+t) + \sum_{s=1}^{m} \mathbf{A}^{m-s,n-1}\mathbf{A}^{01}\mathbf{x}(i+s,j) + \sum_{s=1}^{m} \mathbf{A}^{m-s,n-1}\tilde{\mathbf{b}}u(i+s,j) + \sum_{t=1}^{n} \mathbf{A}^{m-1,n-t}\tilde{\tilde{\mathbf{b}}}u(i,j+t) + \sum_{(1,1)\leq(s,t)\leq(m,n)} \left[\mathbf{A}(m-s,n-t)\mathbf{b} \right] u(i+s,j+t) \tag{5}$$

where the state variables $\mathbf{x}(i,j+t), \mathbf{x}(i+s,j)$, $t = 1,2,\ldots,n$; $s=1,2,\ldots,m$ have been considered as the initial conditions. In (5) A(i,j) denotes the NxN ($N=n_1+n_2$) matrix

$$\mathbf{A}(i,j) = \begin{bmatrix} \mathbf{A}_1^{i-1,j} & \mathbf{A}_2^{i,j-1} \\ \mathbf{A}_3^{i-1,j} & \mathbf{A}_4^{i,j-1} \end{bmatrix} \tag{6}$$

where $\mathbf{A}_1^{i,j}$, $\mathbf{A}_2^{i,j}$, $\mathbf{A}_3^{i,j}$, $\mathbf{A}_4^{i,j}$ are $n_1 x n_1$, $n_1 x n_2$, $n_2 x n_1$, $n_2 x n_2$ submatrices of the two-tuple power of A, $\mathbf{A}^{i,j}$.

Using (6), we construct the following model of block processing, which corresponds to a 2-D filter described by the canonical state-space model (1), as follows:

$$\mathbf{X}'(i,j) = \hat{\mathbf{A}}\mathbf{X}(i,j) + \hat{\mathbf{B}}\mathbf{U}(i,j) \tag{7a}$$

$$\mathbf{Y}(i,j) = \hat{\mathbf{C}}\mathbf{X}(i,j) + \hat{\mathbf{D}}\mathbf{U}(i,j) \tag{7b}$$

where the vectors $\mathbf{X}(i,j)$, $\mathbf{X}'(i,j)$ with dimensions $(k+\ell)Nx1$, $(k+\ell-1)Nx1$ respectively (with $N=n_1+n_2$), are defined as follows:

$$\mathbf{X}(i,j) = \begin{bmatrix} \mathbf{x}_1(i,j) \\ \mathbf{x}_2(i,j) \\ \vdots \\ \mathbf{x}_\ell(i,j) \\ \mathbf{x}_{\ell+1}(i,j) \\ \mathbf{x}_{\ell+2}(i,j) \\ \vdots \\ \mathbf{x}_{k+\ell}(i,j) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(ik,j\ell+1) \\ \mathbf{x}(ik,j\ell+2) \\ \vdots \\ \mathbf{x}(ik,j\ell+\ell) \\ \mathbf{x}(ik+1,j\ell) \\ \mathbf{x}(ik+2,j\ell) \\ \vdots \\ \mathbf{x}(ik+k,j\ell) \end{bmatrix} \tag{8}$$

$$\mathbf{X}'(i,j) = \begin{bmatrix} \mathbf{x}'_1(i,j) \\ \mathbf{x}'_2(i,j) \\ \vdots \\ \mathbf{x}'_{\ell-1}(i,j) \\ \hline \mathbf{x}'_\ell(i,j) \\ \mathbf{x}'_{\ell+1}(i,j) \\ \vdots \\ \mathbf{x}'_{k+\ell-2}(i,j) \\ \hline \mathbf{x}'_{k+\ell-1}(i,j) \end{bmatrix} = \begin{bmatrix} \mathbf{x}(ik+k,j\ell+1) \\ \mathbf{x}(ik+k,j\ell+2) \\ \vdots \\ \mathbf{x}(ik+k,j\ell+\ell-1) \\ \hline \mathbf{x}(ik+1,j\ell+\ell) \\ \mathbf{x}(ik+2,j\ell+\ell) \\ \vdots \\ \mathbf{x}(ik+k-1,j\ell+\ell) \\ \hline \mathbf{x}(ik+k,j\ell+\ell) \end{bmatrix} \tag{9}$$

The input and output vectors $\mathbf{U}(i,j)$, $\mathbf{Y}(i,j)$ with dimensions $(k\ell+k+\ell-1)x1, k\ell x1$ respectively are defined as follows:

$$\mathbf{U}(i,j) = \begin{bmatrix} \mathbf{V}_1(i,j) \\ \mathbf{V}_2(i,j) \\ \hline \vdots \\ \hline \mathbf{V}_{\ell+1}(i,j) \end{bmatrix} = \left[\mathbf{U}(i,j) \right]_T = \left[u(ik+e,j\ell+d) \right] \tag{10a}$$

where

$$(e,d) = \begin{cases} (\tau,0), & \text{for } \tau=1,2,\ldots,k \\ \left(\tau \bmod (k+1), \left\lfloor \frac{\tau}{k+1} \right\rfloor\right), & \tau = k+1,\ldots,k\ell+\ell+k-1 \end{cases} \quad (10b)$$

$$Y(i,j) = \left\lfloor Y(i,j) \right\rfloor_\tau = \left\lfloor y(ik+e,\ell t+d) \right\rfloor \quad (11a)$$

where

$$(e,d) = \left(1+(\tau-1) \bmod k,\ 1+\left\lfloor \frac{\tau-1}{k} \right\rfloor\right) \quad (11b)$$

Note that $\lfloor p/q \rfloor$ denotes the integer part of $p/q \geq 0$. The ordering of the elements $x(ik+e,i\ell+d)$, $u(ik+e,i\ell+d)$, $y(ik+e,i\ell+d)$ in the vectors $X'(i,j)$, $X(i,j)$, $U(i,j)$, $Y(i,j)$ is shown in Figure 1.

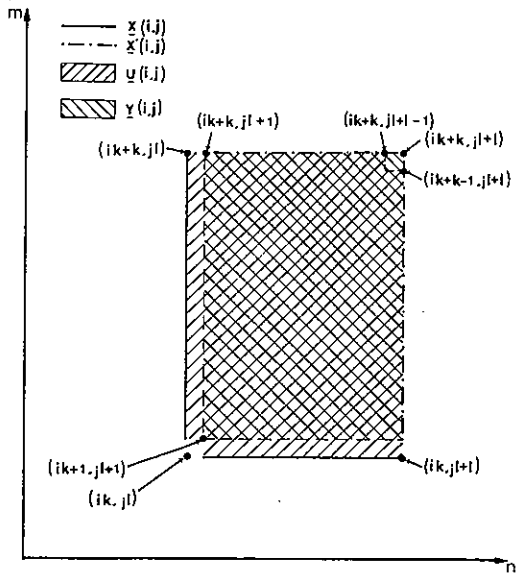


Figure 1

The dimensions k,ℓ of the rectangle, in which the points of each block are ordered, define a rectangular grid on the plane (Figure 2). The state space vectors $X'(i,j)$, $X(i,j)$ are defined only on the lines of the rectangular grid employing the state decimation concept [6,7]. The elements of the vector $Y(i,j)$ corresponding to points in the rectangle (i,j) are properly ordered such that the whole plane is covered by the total of rectangles (Figure 2). The elements of $Y(i,j)$ are computed in terms of the vector $X(i,j)$ as follows:

Each element $y(ik+m,j\ell+n)$ of $Y(i,j)$ is computed using (4) in terms of the elements $x(ik+p,j\ell)$, $p=1,\ldots,m$ and $x(ik,j\ell+q)$, $q=1,\ldots,n$; i.e. of the vector $X(i,j)$.

The matrices $\hat{A}$, $\hat{B}$, $\hat{C}$, $\hat{D}$, appearing in the model (7), result from the application of (5) for the points $(m,n) \in \Delta$ where $\Delta = \{m=k,\ n=1,\ldots,\ell-1\} \cup \{m=1,\ldots,k,\ n=\ell\}$. These matrices are of dimensions $(k+\ell-1)N \times (k+\ell)N$, $(k+\ell-1)N \times (k\ell+k+\ell-1)N$, $k\ell \times (k+\ell)N$, $k\ell \times (k\ell+k+\ell-1)$ respectively, which are given in the sequel. Let the matrix $\hat{A}$ be split as follows:

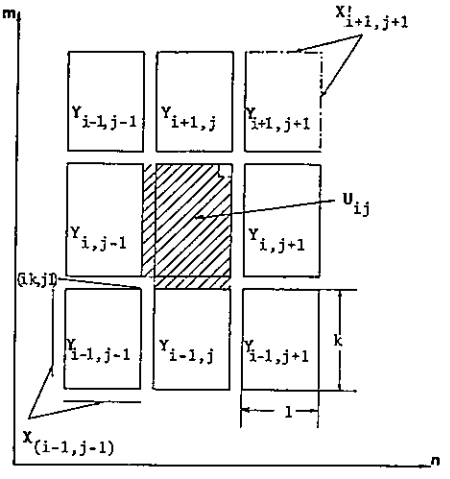


Figure 2

$$\hat{A} = \left[\hat{A}_1\ \hat{A}_2 \ldots \hat{A}_\ell \hat{A}_{\ell+1} \hat{A}_{\ell+2} \ldots \hat{A}_{k+\ell}\right] = \left[\hat{A}_{\mu\nu}\right] \quad (12)$$

where $\hat{A}_\nu$, $\nu=1,2,\ldots,k+\ell$ is the $\lceil (k+\ell-1)N \times N \rceil_\nu$ th submatrix of A. Each matrix $\hat{A}_\nu$ is split in the $N \times N$ submatrices $\hat{A}_{\mu\nu}$, $\mu=1,\ldots,k+\ell-1$ which are given by

$$\hat{A}_{\mu\nu} = \begin{cases} A^{k-1,\mu-\nu}A^{10}, & \text{for } \begin{cases} \mu=1,2,\ldots,\ell-1 \\ \nu=1,2,\ldots,\ell \end{cases} \\ A^{k+\ell-\nu,\mu-1}A^{10}, & \text{for } \begin{cases} \mu=1,2,\ldots,\ell-1 \\ \nu=\ell+1,\ell+2,\ldots,k+\ell \end{cases} \\ A^{\mu-\ell,\ell-\nu}A^{01}, & \text{for } \begin{cases} \mu=\ell,\ell+1,\ldots,k+\ell-1 \\ \nu=1,2,\ldots,\ell \end{cases} \\ A^{\mu-\nu+1,\ell-1}A^{01}, & \text{for } \begin{cases} \mu=\ell,\ell+1,\ldots,k+\ell-1 \\ \nu=\ell+1,\ell+2,\ldots,k+\ell \end{cases} \end{cases} \quad (13)$$

$\hat{B},\hat{C},\hat{D}$ are explicitely given in [17]. They have the form

$$\hat{B} = \left[\hat{B}_1\ \hat{B}_2 \ldots \hat{B}_{\ell+1}\right] = \left[\hat{B}_{\mu\nu}\right] \quad (14)$$

$$\hat{C} = \left[\hat{C}_1 \ldots \hat{C}_\ell \hat{C}_{\ell+1} \ldots \hat{C}_{k+\ell}\right] = \left[\hat{C}_{\mu\nu}\right] \quad (15)$$

$$\hat{D} = \left[\hat{D}_1\ \hat{D}_2 \ldots \hat{D}_{\ell+1}\right] = \left[\hat{d}_{\mu\nu}\right] \quad (16)$$

The matrices $\hat{B}_\nu$, $\hat{D}_\nu$ correspond to the subvectors V_ν, $\nu=1,2,\ldots,\ell+1$ of the input vector U. Hence $\hat{B}_1,\hat{D}_1$, $\hat{B}_{\ell+1}$, $\hat{D}_{\ell+1}$ have k columns each, while $\hat{B}_\nu,\hat{D}_\nu$, $\nu=2,\ldots,\ell$ have $k+1$ columns.

3. MODEL OF BLOCK PARALLEL PROCESSING

From the state-space equations (7) and the form of matrices $\hat{A},\hat{B},\hat{C},\hat{D}$ it is seen that applying the input vector decomposition idea, the ν^{th} elements of $X'(i,j)$, $Y(i,j)$ may be written in the form

$$x'_\nu(i,j) = \sum_{\mu=1}^{k+\ell} \hat{A}_{\mu\nu} x_\mu(i,j) + \sum_{\mu=1}^{\ell+1} (\hat{B}_\mu)_\nu V_\mu(i,j),\ \nu=1,\ldots k+\ell-1 \quad (17a)$$

$$y_\nu(i,j) = \sum_{\mu=1}^{k+\ell} \hat{C}_{\mu\nu} x_\mu(i,j) + \sum_{\mu=1}^{\ell+1} (\hat{D}_\mu)_\nu V_\mu(i,j),\ \nu=1,\ldots,k\ell \quad (17b)$$

where $\hat{A}_{\mu\nu},\hat{C}_{\mu\nu}$ have been determined in (13),(15) and $(\hat{B}_\mu)_\nu$, $(\hat{D}_\mu)_\nu$ denote the ν^{th} rows of $\hat{B}_\mu,\hat{D}_\mu$ respectively while $V_\mu(i,j)$ are considered as in-

dependent input vectors. Now we arrange relations (17a), (17b) in order to obtain $\mathbf{X}'(i,j)$, $\mathbf{Y}(i,j)$ in the left part,as follows:

$$\mathbf{X}'(i,j)=\sum_{\mu=1}^{k+\ell}\mathbf{X}'_{\mu}(i,j)=\sum_{\mu=1}^{k+\ell}\hat{\mathbf{A}}_{\mu}\mathbf{x}_{\mu}(i,j)+\sum_{\mu=1}^{\ell+1}\hat{\mathbf{B}}_{\mu}\mathbf{V}_{\mu}(i,j) \quad (18a)$$

$$\mathbf{Y}(i,j)=\sum_{\mu=1}^{k+\ell}\mathbf{Y}_{\mu}(i,j)=\sum_{\mu=1}^{k+\ell}\hat{\mathbf{C}}_{\mu}\mathbf{x}_{\mu}(i,j)+\sum_{\mu=1}^{\ell+1}\hat{\mathbf{D}}_{\mu}\mathbf{V}_{\mu}(i,j) \quad (18b)$$

We observe from (18) that both $\mathbf{X}'(i,j)$, $\mathbf{Y}(i,j)$ may be formed as the superposition of: (a) $k+\ell$ terms which involve the state variables $\mathbf{x}_{\mu}(i,j)$, $\mu=1,2,\ldots,k+\ell$ and (b) $\ell+1$ terms which involve the input vectors $\mathbf{V}_{\mu}(i,j)$, $\mu=1,2,\ldots,k\ell$.

The idea which leads to the model of block parallel processing is the splitting of (18a,b) in $k+\ell$ subsystems, each one of which produces the partial results $\mathbf{X}'_{\mu}(i,j)$, $\mathbf{Y}_{\mu}(i,j)$, $\mu=1,2,\ldots,k+\ell$, by taking into account only one of the state variables $\mathbf{x}_{\mu}(i,j)$.

The data throughput delay of the block parallel processing model may be approximately defined (the time needed for the interconnections is not considered) by the dimensionless expression

$$TR \triangleq M_c t + A_c \quad (19)$$

where M_c,A_c are the critical number of multiplications and additions in the critical subsystem. After an analysis in order to achieve the most balanced distribution of the computations in the parallel subsystems of (18), it turns out that the TR is almost a linear function of the filter's order N. The reduction of TR is substantial, since the TR in the canonical form depends on N^2 [17].

4. CONCLUSIONS

In this paper a structure for the block implementation of 2-D digital filters is proposed, based on the state-space model. The input and output vectors of the considered block state-space model are ordered in rectagles on the image plane, while the state-space vectors are ordered only on the sides of the rectagles employing the staté decimation concept. The proposed structure is implemented by using many subfilters in parallel. This inherent parallelism leads to a substantial reduction of the data throughput delay. Other figures of merit, such as the throughput rate, finite word- length effects, hardware complexity and cost, should also be taken into account. The presented model is ideally suited for VLSI implementation via systolic and wavefront array processors. The implementation of the present and other associated 2-D block structures via array processors is presently under preparation.

REFERENCES

[1] *Computer*, Special issue, Highly parallel computing, vol. 15, (1982).

[2] Kung, H.T., Special purpose devises for signal and image processing: an opportunity in very large integration (VLSI), *SPIE Real Time Signal Processing*, vol. 241, pp. 76-84, (1980).

[3] Peled, A. and Liu, B., A new hardware realization of digital filters, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 456-462, Dec. (1976).

[4] Burrus, C.S., Block realization of digital filters, *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 230-235, (1972).

[5] Mitra, S.K. and Granasekaran, R., Block implementation of recursive digital filters-new structures and properties, *IEEE Trans. Circuits Syst.*, vol. CAS-25, pp. 200-207, (1978).

[6] Wambergue, C.A. and Roberts, R.A., Block processing structures for fixed-point digital filtering, *Proc. IEEE ICASSP* pp. 498-501, (Paris, France, 1982).

[7] Zeman, J. and Lindgren, A.G., Fast digital filters with low round-off noise, *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 716-723 , (1982).

[8] Nikias, C.L., Fast block data processing via a new IIR digital filter structure,*IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 770-779, (1984).

[9] Azimi-Sadjadi, M.R.,Block implementation of two-dimentional digital filters, *Proc. 1st IEEE Conf. Medical Comp. Science*, pp. 160-169, (1982).

[10] Mitra,S.K. and Gnanasekaran, R., Block implementation of two-dimentional digital filters, *J. Franklin Inst.* vol. 316, pp. 299-316, (1983).

[11] Lee, J.H. and Woods, J.W., Sectioned implementation of two dimensional symmetric half-plane-recursive filters, *IEEE Trans. Acoust., Spech, Signal Processing*, vol. ASSP-33, pp. 419-422, (1985).

[12] Mertzios, B.G., Block realization of 2-D, IIR digital filters, *Signal Processing*, vol. 7, pp. 135-149, (1984).

[13] Mertzios, B.G., Block realization of 2-D half-plane IRR digital filters with matrix convolutions computable in 2-D FFT *Proc. Int. Conf. on Digital Signal Processing*, pp. 59-63, (Florence, Italy, 1984).

[14] Azimi-Sadjadi, M.R., A 2-D block-state realization model, *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 2, pp. 912-922, (1983).

[15] Roesser, R.P., A discrete state-space model for linear image processing, *IEEE Trans. Automat. Contr.*, vol. AC-20, pp. 1-10, Feb. (1975).

[16] Kung, S-Y., Lévy, B.C., Morf, M. and Kailath, Th., New results in 2-D systems theory, Part II: 2-D state-space models,realization and the notions of controllability, observability and minimality, *Proc. IEEE*, vol. 65, pp. 945-961, (1977).

[17] Mertzios, B.G., Block parallel processing of 2-D digital signals, *Circuit, Theory and Appl.* To appear.

Tutorial on TWO-DIMENSIONAL SIGNAL FILTERING

R.M. Mersereau
Georgia Tech.
Digital Processing Laboratory
Atlanta, Georgia
USA

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON A DIRECT APPROACH TO THE REALIZATION OF ONE-DIMENTIONAL AND MULTI-DIMENSIONAL STRUCTURALLY PASSIVE RECURSIVE DIGITAL FILTERS.

S. BASU

Stevens Institute of Technology
Department of Electrical Engineering and Computer Science
Hoboken, New Jersey 07030

Abstract

The problem of structurally passive synthesis of multidimensional digital filters as a cascade interconnection of more elementary building blocks has been addressed via the factorization of the associated discrete lossless two-port transmission matrix. Necessary and sufficient conditions for the factorization to be feasible are obtained. In particular, it is shown that in one-dimension the factorization can always be performed, and as a consequence, known filter structures fall out as special cases of the results developed. Thus, an alternative algorithm for synthesizing one dimensional structurally passive digital filters is also obtained.

1.Introduction:

Various synthesis schemes such as the Darlington synthesis scheme for synthesizing lossless transfer functions as an interconnection of most elementary lossless building blocks such as inductors, capacitors, gyrators etc. in the continuous time domain have now become classical in the network theoretic literature. The corresponding problem in the discrete time domain, namely that of synthesizing a discrete lossless bounded (or positive) transfer function as a structurally passive interconnection of elementary lossless building blocks was first resolved via transformation from prototype problems in the continuous time domain, and the resulting class of filter structures are now known as the wave digital filters [1]. Recently, however, successful attempts to derive these and similar other discrete domain results without making explicit use of tools of classical network theory have been made. Notable among these are the orthogonal filters [2], and the class of filters described in [3], [4] and in related other publications.

In view of interest in the synthesis of multidimensional (k-D) structurally passive digital filters, the problem of synthesis of k-D lossless two-port transfer scattering matrix via the bisection of a prescribed two-port into a cascade connection of two lossless two-port sections of smaller "degree" has been addressed in the continuous time domain in [5]. An attempt to develop a self consistent theory for the synthesis of k-D structurally passive digital filters independent of the continuous time methods have already been initiated in [6] by discussing the discrete domain stability properties of a class of multidimensional polynomials. As a continuation of the above study, the present paper addresses the problem of synthesizing a k-D discrete lossless bounded matrix as the transfer function of a structurally passive two-port digital filter directly in the discrete domain. Our approach is again to bisect the prescribed discrete lossless two-port into a cascade interconnection of two discrete lossless two-ports. Necessary and sufficient conditions as to the feasibility of the bisection is obtained. It falls out that in the one-dimensional (1-D) case the aforementioned bisection is always feasible. Our discussion in the 1-D context thus yields yet another algorithm for the of structurally passive synthesis of 1-D lossless digital filter transfer functions, previously not discussed in the literature.

Supported by Rome Air Development Center, Contract no. F30602-81-C185.

2. Notation, Terminology and Problem Formuation:

Notations such as a, b, c will denote polynomials: $a=a(\underline{z})$, $b=b(\underline{z})$, $c=c(\underline{z})$ in k-variables $\underline{z} = (z_1, z_2, \ldots z_k)$. Notations such as n_{a_i} or $\deg_i a$ will denote the partial degree of a in the variable z_i. The compact notation: $\underline{z}^{\underline{n}_a} \triangleq z_1^{n_{a1}} z_2^{n_{a2}} \ldots z_k^{n_{ak}}$ will also be used. Finally, $\tilde{a} \triangleq a^*(z_1^{*-1}, z_2^{*-1}, \ldots z_k^{*-1})$, $\hat{a} \triangleq \tilde{a}.\underline{z}^{\underline{n}_a}$, where * denotes complex conjugation. Corresponding notations for various polynomials other than the polynomial a will also be used.

A k-D discrete lossless two-port is characterized [6] by an associated transfer function matrix H as in (1) or by a transmission matrix T as in (2).

$$[H]_{11}=b/\hat{a},\quad [H]_{12}=dc\underline{z}^{\underline{n}_a}/\hat{a}, \quad \text{(1a,b)}$$

$$[H]_{21}=c/\hat{a},\quad [H]_{22}=-db\underline{z}^{\underline{n}_a}/\hat{a} \quad \text{(1c,d)}$$

$$[T]_{11}=da/c,\quad [T]_{12}=b/c, \quad \text{(2a,b)}$$

$$[T]_{21}=db\underline{z}^{\underline{n}_a}/c,\quad [T]_{22}=\hat{a}/c \quad \text{(2c,d)}$$

where a,b,c are polynomials such that $\hat{a}$ is scattering Schur [6], $\deg_i b \leq \deg_i a$, $\deg_i c \leq \deg_i a$ for all i=1 to k, d is a unimodular complex

constant i.e., $|d| = 1$ and

$$a\tilde{a} = b\tilde{b} + c\tilde{c}, \qquad (3)$$

Note that (1) can be regarded as a discrete k-D counterpart of Belevitch canonical form for the representation of lossless bounded two-port scattering matrices, well known in classical network theory.

In more specific terms the problem dealt with in the present paper can be described as follows. Given T as in (2), two unimodular complex constants d', d" with $d = d'd"$, and the polynomial factorization $c = c'c"$, along with two sets of integers $\underline{n}'=(n_1',n_2',\ldots n_k')$ and $\underline{n}"=(n_1",n_2",\ldots n_k")$ such that $\deg_i c' \le n_i'$, $\deg_i c" \le n_i"$ and $n_{ai}=n_i'+n_i"$ for all i= to k, we seek a factorization $T = T'T"$, where T' and T" are also discrete lossless two-port transmission matrices with associated polynomials (a',b',c') and (a",b",c") respectively. In addition, the requirements $\deg_i a' \le n_i'$ and $\deg_i a" \le n_i"$ needs to be satisfied. Thus, both T' and T" are also required to have representations similar to those expressed in (2). In particular, the polynomial triples (a',b',c') and (a",b",c") are also required to satisfy the condition that â', â" are scattering Schur, $\deg_i b' \le \deg_i a'$, $\deg_i b" \le \deg_i a"$ for all i=1 to k and (4) holds true.

$$a'\tilde{a}' = b'\tilde{b}'+c'\tilde{c}' \qquad (4a)$$
$$a"\tilde{a}" = b"\tilde{b}" + c"\tilde{c}" \qquad (4b)$$

It then easily follows by considering representations of T' and T" such as that expressed in (2) for T that the condition $T=T'T"$ is equivalent to the conditions expressed in (5a) and (5b) in the following.

$$a = a'a" + d'^{*}\tilde{b}'b"\underline{z}^{\underline{n}_a"} \qquad (5a)$$
$$b = d'a'b" + b'\tilde{a}" \qquad (5b)$$

Definition 2.1: The pair of polynomial two-tuples (a',b') and (a",b") is said to be a solution to the algebraic equation if equations (4) and (5) along with the degree restrictions $\deg_i a' \le n_i'$ and $\deg_i a" \le n_i"$ for i=1 to k are satisfied.
We note that in the above definition the degree restrictions on the polynomials a' and a" are expressed as weak inequalities rather than equalities as is required by the solution to the original problem. Also, the restrictions that the polynomials â' and â" be scattering Schur polynomials are not imposed at all.

Definition 2.2: A polynomial triple (a",b",b') is said to satisfy the fundamental equation if (6) along with (7) holds true.

$$d'.ab" - ba" = -b'c"\tilde{c}"\underline{z}^{\underline{n}"} \qquad (6)$$
$$\deg_i a" \le n_i" \text{ and } \deg_i b' \le n_i' \qquad (7)$$

Note that equation (6) is obtained by eleminating the polynomial a' from (5a,b) and (4b). Obviously then any solution of the algebraic equation also satisfies the fundamental equation.

3. Solution to the Algebraic equation:

Clearly, any solution to the problem of factorization of T into T'T" is also a solution to the algebraic equation. The following theorem shows that any solution to the algebraic equation is also a solution to the problem of factorization of T into T'T".

Theorem 3.1: If the pair of polynomial two-tuples (a',b') and (a",b") constitute a solution to the algebraic equation then the polynomials â' and â" are scattering Schur and $\deg_i a'=n_i'$, $\deg_i a"=n_i"$ for all i = 1 to k.

Proof: Consider the rational function defined as:

$$\psi=(\tilde{a}'\tilde{a}")/\tilde{a}=[(\hat{a}'\hat{a}")/\hat{a}]\underline{z}^{\underline{p}} \qquad (8)$$
where $\underline{p}=(p_1,p_2,\ldots p_k)$,
and $p_i=n_{ai}-(n_{a'i}+n_{a"i})$ (9)

Since â is a scattering Schur polynomial, $n_i'+n_i"=n_{ai}$, and factors of a scattering Schur polynomial are also scattering Schur, the denominator polynomial of ψ is also scattering Schur.

Furthermore, straightforward algebraic manipulation of equations (4b) and (5) yield the following.

$$\psi=(a"/c")(\tilde{a}"/\tilde{c}")[1-d'(\tilde{b}/\tilde{a})(b"/a")] \qquad (10)$$

Since it follows from (3) that $|\tilde{b}/\tilde{a}| \le 1$ and $|b"/a"| \le 1$ for $|z_i|=1$ for i=1 to k, an examination of (10) yields that $Re\psi \ge 0$ for $|z_i|=1$, wherever ψ is well defined. Thus, by invoking a result proved in [6] it follows that ψ is a discrete positive function. Consequently, the numerator polynomial of ψ, in irreducible form, is a widest sense Schur polynomial. This, however, implies that $n_{ai}=n_{a'i}+n_{a"i}$ for all i=1 to k. The last equality along with the facts that $n_i' \ge n_{a'i}$, $n_i" \ge n_{a"i}$ and $n_{ai}=n_i'+n_i"$ together imply that $n_i'=n_{a'i}$ and $n_i"=n_{a"i}$.

The widest sense Schur property of â' has already been established. Next, if for some $\underline{z}_0$ on the distinguished boundary of the polydisc $|z_i| \le 1$, i=1 to k we have $a'(\underline{z}_0)=0$ then from (4a) it follows that $b'(\underline{z}_0)=0$, which in turn due to (5a) imply that $a(\underline{z}_0)=0$. Consequently, if â', and thus a, had a sequentially almost complete set [6] of zeros on the distinguished boundary then a would also have a sequentially almost complete set of zeros there, which is impossible if â scatterinfg Schur. Therefore, â cannot have sequentially almost complete set of zeros on the distinguished boundary. The scattering Schur property of â is thus established in view of results in [5]. Similar arguments hold for â".

A lossless two-port is said to be an allpass if the polynomial b associated with it is identically equal to zero.

Theorem 3.2: Any discrete lossless two-port transmission matrix T can be factored as

$T=T_f T_o T_r$, where T_f, T_o, T_r are also discrete lossless two-port transmission matrices such that T_f and T_r are allpass and if T_o has representation in terms of polynomials a, b, c as in (2) then the polynomial a is relatively prime with b as well as $\breve{b}\underline{z}^{-n_a}$.

In physical terms the above factorization amounts to extraction of discrete lossless two-port sections from the front and rear end of the prescribed transmission matrix. Thus, without loss of generality it will be assumed in all forthcomming discussions that the polynomial a is relatively prime with b as well as with $\breve{b}\underline{z}^{-n_a}$.

4. Solution to the fundamental equation:

Lemma 4.1: If the polynomial a is relatively prime with b as well as $\breve{b}\underline{z}^{-n_a}$ then neither a nor â can have a factor in common with the polynomial $\underline{z}^{-n''}c''\breve{c}''$.

Lemma 4.2: If the polynomial triple $(\alpha'',\beta'',\beta')$ is a solution to the fundamental equation then $\deg_i \beta'' \le n_{\alpha''i}$. Furthermore, there exists a polynomial α' given by (11) such that the polynomial triple $(\beta''\underline{z}^{-n''}, \breve{\alpha}''\underline{z}^{-n''}, -\alpha' d')$ is also a solution to the fundamental equation. Also, we have that $\deg_i \alpha' \le n'_i$ for all i=1 to k.

$$\alpha' = (\breve{\alpha}''c'\breve{c}' + \underline{z}^{-n''}\breve{\beta}b)/\hat{a} \qquad (11)$$

Proof: The fact that $\deg_i \beta'' \le n_{\alpha''i}$ follows directly from the fundamental equation for the triple $(\alpha'',\beta'',\beta')$. Next, by straightforward algebraic manipulations with the fundamental equation for $(\alpha'',\beta'',\beta')$, $c=c'c''$ and (3) yield (12) in the following.

$$\underline{z}^{-n''}(a\breve{\alpha}'' - d^{*}b\beta'') = (\underline{z}^{-n''}c''\breve{c})(\breve{\alpha}''\breve{c}'c' + \underline{z}^{-n''}\breve{\beta}b)\underline{z}^{-n_a}/\hat{a} \qquad (12)$$

Since the left hand side of (12) is a polynomial, due to lemma 4.1, â must divide $(\breve{\alpha}''\breve{c}'c' + \underline{z}^{-n''}\breve{\beta}b)\underline{z}^{-n_a}$. Thus, α' in (11) is a polynomial. The fact that $\deg_i \alpha' \le n'_i$ then follows by considering the degree restrictions on c', c'', α'', b, and β and a.

Lemma 4.3: If the polynomial a is relatively prime with b as well as $\breve{b}\underline{z}^{-n_a}$, and $(\alpha''_1,\beta''_1,\beta'_1)$ and $(\alpha''_2,\beta''_2,\beta'_2)$ are two polynomial triples satisfying the fundamental equation then the rational function given in (13) is a constant.

$$(\alpha''_1\beta''_2 - \beta''_1\alpha''_2)/(\underline{z}^{-n''}c''\breve{c}'') \qquad (13)$$

Proof: By multiplying the fundamental equations for $(\alpha''_1,\beta''_1,\beta'_1)$ and $(\alpha''_2,\beta''_2,\beta'_2)$ respectively by α''_2 and $(-\alpha''_1)$ and adding the resulting equations one obtains equation (15).

$$d'(\alpha''_1\beta''_2 - \alpha''_2\beta''_1) = (\beta'_1\alpha''_2 - \alpha''_1\beta'_2)(\underline{z}^{-n''}c''\breve{c}'')/a \qquad (14)$$

Since the lefthand side of (14) is a polynomial, by invoking lemma 4.1 it then follows that a must divide the polynomial $P=(\beta'_1\alpha''_2 - \alpha''_1\beta'_2)$. Since $\deg_i P \le n'_i + n''_i = n_{ai} = \deg_i a$ for all i=1 to k we have that P/a is a constant. The result then follows by noting that the expression in (13), in view of (14), is equal to (Pd'/a).

Lemma 4.4: If the polynomial a is relatively prime with b as well as $\breve{b}\underline{z}^{-n_a}$, and $(\alpha'', \beta'', \beta')$ is a polynomial triple satisfying the fundamental equation then the expression given in (15) is a constant.

$$(\alpha''\breve{\alpha}'' - \beta''\breve{\beta}'')/c''\breve{c}'' \qquad (15)$$

Proof: Follows from lemma 4.2 and lemma 4.3.

Lemma 4.5: If the polynomial triple $(\alpha'',\beta'',\beta')$ is a solution to the fundamental equation then there exists an α' as given by lemma 4.2 such that $(p\alpha'' + q\underline{z}^{-n''}\breve{\beta}'', p\beta'' + q\underline{z}^{-n''}\breve{\alpha}'', p\beta' - qd\alpha')$ is also a solution to the fundamental equation, where p and q are arbitrary complex numbers.

Proof: Obviously follows from lemma 4.2.

5. Factorization of the discrete lossless two-port transmission matrix:

Two polynomial triples $(\alpha''_1,\beta''_1,\beta'_1)$ and $(\alpha''_2,\beta''_2,\beta'_2)$ each satisfying the fundamenatal equation will be said to be linearly dependent if there exists constants p and q not simultaneously zero such that $p\alpha''_1 + q\alpha''_2 \equiv p\beta''_1 + q\beta''_2 \equiv p\beta'_1 + q\beta'_2 \equiv 0$.

Also, a solution $(\alpha'',\beta'',\beta')$ to fundamental equation will be said to be nonsingular if $\alpha''\breve{\alpha}'' \not\equiv \beta''\breve{\beta}''$.

The following two theorems constitute the major results of this paper.

Theorem 5.1: Assuming that the polynomial a is relatively prime with b as well as $\breve{b}\underline{z}^{-n_a}$, the problem of factorization of discrete lossless two-port transmission matrix T admits a solution if and only if there exists a nonsingular solution $(\alpha'',\beta'',\beta')$ to the fundamental equation.

Proof: Necessity is obvious. If $(\alpha'',\beta'',\beta')$ is a nonsingular solution to the fundamental equation then due to lemma 4.5, $a'' = p\alpha'' + q\underline{z}^{-n''}\breve{\beta}''$, $b'' = p\beta'' + q\underline{z}^{-n''}\breve{\alpha}''$, $b' = p\beta' - qd\alpha'$ is a solution to the fundamental equation. Straightforward algebraic manipulation then yields that

$$(a''\breve{a}'' - b''\breve{b}'')/c''\breve{c}'' = (|p|^2 - |q|^2)K \qquad (16)$$

where

$$K = (\alpha''\breve{\alpha}'' - \beta''\breve{\beta}'')/c''\breve{c}'' \qquad (17)$$

Since due to lemma 4.3 and nonsingularity of $(\alpha'',\beta'',\beta')$, K is a nonzero constant, by proper choice of p and q in the right hand side of (16) it is possible to have $(a''\breve{a}'' - b''\breve{b}'') = c''\breve{c}''$.

Furthermore, there exists a' such that ($\tilde{b}''z^{n''}$, $\tilde{a}''z^{n''}$, $-a'd$), by virtue of lemma 4.2, satisfies the fundamental equation. It can then be verified via routine algebraic manipulation that the pair of two-tuples (a',b') and (a",b") satifies the algebraic equation, and thus, due to theorem 3.1, is a solution to the problem of factorization of T.

Theorem 5.2: Assuming that the polynomial a is relatively prime with b as well as with $\tilde{b}z^{n_a}$, the problem of factorization of discrete lossless two-port transmission matrix T admits a solution if and only if there exists two linearly independent polynomial triples $(\alpha_i'', \beta_i'', \beta_i')$, i=1,2 each of which satisfy the fundamental equation.

Proof: Necessity is obvious. If one of the solutions $(\alpha_i'', \beta_i'', \beta_i')$, i=1,2 is nonsingular then sufficiency follows from theorem 5.1. If both solutions are singular then the triple (a",b",b') obtained as: $a''=p\alpha_1''+q\alpha_2''$, $b''=p\beta_1''+q\beta_2''$, $b'=p\beta_1'+q\beta_2'$, where p and q are complex numbers, satisfies the fundamental equation. Algebraic manipulation then yields that

$$(a''\tilde{a}''+b''\tilde{b}'')/c''\tilde{c}'' = L + \tilde{L} \qquad (18)$$

$$L = p\,q(\alpha_2''\tilde{\alpha}_1''-\beta_2''\tilde{\beta}_1'')/c''\tilde{c}'' \qquad (19)$$

By invoking lemmas 4.2 and 4.3 it then follows that L in (19) is a constant, and thus, $\tilde{L}=L^*$. Furthermore, by following arguments similar to those in [5] it can be proved via the use of results in [7] that $L\neq 0$ if $p\neq 0$ and $q\neq 0$. (The details of this derivation is left out of here for the sake of brevity). Consequently, by proper choice of p and q in (18) and (19) it is possible to have $a''\tilde{a}''-b''\tilde{b}''=c''\tilde{c}''$. The rest of the proof follows by imitating the last paragraph in the proof of theorem 5.1.

6. Discussions and Conclusions:

The fundamental equation (6), when considered as a set of linear simultaneous equations involving the coefficients of the polynomials a", b", b', along with the upper bounds on their degrees, turns out to be overdetermined in general. Therefore, in a generic situation a solution to the problem may not exist. It can be shown, however, that in the one-dimensional case i.e., if k = 1, there are two more unknown coefficients than the number of linear equations in the aforementioned set. Thus, there are at least two linearly independent solutions of the fundamental equation, and in view of theorem 5.2, the problem of factorization of T admits of a solution. Consequently, in the 1-D context structurally passive synthesis for T is achieved by performing a sequence of further factorizations of T' and T" into discrete lossless transmission matrices of progressively lower complexity, until a stage is reached when each of the resulting transmission matrices cannot be factorized any further. It turns out that this latter class of one-dimensional discrete lossless transmission matrices correspond to elementary lossless digital two-port sections previously discussed in the literature [1],[2],[3]. Furthermore, in order for the digital network so syntheized to be 'computable' it may not contain delay free loops arising from cascading of two elementary sections. In spite of the fact that it is known [8] that this problem can always be circumvented by incorporating digital equivalents of unit elements, it is of interest to note that by properly utilizing the flexibility in the choice of p and q in (19) it is always possible to avoid the occurrence of such delay free loops in the filter structure. Finally, it may be remarked that eventhough in the k>1 case synthesis may not be feasible for an arbitrary discrete lossless T, the possibility of synthesis for special classes of discrete lossless T is by no means ruled out. This is especially true in view of synthesizability of certain classes of two-dimensional continuous time systems arising in studies of lumped-distributed netwoks. The class of multidimensional (k>1) discrete lossless two-port transmission matrices T, which admits of such synthesis remains, however, to be identified.

References

[1] A. Fettweis, Digital filter structures related to classical filter networks, AEU, vol.25, pp.79-81 Feb.1971.

[2] P. Dewilde and E. Deprettere, Orthogonal cascade realizations of real multiport digital filters, Int. J. Circuit Th. & Appl., vol.8, pp.245-277, 1980.

[3] P. P. Vaidyanathan and S. K. Mitra, Low passband sensetivity digital filters: A generalized viewpoint and synthesis procedures, IEEE Proc. pp. 404-423, April 1984.

[4] S. K. Rao and T. Kailath, Orthogonal digital filters for VLSI implementation, IEEE Trans. CAS, vol.31,pp.933-945, Nov.1984.

[5] S. Basu and A Fettweis, On the factorization of scattering transfer matrices of lossless multidimensional two-ports, IEEE Trans. CAS, vol.32, pp.925-934, Sept.85.

[6] S. Basu and A. Fettweis, Multidimensional discrete reactance Hurwitz and discrete immittance Hurwitz polynomials, IEEE Int. Symp. on CAS, 1986.

[7] S. Basu and A. Fettweis, On discrete scattering Hurwitz polynomials, Int. J. Circuit Th. & Appl. vol.13, pp.47-59, 1985.

[8] A. Fettwies, Wave digital filters: Theory and practice, Proc. IEEE, vol.74, pp.270-327, February 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

STATE SPACE TECHNIQUES IN STABILIZING TWO-DIMENSIONAL FILTERS

Mauro BISIACCO, Ettore FORNASINI, Giovanni MARCHESINI

Istituto di Elettrotecnica e di Elettronica, Università di Padova
Via Gradenigo 6A
35131 Padova PD, Italy

State and output feedback techniques for stabilizing two-dimensional filters are analysed and compared. Stabilizability depends both on structural properties of state space models and on the existence of non essential singularities of the second kind of transfer functions.

1. INTRODUCTION

Recently several papers have appeared in the literature [1,2], dealing with the synthesis of dynamic feedback compensators whose purpose is to improve the performance of two-dimensional filters and in particular to provide satisfactory stability behaviours.
The stabilization problem exhibits different aspects according to whether we deal with input-output descriptions or internal (state space) representations. As in the 1D case, internal stabilization guarantees external, while in general the viceversa does not hold.
The aim of this paper is to give a short account on state and output feedback stabilizing techniques and to compare these results with those obtainable using transfer function methods.

As known [3], a 2D filter, having transfer function

$$W(z_1,z_2) = \frac{n(z_1,z_2)}{d(z_1,z_2)} \qquad (1)$$

with n and d factor coprime polynomials and $d(0,0) = 1$, is BIBO stable if d is devoid of zeros in the unit closed polydisc

$$\bar{P} = \{(z_1,z_2) \in \mathbb{C} \times \mathbb{C}: |z_1| \leq 1, |z_2| \leq 1\}$$

In [4], Goodman gave some examples of BIBO stable 2D transfer functions whose denominator d vanishes at some points of the distinguished boundary

$$T = \{(z_1,z_2) \in \bar{P} : |z_1| = 1, \quad |z_2| = 1\}$$

However, in these cases, the zeros of d in T are also zeros of n, i.e. they are nonessential singularities of the second kind.
It is worthwhile to point out that the condition that the zeros of d do not belong to $\bar{P}$-T and those in T are zeros of n is not sufficient to guarantee BIBO stability of (1). As far as we know, a characterization of BIBO stable transfer functions exhibiting Goodman's pathology is not yet available.

Internal stability refers to state space models (or 2D systems) given by the following equations [5]

$$\begin{aligned} x(h+1,k+1) &= A_1x(h,k+1)+A_2x(h+1,k)+ \\ &\quad +B_1u(h,k+1)+B_2u(h+1,k) \qquad (2) \\ y(h,k) &= Cx(h,k) \end{aligned}$$

The system (2) is internally stable if, for any initial "global state"

$$\mathcal{X}_o = \{x(i,-i), i \in \mathbb{Z}\}$$

with $\sup \| x(i,-i) \| < \infty$, the free state evolution goes to zero

$$\lim_{h+k \to +\infty} x(h,k) = 0$$

A necessary and sufficient condition for internal stability of system (2) is that the characteristic polynomial

$$\det(I-A_1z_1-A_2z_2)$$

is devoid of zeros in $\bar{P}$.

In the sequel we will be concerned with two different stabilization techniques.

The first one applies to input-output models (1) and is based on dynamic output feedback compensators represented by their transfer functions $r(z_1,z_2)/s(z_1,z_2)$. The overall transfer function is given by

$$\frac{ns}{nr+ds} \tag{3}$$

so that stabilizability essentially depends on the zeros location of the polynomial nr+ds as r and s vary.
The second stabilizing technique consists in constructing compensators in state space form, whose inputs are given by the output or the state variables of system (2).

2. INPUT-OUTPUT AND STATE SPACE STABILIZATION APPROACHES

If we look at the structure of the overall transfer function (3), it is clear that BIBO stabilizability depends on the possibility of choosing r and s so that the variety V(rn+ds) does not intersect the unitary polydisc. As r and s run over the ring of 2D polynomials, the varieties associated with rn+ds have a common intersection, given by the finite set

$$S \triangleq V(n) \cap V(d)$$

There are not further constraints, besides this, on the structure of the variety V(nr+ds), except that it does not cross the origin. Therefore, if $S \cap \bar{P} = \phi$, there exist polynomials r and s that make (3) stable, while if $S \cap (\bar{P}-T) \neq \phi$, the transfer function (3) cannot be stabilized. Due to the fact that a characterization of stable transfer functions having Goodman's pathology is not available, the case when

$$S \cap (P-T) = \phi \quad \text{and} \quad S \cap T = \phi$$

is critical in the sense that we don't know whether BIBO feedback stabiling techniques do exist.

When the stabilization problem is solvable, the procedure for obtaining r and s consists in selecting an appropriate polynomial $\bar{d}$ with the constraints

$$V(\bar{d}) \supset S \quad \text{and} \quad V(\bar{d}) \cap \bar{P} = \phi$$

and solving the Bézout equation

$$nr+ds = \bar{d}^{i} \tag{4}$$

which admits solution for sufficiently large values of i (Hilbert's Nullstellensatz). For the computational aspects involved in solving (4), the reader is referred to [6].

If we assume $\bar{d}=1$, we obtain a dead-beat compensator, that leads to an overall system whose transfer function is polynomial, so that its impulse response is finite. A necessary and sufficient condition for the solvability of (4) with $\bar{d}=1$ is that $S=\phi$, that corresponds to zero coprimeness of n and d.

If we refer now to the state model (2), stabilizability does not depend on the existence of Goodman's pathologies in the transfer function, as it actually does for input-output models. In fact stabilizability is fully characterized on the basis of the structural properties of the state equations, where input-state and state-output maps play an essential role.
More precisely, we are concerned with the rank of the following matrices

$$\left[I-A_1z_1-A_2z_2 \mid B_1z_1+B_2z_2\right] \tag{5}$$

and

$$\begin{bmatrix} I-A_1z_1-A_2z_2 \\ C \end{bmatrix} \tag{6}$$

which constitute the 2D analogue of the matrices appearing in PBH tests of controllability and reconstructibility.

The state feedback dynamic compensators are represented by the following state model

$$\begin{aligned} \bar{x}(h+1,k+1) &= F_1\bar{x}(h,k+1)+F_2\bar{x}(h+1,k)+ \\ &\quad +G_1x(h,k+1)+G_2x(h+1,k) \\ u(h,k) &= H\bar{x}(h,k)+Jx(h,k) \end{aligned} \tag{7}$$

We say that (7) is a stabilizing compensator if the overall system resulting from the connection of (2) and (7) is internally stable.
We have the following Theorem:

Theorem 1 [1] *System (2) can be made internally stable by a state feedback compensator (7) if and only if (5) is full rank for any* (z_1,z_2) *in* $\bar{P}$.

If (5) is full rank in $\mathbb{C}\times\mathbb{C}$, there exists a 2D compensator such that the free state evolution of the resulting overall system goes to zero in a finite number of steps, for any initial global

state of (2) and (7).
From a structural point of view, in this case the rank condition on (5) corresponds to controllability of system (2), that is there exists an input function $u(\cdot,\cdot)$ that forces the state of (2) to go to zero in a finite number of steps. As a consequence of Theorem 1 such an input $u(\cdot,\cdot)$ can be obtained as the output of a multivariable 2D dynamic compensator, driven by the state of (2).

When the output of system (2) is assumed as the input to the compensator, the state equations of the compensator become

$$\begin{aligned}\bar{x}(h+1,k+1) &= F_1\bar{x}(h,k+1)+F_2\bar{x}(h+1,k)+\\ &+G_1y(h,k+1)+G_2y(h+1,k) \qquad (8)\\ u(h,k) &= H\bar{x}(h,k)+Jy(h,k)\end{aligned}$$

Here stabilizability of the overall system depends on the rank of both matrices (5) and (6), as stated in the following Theorem.

Theorem 2 [1] *System (2) can be made internally stable by an output feedback compensator (8) if and only if (5) and (6) are full rank for any* (z_1,z_2) *in* $\bar{P}$.

Similarly to what happens with the state feedback compensator, the existence of a dead-beat output feedback compensator is equivalent to the full rank condition in $\mathbb{C}\times\mathbb{C}$ of (5) and (6).

We shall now analyse the connections between BIBO stabilizability of a transfer function $W(z_1,z_2)$ and internal stabilizability of its realizations, namely of 2D systems which satisfy

$$C(I-A_1z_1-A_2z_2)^{-1}(B_1z_1+B_2z_2) = W(z_1,z_2)$$

Let K and R denote the subsets of $\mathbb{C}\times\mathbb{C}$ where (5) and (6) are not full rank. When K(R) is a finite set, the polynomial matrices $I-A_1z_1-A_2z_2$, $B_1z_1+B_2z_2$ $(I-A_1z_1-A_2z_2,C)$ are left (right) factor coprime. For a proof of this fact, see [7]. A preliminary result in this framework is given by the following Lemma:

Lemma 1 [7] *Let (2) be a realization of the transfer function (1), where n and d are factor coprime. Then* $S \subseteq K\cup R$, $d\,|\det(I-A_1z_1-A_2z_2)$ *and the following facts are equivalent:*

i) $d = \det(I-A_1z_1-A_2z_2)$

ii) $S = K\cup R$

iii) $I-A_1z_1-A_2z_2$, $B_1z_1+B_2z_2$ *are left factor coprime and* $I-A_1z_1-A_2z_2$, C *are right factor coprime.*

For sake of conciseness, it seems convenient to discuss separately the two different situations that arise in the stabilization problem

1. $S\cap\bar{P} = \phi$

In this case, the realizations that satisfy

$$d = \det(I-A_1z_1-A_2z_2)$$

are stabilizable both by state and output feedback. In fact, because of Lemma 1, the sets K and R do not intersect the unit polydisc, so that Theorems 1 and 2 apply.
When we consider realizations where d is a proper factor of $\det(I-A_1z_1-A_2z_2)$, i.e.

$$\det(I-A_1z_1-A_2z_2) = dh \quad , \quad h \neq \text{const}$$

we have

$$K\cup R = S\cup V(h)$$

Consequently output feedback stabilization is possible if and only if $V(h)\cap\bar{P} = \phi$.
If $V(h)\cap\bar{P}\neq\phi$, stabilization can only be achieved by state feedback, provided that $V(h)\cap\bar{P}$ does not intersect K.

2. $S\cap\bar{P} \neq \phi$

Since (5) and/or (6) are not full rank in $\bar{P}$, there are no realizations of $W(z_1,z_2)$ which are stabilizable by output feedback. However those realizations which satisfy

i) $S\cap\bar{P}\cap K = \phi$

ii) $V(h)\cap\bar{P}\cap K = \phi$

are stabilizable by state feedback.
It is interesting to notice that since transfer functions having Goodman's property have non essential singularities of the second kind in T, they satisfy $S\cap\bar{P}\neq\phi$. Whether these transfer functions are BIBO stable or not, every realization is internally unstable and cannot be stabilized by output feedback.

If we look at the same kind of problems in the case of 1D systems, we easily see that some important differences arise due to the fact that 1D transfer functions do not exhibit non essential singularities of the second kind.
Actually any 1D transfer function admits realizations which are stabilizable by dynamic output feedback: it is enough to consider minimal realizations and proceed to complete pole alloca-

tion via output dynamic compensation.

On the other side, consider a transfer function $W(z_1,z_2)$ and let (ξ_1,ξ_2) be in S. Then, for every realization of W at least one of matrices (5) and (6) is not full rank in (ξ_1,ξ_2).
Consequently, the characteristic polynomial of any system obtained by output feedback compensation vanishes in (ξ_1,ξ_2). For this, denote by $\bar{A}_1$ and $\bar{A}_2$ the state matrices of the overall system and assume that (5) is not full rank in (ξ_1,ξ_2). Then there exists a non zero vector v such that

$$v^T\left[I-A_1\xi_1-A_2\xi_2 \mid B_1\xi_1+B_2\xi_2\right] = 0$$

and

$$\left[v^T \mid 0\right]\left[I-\bar{A}_1\xi_1-\bar{A}_2\xi_2\right] = \left[v^T \mid 0\right]\begin{bmatrix} I-(A_1+B_1JC_1)\xi_1 & -B_1H\xi_1-B_2H\xi_2 \\ -G_1C\xi_1-G_2C\xi_2 & I-F_1\xi_1-F_2\xi_2 \end{bmatrix} = 0$$

Similar reasonings apply when (6) is not full rank in (ξ_1,ξ_2).
So, no output feedback stabilization applies to the realizations of transfer functions satisfying $S \cap \bar{P} \neq \phi$. However, state feedback stabilization is feasible when we deal with realizations that satisfy $K \cap P = \phi$.
As shown in [7], it is always possible to construct realizations having this property.

3. CONCLUDING REMARKS

The synthesis of state feedback compensators is based on the solution of the following 2D Bézout matrix equation

$$(B_1z_1+B_2z_2)N(z_1,z_2)+(I-A_1z_1-A_2z_2)M(z_1,z_2) = D(z_1,z_2) \qquad (9)$$

A necessary condition for (9) to be salvable is $\det D \neq 0$ in K and stabilization is possible if $\det D \neq 0$ in $\bar{P}$. Once M and N have been computed, a state equation of the compensator is obtained by constructing any controllable and detectable realization of NM^{-1} [1].
In particular, assuming D=I in (9), the synthesis procedure leads to dead-beat compensators.

One way to construct an output feedback compensator is to connect an asymptotic observer and a state feedback compensator driven by the state estimates. In this case, we need to solve equation (9) and the following Bézout equation [8]

$$P(z_1,z_2)C + Q(z_1,z_2)(I-A_1z_1-A_2z_2) = E(z_1,z_2) \qquad (10)$$

with $\det E \neq 0$ in $\bar{P}$ and $\det E = 0$ in R.
Once we have computed $P(z_1,z_2)$ and $Q(z_1,z_2)$, an asymptotic observer is given by any internally stable realization of

$$\left[Q(B_1z_1+B_2z_2) \mid P\right]$$

In particular, assuming D=I and E=I in (9) and (10), leads to a dead-beat output feedback compensator.

REFERENCES

[1] Bisiacco, M., State and output feedback stabilizability of 2D systems, IEEE Trans. Circuit Syst., vol. CAS-32, pp. 1246-54, Dec. 1985.

[2] Bisiacco, M., Fornasini, E. and Marchesini G., On some Connections between BIBO and internal stability of two-dimensional filters, IEEE Trans. Circuit Syst., vol. CAS-32, pp. 948-953, Sept. 1985.

[3] Shanks, J.L., Treitel, S. and Justice J.H., Stability and synthesis of two-dimensional recursive filters, IEEE Trans. Audio Elect., vol. AU-20, pp. 115-128, June 1972.

[4] Goodman, D., Some stability properties of two dimensional linear shift invariant digital filters, IEEE Trans. Circuit Syst., vol. CAS-24, pp. 201-208, Apr. 1977.

[5] Fornasini, E. and Marchesini G., Stability analysis of 2D systems, IEEE Trans. Circuit Syst., vol. CAS-27, pp. 1210-17, Dec. 1980.

[6] Sebek, M., 2D polynomial equations, Kybernetika, vol. 19, pp. 212-224, 1983.

[7] Bisiacco, M., Fornasini, E. and Marchesini G., Controller design for 2D systems, Proceedings MTNS 85, Stockholm, June 1985.

[8] Bisiacco, M., On the structure of 2D observers to appear in IEEE Trans. Aut. Contr..

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CONVERGENCE PROPERTIES OF 2-D ADAPTIVE GRADIENT LATTICE

H. Youlal, M. Janati-I. and M. Najim

LEESA, Faculté des Sciences, BP. 1014, Rabat, MOROCCO.

In this paper a class of two dimensional adaptive gradient lattice algorithms is developped. Starting with a 2-D lattice (AR) model of a quarter plane causal 2-D system, an adaptive solution based on the gradient techniques for calculating the 2-D lattice parameter reflection factors is derived. The 2-D adaptive gradient lattice algorithm with fixed stepsize matrix which corresponds to the extension of the 1-D lattice LMS algorithm is examined. Using a description in operator form of the lattice structure, convergence properties in the mean squares sense of the algorithm are investigated.

1. INTRODUCTION

In modeling of 2-D signals, several investigators have succeeded in extending potential results of 1-D linear prediction and autoregressive modeling to 2-D case /1,2,3/. Recently a lattice structure for 2-D AR modeling has been proposed /1/. Relationships between the 2-D lattice parameters and the 2-D quarter plane transfer function are also reported.
It is well established that lattice algorithms have superior convergence properties over classic methods, i.e. least squares and stochastic gradient techniques / 4,5 /. It is therefore, tempting to investigate similar approach for 2-D systems.
In this paper a new adaptive gradient lattice algorithm for 2-D systems is developed. This adaptive method consists of calculating the reflection coefficients based on estimates of the prediction errors correlations which are obtained from actual data fields.
The stepsize matrix elements can be either fixed or decreasing. The former case, corresponds to the 2-D extension of the 1-D lattice LMS algorithm.
Using a description in operator form, the convergence properties (in the mean squares sense)of the algorithm are investigated. Such properties include conditions on stepsizes and analytic expression for the misadjustment. In this approach the 2-D lattice structure is considered as the cascade of elementary operators associated with each stage. In the case where the adaptive method is used, these operators become time-varying and it is shown that convergence of the adaptive element is determined by the convergence of the reflection factors parameters defined at each stage.
The paper is organized as follows : In section 2 the 2-D lattice AR model is formulated in the non adaptive case.
In section 3, A 2-D stochastic gradient lattice algorithm is developed. The convergence analysis is presented in section 4. Finally some conclusions are provided.

2. 2-D LATTICE STRUCTURE

The lattice structure of 2-D fields is based upon the following equations :

$$e_{00}^{0,0}(i,j)=e_{10}^{0,0}(i,j)=e_{11}^{0,0}(i,j)=e_{01}^{0,0}(i,j)=y(i,j) \tag{2.1}$$

$$\begin{bmatrix} e_{00}^{m+1,n+1}(i,j) \\ e_{00}^{m+1,n+1}(i,j) \\ e_{11}^{m+1,n+1}(i,j) \\ e_{01}^{m+1,n+1}(i,j) \end{bmatrix} = K^{m+1,n+1} \begin{bmatrix} e_{00}^{m,n}(i,j) \\ e_{10}^{m,n}(i-1,j) \\ e_{11}^{m,n}(i-1,j-1) \\ e_{01}^{m,n}(i,j-1) \end{bmatrix}$$

$$i=1,2,\ldots,I \text{ and } j=1,2,\ldots,J \tag{2.2}$$

Where

$$e_{00}^{m,n}(i,j),\ e_{10}^{m,n}(i,j),\ e_{11}^{m,n}(i,j) \text{ and } e_{01}^{m,n}(i,j)$$

are the first, second, third and fourth quarter plane prediction error fields at stage (m,n).

and

$$K^{m,n}=\begin{bmatrix} 1 & -K_{10}^{m,n} & -K_{11}^{m,n} & -K_{01}^{m,n} \\ -K_{10}^{m,n} & 1 & -K_{01}^{m,n} & -K_{11}^{m,n} \\ -K_{11}^{m,n} & -K_{01}^{m,n} & 1 & -K_{10}^{m,n} \\ -K_{01}^{m,n} & -K_{11}^{m,n} & -K_{10}^{m,n} & 1 \end{bmatrix} \tag{2.3}$$

The $K_{..}$ in (2.3) are the 2-D lattice parameter reflection factors. Starting with the zero order model m=n=0, four prediction error fields are generated. These data fields are then combined linearly to calculate the prediction error field for successively higher order stages. The (m,n) th stage of the basic lattice structure is depicted in fig. 1.

The equations for calculating the lattice parameter reflection factors at each stage can be

derived by minimizing the square value of the prediction error fields with respect to the reflection factors. The mean square error for the (m+1,n+1)th order lattice model is defined as :

$$Q^{m,n}=E\{ e^{m,n}(i,j)^T \; W \; e^{m,n}(i,j)\} \qquad (2.4)$$

Where

$$e^{m,n}(i,j)= \left[e^{m,n}_{00}(\;),e^{m,n}_{10}(\;),e^{m,n}_{11}(\;),e^{m,n}_{01}(\;)\right]^T$$

$$W = \text{diag}\left[w_1,w_2,w_3,w_4 \right] \qquad (2.5)$$

E{.} denotes the expected value over the field of dimension (I-m)x(J-n).
w_1,w_2,w_3,w_4 are arbitrary weights equal to either 0 or 1 associated with the expected values of the prediction error fields e_{00},e_{10}, e_{11} and e_{01} respectively. In the case where $w_1=w_2=w_3=w_4=1$, the total prediction error is minimized.
Substituting (2.1) into (2.4) and minimizing with respect to the reflection factors, the following equation is obtained :

$$R^{m,n}\;\underline{K}^{m+1,n+1} = P^{m,n} \qquad (2.6)$$

where

$$\underline{K}^{m,n} = \left[K^{m,n}_{10},\; K^{m,n}_{11},\; K^{m,n}_{01}\;\right]^T \qquad (2.7)$$

where the entries of the symetric matrix R and the components of the vector P depend on the autocorrelation and cross-correlation terms of the prediction error fields, and of course on the values of the weights in (2.4).
However, if the original data field has four quadrant symetry, then all the criterions produce identical results.
Equations (2.6) allows the calculation of the reflection factor parameters for the (m+1,n+1)th lattice order from the prediction error field data of the (m,n)th stage. In the deterministic case where the autocorrelation and cross-correlation terms are completly known, solution of (2.6) involves the invertion of a 3x3 matrix,for each stage. On the other hand, the entries of R involve the calculation of the prediction error fields correlation, and are carried out by averaging over the known data points. For the case where the aformentioned quantities are not available a priori, solution to (2.6) is not trivial.Therefore, we propose in subsequent section a solution to the 2-D lattice modeling based on adaptive gradient concept.

3. 2-D ADAPTIVE GRADIENT LATTICE ALGORITHM

A deterministic form of a 2-D gradient solution of (2.6) can be written as :

$$\underline{K}^{m,n}(i+1,j+1)=\underline{K}^{m,n}(i+1,j)-G^{m,n}(i+1,j+1)\nabla_{\underline{K}}Q^{m,n} \qquad (3.1)$$

where

$$\nabla_{\underline{K}}Q^{m,n}=R^{m,n}\underline{K}^{m,n}-P^{m,n} \qquad (3.2)$$

R and P are defined as in (2.6). A diagonal matrix G(.,.) is the so-called stepsize of the algorithm at stage (m,n). Practically, the exact gradient is not available and must be estimated or measured from the received data. A simple adaptive gradient implementation of the lattice algorithm results when the unknown gradient is proxied by the gradient of the instantaneous square prediction error

$$\hat{Q}^{m,n}(.,.)=e^{m,n}(.,.)^T W \; e^{m,n}(.,.) \qquad (3.3)$$

which is available at space position (i,j). The noisy gradient can then be used in (3.1) yielding the following stochastic gradient algorithm

$$\underline{K}^{m,n}(i+1,j+1)=\underline{K}^{m,n}(i+1,j)-G^{m,n}(i+1,j+1)\hat{\nabla}_{\underline{K}}Q^{m,n}(i,j) \qquad (3.4)$$

where

$$\hat{\nabla}_{\underline{K}}Q^{m,n}=\hat{R}^{m,n}(i+1,j+1)\underline{K}^{m,n}(i+1,j)-\hat{P}^{m,n}(i+1,j+1) \qquad (3.5)$$

$\hat{R}^{m,n}(.,.)$ and $\hat{P}^{m,n}(.,.)$ are the realizations at the stage (m,n) on the space position (i+1, j+1) of R and P respectivelly, i.e.

$$R^{m,n}=E\{\hat{R}^{m,n}(.,.)\} \text{and } P^{m,n}=E\{\hat{P}^{m,n}(.,.)\}$$

The recursion in(3.4) proceeds for the i=o column, starting with $K(0,0)=K_{00}$, some initial values , and recursively computes K(0,j),0<j<J and then shifting to the next column(s) 0<i<I nnd 1<j<J, with the boundary conditions K(i,0)=K(i-1,J).

The elements of stepsize matrix G can be chosen in different ways, yielding different adaptive algorithms. We examine in the sequel the case where the elements of G are fixed which is similare to the 1-D LMS algorithm.

4. CONVERGENCE PROPERTIES OF THE 2-D LATTICE LMS ALGORITHM

4.1. Mathematical Background

The input-output relationship for the (m,n)th stage (2.2), can be rewritten in operator form as follows :

$$e^{m,n}(i,j)=H^{m,n}e^{m-1,n-1}(i,j)=K^{m,n}Ze^{m-1,n-1}(i,j) \qquad (4.1)$$

where $Z=\text{diag}\left[1,z_1^{-1},z_1^{-1}.\; z_2^{-1},\; z_2^{-1}\right]$ and $K^{m,n}$ as given by (2.3).
$H^{m,n}=K^{m,n}Z$ is the operator associated with the (m,n) stage of the lattice,
It is assumed herein to be an operator on the Hilbert space of zero mean wide sense stationary stochastic data field, with the inner product:

$$<e^{m,nT}(.),e^{m,n}(.)>= E\{e^{m,nT}(.)e^{m,n}(.)\} \qquad (4.2)$$

where E{.} denotes the expected value over the data field. The norm of H induced by the inner product in(4.2) is given by :

$$||H^{m,n}|| = \sup_{||e(.)||=1} ||K^{m,n} Z\, e^{m-1,n-1}(.)|| \quad (4.3)$$

Since Z is a unitary operator, the norm of H reduces to that of K at stage (m,n). However, in the case where the adaptive algorithm (3.4) is used, H becomes a time-varying stochastique operator . Thus, further assumptions to ensure stationarity of the inputs as well as gaussian statistics of the data fields must be introduced:

A1 - The stepsize matrix elements of the adaptive algorithm is sufficiently small such that the inputs to successive stages are at least locally stationary and sufficiently gaussian .

A2 - The stochastic operator elements are uncorrelated with the data fields.

The mean square norm of the stochastic lattice element can be expressed using the Schwarz inequality and the definition of the matrix norm, as :

$$||H^{m,n}||^2 = \sup_{||e(.)||=1} ||K^{m,n} e^{m-1,n-1}||^2$$

$$= \sup_{||e(.)||=1} E\{e^T K^{m,n\,2} e\} \; ||E\, K^{m,n}||^2 \quad (4.4)$$

4.2. Condition on the stepsize

The lattice element H converges to the optimal operator Hopt if :

$$\lim_{i,j\to\infty} ||H_{ij}^{m,n} e^{m-1,n-1}(.) - H_{opt}^{m,n} e^{m-1,n-1}(.)|| = ||(K^{n,n}(i,j) - K_{opt}^{m,n}) Z e^{m-1,n-1}(.)|| \quad (4.5)$$

Under assumptions A1 and A2 and using (4.4) it is is straightforward to see that convergence of (4.5) results from that of the reflection factors Therefore, convergence of the adaptive element H is determined by the convergence of the corresponding reflection parameter factors. In order to derive conditions on the stepsize of the algorithm (3.4),the following stochastic fixed-point theorem is needed.

Theorem 1:

Let F_{ij} be a sequence of random operators on the Hilbert space $\mathcal{H}$ and let $F_{ij} \to F$, where F is a contaction mapping, i.e.

$$\lim_{i,j\to\infty} ||F_{ij} y - Fy|| = 0, \text{ for all } y \text{ in } \mathcal{H}$$

and

$$||Fx - Fy|| < ||x - y||, \text{ for all } x,y \text{ in } \mathcal{H}$$

If F has a fixed point, then the sequence generated by

$$y(i+1,j+1) = F_{i+1,j}\, y(i+1,j)$$

converges strongly to the fixed point of F for every choice of the initial value.

Equation (3.4) can be rewritten to fit to the above stochastic fixed point theorem as follows

$$(\underline{K}^{m,n}(i+1,j+1) - \underline{K}_{opt}^{m,n}) =$$

$$(I - G^{m,n}\hat{R}^{m,n}(i+1,j+1))(\underline{K}^{m,n}(i+1,j) - \underline{K}_{opt}^{m,n}) -$$

$$G^{m,n}(\hat{R}^{m,n}(i+1,j+1)\underline{K}_{opt}^{m,n} - \hat{P}^{m,n}(i+1,j+1)) \quad (4.6)$$

Define the stochastic operator F_{ij} on the space of Gaussian variables :

$$F_{ij}^{m,n} x = (I - G^{m,n}\hat{R}^{m,n}(i+1,j+1))x -$$

$$G^{m,n}(R^{m,n}(i+1,j+1)\underline{K}_{opt}^{m,n} - \hat{P}^{m,n}(i+1,j+1)) \quad (4.7)$$

Then equation (4.6) can be expressed as :

$$(\underline{K}^{m,n}(i+1,j+1) - \underline{K}_{opt}^{m,n}) = F_{i+1,j}^{m,n}(\underline{K}^{m,n}(i+1,j) - \underline{K}_{opt}^{m,n}) \quad (4.8)$$

Using theorem 1, the iteration (4.8) will converge to its unique solution if the stochastic operator F_{ij} is a contraction mapping,i.e.

$$||I - G^{m,n}\hat{R}^{m,n}(i+1,j+1)||^2 < 1 \quad (4.9)$$

Using the schwarz inequality, and since G is a diagonal matrix, the following upper limit on the stepsize elements is obtained /6/:

$$g_s^{m,n} < \frac{2||\hat{R}^{m,n}(i+1,j+1)||}{||\hat{R}^{m,n}(i+1,j+1)^2||} \quad s=1,2,3 \quad (4.10)$$

Notice the formale analogy with the 1-D case. As expected, the upper limit on the stepsize depends on the correlation and cross-correlation terms at each stage of the lattice.

4.3. Misadjustment

Since the gradient is estimated using a finite amount of data, the adaptive algorithm can only result in outputs which are within certain distance from the optimal values, i.e.

$$\lim_{\substack{i\to\infty \\ i\to\infty}} ||H_{ij}^{m,n} e^{m-1,n-1}(.) - H_{opt}^{m,n} e^{m-1,n-1}(.)||^2 = M^{m,n} \quad (4.11)$$

where M is the unormalized misadjustment due to the (m,n)th stage.

With the norm as given by (4.3), equation (4.11) yields:

$$M^{m,n} = \lim_{\substack{i\to\infty \\ j\to\infty}} E\{(\underline{K}^{m,n}(.) - \underline{K}_{opt}^{m,nT}) R^{m,n} (\underline{K}^{m,n}(.) - \underline{K}_{opt}^{m,n})\} \quad (4.12)$$

R is a positive definite matrix, therefore(4.12) is bounded by :

$$E\{(\underline{K}^{m,n}(.) - \underline{K}_{opt}^{m,nT}) R^{m,n} (\underline{K}^{m,n}(.) - \underline{K}_{opt}^{m,n})\}$$

$$< \lambda_{max} ||\underline{K}^{m,n}(.) - \underline{K}_{opt}^{m,n}|| \quad (4.13)$$

where λ_{max} is the maximal eigenvalue of R. The normalized misadjustment due to the (m,n)th stage is defined as the ratio

$$\bar{M}^{m,n} = M^{m,n} / Q_{min} \qquad (4.14)$$

where Q_{min} is the minimal value of the criterion (2.4).
Further manipulation of (4.13-14) yields the following upper bound of the normalized misadjustment /6/,

$$\bar{M}^{m,n} < \lambda_{max} \frac{||G|| \, trR}{2||\tilde{R}|| - ||G|| . ||\tilde{R}\,\tilde{R}^T||} \qquad (4.15)$$

Expression (4.15) shows that the misadjustment is function of the stepsize, and one can expect better convergence for small stepsize.

4.4. Stability analysis

It is well known from 1-D case that stability testing of the lattice model is much simpler than the taped dalay-line structure. In the stability analysis of the 2-D lattice model, simple conditions, on the lattice parameter reflection factors, have been obtained /1/,/6/. which can be checked at each stage as follows

$$|K_{10}| < 1, \quad |K_{01}| < 1, \quad \frac{|K_{11}+K_{01}|}{|1 - K_{10}|} < 1$$

and

$$\frac{|K_{11} - K_{10}|}{|1 + K_{01}|} < \frac{1}{b}$$

where b is a constant which depends in some way on the transfer function amplitude. In the adaptive case the above conditions are checked for every data sample on each stage.

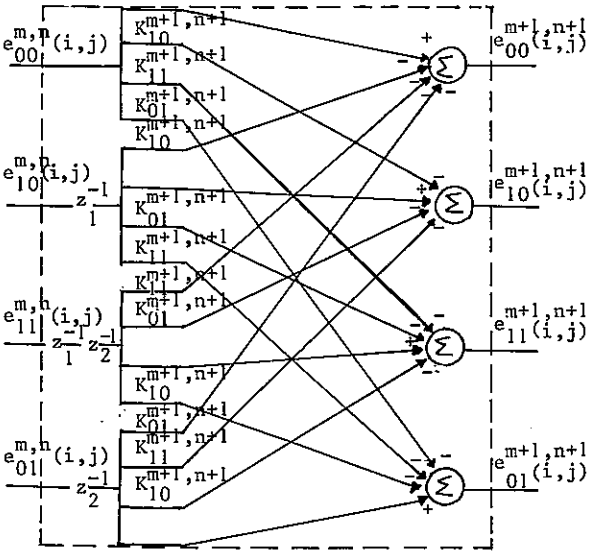


Fig.1. 2-D Lattice stage structure.

CONCLUSION

In this paper we have presented a 2-D adaptive lattice algorithm based on the gradient techniques. Convergence properties of the 2-D LMS lattice algorithm, which corresponds to a fixed stepsize matrix, are covered. Conditions on both stepsize and misadjustment of the algorithm are obtained.
These results shows that better convergence is obtained for small stepsize matrix elements. However one can expect a slower convergence rate. Formal analogy with the 1-D case is pointed out.

REFERENCES

/1/-Parker,S.R. and Kayran, A.H., IEEE Trans. ASSP-32 (1984) p. 872-885

/2/-Maragos, P.A., Shafer R.W. and Mersereau, R.M., IEEE Trans. ASSP-32,(1984)p.1213-1229.

/3/-Marzetta, T.M., IEEE Trans. ASSP-28(1980) p.725-733.

/4/-Cowan C.F.N. and Grant, P.M. (eds.),Adaptive filters (Prentice-Hall Inc. Englewood Cliffs, New Jersey, 1985).

/5/-Sohie, G.R. and Sibul, L.H.,IEEE Trans. ASSP-32 (1984) p.102-107.

/6/-Janati-I.,M. and Youlal,H.,Sur les Algorithmes du Gradient Bidimensionels en Treillis, Report LEESA (1986) (in French).

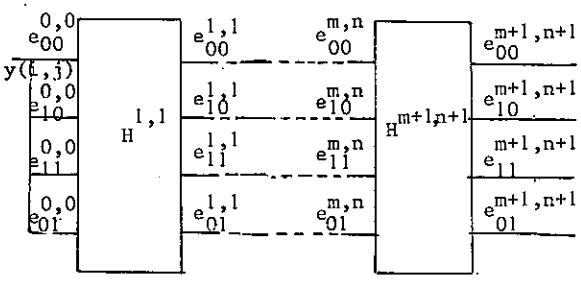


Fig.2. 2-D Lattice representation in operator form.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

THE SUPPORT OF THE CEPSTRUM AND 2-D MINIMUM-PHASE SEQUENCES

Eduard Krajník

Faculty of Electrical Engineering, Czech Technical University
166 27 Prague 6, Czechoslovakia

In this paper we present a necessary and sufficient condition for the existence of the cepstrum for a general support n-D sequence and investigate the support of the cepstrum. Then we state conditions under which a 2-D sequence with the support in a sector of angle less than π or in a half-plane is minimum-phase with respect to the appropriate set.

1. INTRODUCTION

In this paper we consider n-dimensional (n-D) sequences and their cepstra. We are interested in two types of questions; namely the existence of the cepstrum for n-D sequences and conditions under which an n-D sequence is minimum-phase with respect to a semigroup S in Z^n.

The existence question can be roughly stated as follows. Given an n-D absolutely summable sequence, when does it have a cepstrum that is also absolutely summable, and, in addition, if the support of the sequence lies within a semigroup S, when does the support of the cepstrum lie also in S?

The minimum-phase question can be stated as follows. Suppose x is an absolutely summable sequence with the support in a semigroup $S \in Z^n$ having an absolutely summable cepstrum. When do the supports of the inverse and the cepstrum both lie within the semigroup S?

A mathematical framework for studying these questions has been developed in [7] and [8] and enables to give a necessary and sufficient condition for an n-D sequence to have an absolutely summable cepstrum (Theorem 2.4). This condition has previously been stated only for 2-D sequences with rational z-transforms [1].

Then, if a 2-D sequence has its support in a sector of angle less than π or in a half-plane and its cepstrum exists, we show when the support of the cepstrum is contained in the same sector or in the same half-plane respectively (Theorem 3.2). A similar result has been previously proved only for sectors bounded by a coordinate axis [2] or only for finite-extent sequences [3].

Finnaly, for a given absolutely summable 2-D sequence, we investigate relationships between the supports of its inverse and its cepstrum and derive necessary and sufficient conditions for a 2-D sequence to be minimum-phase with respect to a sector of angle less than π and with respect to a half-plane.

Because of limited space of the contribution the proofs are mostly omitted. They can be found in [7] and [8].

2. THE CEPSTRUM

In this section we introduce some notation and state a general existence theorem on the cepstrum.

Let Z^n denote the set of ordered n-tupples of integers. By an n-D sequence we undertand a mapping from Z^n into the set of complex numbers C. The set

$$\text{supp } x = \{k=(k_1,\ldots,k_n) \in Z^n: x(k) \neq 0\}$$

is called the support of x. A set S in Z^n is called a semigroup (under addition) if $0 = (0,\ldots,0)$ is in S and $p \in S$, $q \in S$ implies $p + q \in S$.

Let $l_1(Z^n)$ denote the set of all absolutely summable n-D sequences, i.e.

$$l_1(Z^n) = \{x: Z^n \to C : \sum_{k \in Z^n} |x(k)| < \infty\}$$

For a semigroup S in Z^n let $l_1(S)$ denote the subset of $l_1(Z^n)$ consisting of the sequences whose support is in S. If we define the norm in $l_1(Z^n)$ to be

$$|x| = \sum_{k \in Z^n} |x(k)|$$

and if we define the product of two se-

quences in $l_1(Z^n)$ to be their convolution,

$$(x * y)(k) = \sum_{p+q=k} x(p)\, y(q)$$

then $l_1(Z^n)$ is a Banach algebra with unit δ and $l_1(S)$ is a closed subalgebra of $l_1(Z^n)$ also containing the unit δ.

We say that a sequence x is invertible in $l_1(S)$ if $x \in l_1(S)$ and x has a convolutional inverse in $l_1(S)$, i.e. there exists a sequence $y \in l_1(S)$ such that $x * y = \delta$. Instead of saying x is invertible in $l_1(Z^n)$ we shall simply say x is invertible.

For $k \in Z^n$ we let T_k denote the shift operator in $l_1(Z^n)$. If $y = T_k x$, then for any $p \in Z^n$ we have $y(p) = x(p-k)$ and also

$$y = \delta^k * x, \text{ where}$$

$\delta^k(p) = 1$ for $p=k$ and $\delta^k(p) = 0$ for $p \neq k$, $p \in Z^n$.

With every $x \in l_1(Z^n)$ we can associate its Fourier transform X defined by

$$X(e^{j\omega_1},..,e^{j\omega_n}) = \sum_{k\in Z^n} x(k)\exp j(\omega_1 k_1 + .. + \omega_n k_n)$$

The Fourier transform of a sequence $x \in l_1(Z^n)$ can be considered as a restriction of its z-transform

$$X(z_1,\ldots,z_n) = \sum_{k\in Z^n} x(k) z_1^{k_1} \ldots z_n^{k_n}$$

to the set

$$T^n = \{z=(z_1,\ldots,z_n):\ z_i \in C,\ |z_i|=1\}$$

Theorem 2.1 [6]: A sequence x is invertible if and only if $X(z) \neq 0$ whenever $z \in T^n$.

We associate with every invertible sequence x an element IND $x \in Z^n$, called the *generalized index* of the sequence x and defined as follows. Let γ_i, $i = 1,\ldots,n$ denote the maps of the interval $\langle 0, 2\pi\rangle$ into T^n defined by

$$\gamma_i(\omega) = (1,\ldots,1,e^{j\omega},1,\ldots,1)$$

where $e^{j\omega}$ stands on the i-th place. Then $X\cdot\gamma_i$ is a cloced curve in C (a Nyquist plot) which due to Theorem 2.1 does not pass through the origin. Hence each curve $X\cdot\gamma_i$ has a well-defined index ind $X\cdot\gamma_i$ and we define

$$\text{IND } x = (\text{ind } X\cdot\gamma_1,\ldots,\text{ind } X\cdot\gamma_n)$$

Example: IND $\delta^k = k$ for any $k \in Z^n$ since the z-transform of δ^k is the function

$$\Delta(z) = z^k = z_1^{k_1}\ldots z_n^{k_n}$$

and ind $\Delta\cdot\gamma_i = k_i$.

Lemma 2.2: Let x and y be invertible sequences. Then

$$\text{IND } (x * y) = \text{IND } x + \text{IND } y$$

Let x be a n-D sequence with a z-transform X. Then the cepstrum of x, denoted by $\hat{x}$, is usually defined as the inverse z-transform of $\log X(z)$. In general, however, the sequence x has to be first suitably shifted to ensure the periodicity of $\log X$ in T^n and we therefore propose

Definition 2.3: Let x be an n-D sequence and let there exist a $k \in Z^n$ and an n-D sequence $\hat{x}$ such that

$$\hat{x} = Z^{-1}\{\log Z(T_k x)\} \qquad (1)$$

where Z is the z-transform operator and Z^{-1} its inverse. The sequence $\hat{x}$ is then called the cepstrum of x.

We shall consider only the case that is most important in applications, namely the one when the cepstrum is absolutely summable (stable), i.e. when $\hat{x} \in l_1(Z^n)$. Then x must also be in $l_1(Z^n)$ [7], [8], the z-transform in (1) can be replaced by the Fourier transform and mathematically, the stable cepstrum is equivalent to the logarithm of the suitably shifted sequence x in the Banach algebra $l_1(Z^n)$ [7], [8].

Hereafter, in the interest of brevity, we shall say "cepstrum" when we really mean "stable cepstrum". With this setup we are ready to state the main result of this section.

Theorem 2.4: Suppose $x \in l_1(Z^n)$. Then x has a cepstrum if and only if x is an invertible sequence.

Corollary 2.5: Suppose $x \in l_1(Z^n)$. The cepstrum of x exists if and only if $X(z_1,\ldots,z_n) \neq 0$ in T^n.

Remark. It is known [2] that the shift k in Definition 2.3 ensures the periodicity of $X(z)$ in T^n. The value of k is always uniquely determined and

$$k = -\text{IND } x.$$

3. THE SUPPORT OF THE CEPSTRUM

In this section we shall consider absolutely summable invertible sequences whose support lies in a semigroup S. Theorem 2.4 ensures that such sequences have cepstra; we investigate when the support of the cepstrum lies also in S. Mathematically, this is equivalent to determining conditions under which an element of $1_1(S)$ has a logarithm in $1_1(S)$. The results seem to depend on the structure of the semigroup S and we therefore restrict our attention only to the practically most important cases when S is either a "half-plane" or a "wedge".

We let H denote the semigroup in Z^2 generated by the elements (0,1), (0,-1) and (1,0); thus H contains all the lattice points in the half-plane consisting of the first and fourth quadrants.

Furthemore, let $v_1 = (v_{11},v_{21})$ and $v_2 = (v_{12},v_{22})$ be two vectors in Z^2 with mutually prime coordinates and with

$$d = \det V = \det \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix} > 0$$

If we define

$$W = \{s \in Z^2 : s = b_1 v_1 + b_2 v_2,\ b_i \geq 0,\ \text{reals}\}$$

then W is a semigroup in Z^2 consisting of all lattice points in the sector determined by the vectors v_1 and v_2. The opening of the sector is always less than π. We shall refer to W as the wedge semigroup. The matrix V with the above described properties will be called the wedge matrix. Obviously, there is a one-to-one correspondence between wedge semigroups and wedge matrices. Wedge semigroups are exactly those used in [4] and [5]. In the next section we shall need the following characterization of wedge semigroups.

Lemma 3.1: Let W be a wedge semigroup with a wedge matrix V. Let $d = \det V$, $v_1 = (v_{11},v_{21})$, $v_2 = (v_{12},v_{22})$. Then $s \in W$ if and only if

$$s = (r_1/d)v_1 + (r_2/d)v_2 \qquad (2)$$

where r_1 and r_2 are nonnegative integers and $r = (r_1,r_2)'$ satisfies the congruence equation

$$Vr \equiv 0 \pmod{d}$$

The following theorem gives a characterization of the support of the cepstrum.

Theorem 3.2: Let x be an absolutely summable 2-D sequence with the cepstrum $\hat{x}$. Let S be either a wedge semigroup or the half-plane semigroup in Z^2. Then supp $\hat{x}$ is in S if and only if
a) supp x is in S
b) supp $T_k x$ is in S, where $k = -\text{IND } x$

4. MINIMUM-PHASE SEQUENCES

The computation of a 2-D cepstrum is usually based either on the discrete Fourier transform or on recursion relations which hold when the supports of the sequence, its inverse and its cepstrum lie all in a wedge semigroup W [2]. Such sequences are called in [3] minimum-phase sequences with respect to W. In this section we present necessary and sufficient conditions for a 2-D sequence to be minimum-phase with respect to a wedge semigroup or with respect to the half-plane semigroup.

Definition 4.1: Let x be an n-D absolutely summable sequence and S a semigroup in Z^n. We say that the sequence x is minimum-phase with respect to S if its inverse and cepstrum are both absolutely summable and the supports of x, its inverse and its cepstrum are all in S.

Remark. 1-D minimum-phase sequences with respect to Z_- are usually called maximum-phase, whereas minimum-phase sequences with respect to Z_+ are the classical minimum-phase sequences.

Combining Theorem 3.2 and the results on invertibility in $1_1(S)$ [4, Theorem 4.2] we have

Theorem 4.2: Let S be either a wedge semigroup or the half-plane semigroup in Z^2 and $x \in 1_1(S)$. Then x is minimum-phase with respect to S if and only if
a) $X(z_1,z_2) \neq 0$ whenever $|z_1|=|z_2|=1$
b) $\text{IND } x = 0$

We recall that condition b) means

$$\text{ind } X(e^{j\cdot},1) = 0 \text{ and ind } X(1,e^{j\cdot}) = 0.$$

For the wedge semigroups we also have two simpler criteria.

Theorem 4.3: Let W be a wedge semigroup in Z^2 and $x \in l_1(W)$. Then x is minimum-phase with respect to W if and only if x is invertible in $l_1(W)$.

Finally, we can simplify condition b) of Theorem 4.2 in the way similar to [4, Theorem 4.1]. Let W be a wedge semigroup in Z^2 with the wedge matrix V. Let $d = \det V$. It is easy to verify that the transpose of the matrix dV^{-1} is also a wedge matrix; we denote the corresponding semigroup by W^*. Thus

$$W^* = \{a \in Z^2: a = dbV^{-1},\ b \in Z_+^2\}$$

Since any $s \in W$ satisfies by (2)

$$s' = \frac{1}{d}Vr' \quad \text{for some } r \in Z^2,$$

(the prime denotes transposition), we have for any $s = (s_1, s_2) \in W$ and any $a = (a_1, a_2) \in W^*$

$$a_1s_1+a_2s_2 = a \cdot s' = b \cdot r' = b_1r_1+b_2r_2 \geq 0.$$

Hence W^* characterizes exactly those $a \in Z^2$ used in [4, Theorem 4.1] and from Theorem 4.3 we deduce

Theorem 4.4: Let W be a wedge semigroup in Z^2 and $x \in l_1(W)$. Then x is minimum-phase with respect to W if and only if

a) $X(z_1, z_2) \neq 0$ whenever $|z_1| = |z_2| = 1$

b) ind $X(e^{ja_1 \cdot}, e^{ja_2 \cdot}) = 0$ for some $(a_1, a_2) \in W^*$

5. CONCLUSIONS

Summarizing the results from an engineering viewpoint we see that "$X(z_1, z_2) \neq 0$ whenever $|z_1| = |z_2| = 1$" is a necessary and sufficient condition for the existence of $\hat{x}$. The additional index conditions only ensure that $\hat{x}$ belongs to an appropriate subalgebra of $l_1(Z^2)$, i.e. that supp $\hat{x}$ is contained in a semigroup $S \in Z^2$. This, a little surprisingly, is the same conclusion which has been drawn in [4] about invertibility in $l_1(Z^2)$ and $l_1(S)$.

The criteria for a sequence $x \in l_1(Z^2)$ supported either by a wedge semigroup or by the half-plane semigroup to be minimum-phase with respect to the appropriate semigroup show that for a wedge semigroup they are equivalent to those for invertibility whereas for the half-plane semigroup one more index condition is required in addition to those for invertibility.

REFERENCES

[1] D. E. Dudgeon, "The existence of cepstra for two-dimensional rational polynomials," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-23, pp. 242-243, Apr. 1975.

[2] D. E. Dudgeon, "The computation of two-dimensional cepstra," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-25, pp. 476-484, Dec. 1977

[3] D. E. Dudgeon and R. M. Mersereau, Multidimensional Digital Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[4] K. R. Davidson and M. Vidyasagar, "Causal invertibility and stability of asymmetric half-plane digital filters," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, pp. 195-201, Feb. 1983.

[5] B. T. O'Connor and T. S. Huang, "Stability of general two-dimensional recursive digital filters," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-26, pp. 550-560, Oct 1978.

[6] W. Rudin, Functional Analysis. New York: McGraw-Hill, 1973.

[7] E. Krajník, "Multidimensional cepstral transform," Doctoral thesis, Fac. Elec. Eng., Czech Technical Univ., Prague, 1986 (in Czech).

[8] E. Krajník, The multidimensional cepstrum of general-support sequences," to be published.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOLUTION OF n-D DIFFERENCE EQUATIONS BY THE z-TRANSFORM

Jiří GREGOR

Czech Technical University Prague, Faculty of Electrical Engineering, Dept. of Mathematics
Suchbátarova 2, 166 27 Praha 6, Czechoslovakia

The definition of the z-transform is generalized to the z-transform of n-D sequences and some theorems are proved. Taking into account initial conditions, which for every partial difference equation guarantee the existence of its unique solution, the scope and limitations of the z-transform in finding this solution is investigated.

Input/output relations and state-space models of discrete n-D linear systems are usualy given as linear difference equations in several variables or systems of such equations. The "shift-invariant" cases, i.e., linear partial difference equations with constant coefficients are treated by z-transform methods so as to find their solution or to reveal some of its qualitative properties (such as boundedness, periodicity e.t.c.). In 1-D systems the theory of the z-transform and its applications is well established; as a tool in solving difference equations it is based on the firm ground of existence and uniqueness results for these equations. Although the substantial differences between one- and multi-dimensional cases have been widely recognized, comparatively little attention has been paid to linear partial difference equations in general; commencing with special cases (e.g. 2-D QP filters, ASHP etc. /see [4]/) some basic definitions and results still seem to be contradictory, when generalized.

The formal definition of the z-transform of an n-dimensional sequence f (compare e.g. [3], [4]) given by $\sum_{\alpha \in Z^n} f(\alpha) z^{\alpha}$ where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ runs through all n-tuples of integers Z^n and $z^{\alpha} = z_1^{\alpha_1} z_2^{\alpha_2} \ldots z_n^{\alpha_n}$, seems to be of little use in systems theory, since its application cannot respect given initial conditions; therefore, with this approach, no meaningfull definition of the transfer function can be formulated if not mentioning the difficulties with the notion of linearity. Besides, such z-transform "does not work" even in the 1-D case. On the other hand, some 2-D difference equations are suspected to have no solutions at all, or no "recursively computable" solutions [3], nevertheless their solution by this formal z-transform seems possible. Also the treatment of homogeneous (zero input) difference equations is evidently impossible this way.

For these reasons it seems justified to investigate in more details the application of the z-transform to solution of linear partial difference equation without unnecessary restrictions on dimension, shape of "output mask" e.t.c., but so that existence and uniqueness of solutions as well as a one-to-one correspondence between n-D sequences and their z-transform be guaranteed as far as possible.

In this paper besides the above used notation we shall denote sets of n-tuples by capitals A, B, ..., e.g. $A \subset Z^n$ or $D \subset \mathbb{C}^n$ (n-tuples of complex numbers), or $A = \{ \alpha : \alpha \in Z^n, \alpha_i \geq 0 \}$, greek letters being reserved for elements of Z^n. Addition of n-tuples is as usual, as well as the union and intersection of sets. For $\alpha \in Z^n$, $A \subset Z^n$, we shall write $A + \beta$ for the set $\{\gamma : \gamma = \alpha + \beta, \alpha \in A\}$ and similarly $A + B = \{\gamma : \gamma = \alpha + \beta, \alpha \in A, \beta \in B\}$.

With this notation a linear partial difference equation with constant coefficients is under-

stood to be

(1) $\sum_{\beta \in B} a(\beta) f(\alpha+\beta) = x(\alpha), \quad \alpha \in A \subset Z^n$

where B is a finite set of at least two elements, $B \subset Z^n$ (the "output mask"), the coefficients $a(\beta) \neq 0$ are complex constants for every $\beta \in B$, f and x are mappings, $f : A + B \to \mathbb{C}$, $x : A \to \mathbb{C}$, commonly called sequences. The set of all sequences, assuming finite values on a set $A \subset Z^n$, will be denoted by L_A^n. Evidently, L_A^n is a linear space. The z-transform of sequences from L_A^n is defined as follows :

Definition 1. Let for a sequence $f \in L_A^n$ the series

(2) $\sum_{\alpha \in A} f(\alpha) z^{\alpha}, \quad z \in D$

converge in a relatively complete Reinhardt domain D. Its sum $F(z)$, with $F : D \to \mathbb{C}$, $D \subset \mathbb{C}^n$ is called the z_A-transform of the sequence f. ▯

Definition 2. A nonempty open set $D \subset C^n$ is called a Reinhardt domain (RD) if $z = (z_1, z_2, \ldots, z_n) \in D$ implies $(z_1 e^{j\phi_1}, z_2^{j\phi_2}, \ldots, z_n^{j\phi_n}) \in D$ for every n-tuple $(\phi_1, \phi_2, \ldots, \phi_n) \subset R^n$.

An RD is called complete (CRD), if $z' \in D$ implies $z \in D$ for all z such that $|z_i| < |z_i'|$, $i = 1, 2, \ldots, n$.

An RD is called relatively complete (RCRD) if it is either complete, or all its intersections with the sets $\overset{0}{\mathbb{C}}{}_k^n = \{z : z \in \mathbb{C}^n, z_k = 0\}$ are empty sets, i.e., $D \cap \overset{0}{\mathbb{C}}{}_k^n = \emptyset$, $k = 1,2,\ldots,n$. ▯

The unit disc $U^n = \{z \in \mathbb{C}^n, |z_k| < 1, k = 1,2\ldots,n\}$ is an example of CRD, while the cartesian product $D = U^1 \times A^1$, $A^1 = \{z : z \in \mathbb{C}^1, 0<|z|<1\}$ is a simple example of RCRD, which is not CRD.

The Z_A-transform correspondence, henceforth written as $f_A \leftrightarrow f_D$ (or $f_A(\alpha) \leftrightarrow F_D(z)$), alternatively also as $F_D = Z(f_A)$, includes the sets $A \subset Z^n$ and $D \subset \mathbb{C}^n$. E.g., the linearity of the transform defined by (2) makes sense on linear spaces, therefore $f_A + g_B$ can be considered only as a sequence defined on $A \cap B$ and even in this case the z-transform need not exist if the corresponding RCRD's have an empty intersection.

Nevertheless a number of interesting and useful theorems on this z-transform can be proved. Some of them are simple generalizations of the corresponding 1-D results, some of them have no reasonable counterpart in the 1-D theory. If A in (1) is considered to be the set of nonnegative integers, $A \subset Z_+^1$, we obtain the common 1-D z-transform "in positive powers of z", which might be taken as somewhat unusual. However, except of tradition, there is no reason to prefer "negative powers of z" and, moreover, results on convergence, inverse transform, stability and many others become much more complicated for $n > 1$ due to more involved compactification problem of the set $\mathbb{C}^n$.

Since the z-transform is considered to be a method of solving difference equations and, in addition, to be a reliable tool in stability and realizability investigations of digital systems, we shall concentrate on its scope and limitations with respect to equation (1). For this purpose the most important result would be on z-transform of a "shifted sequence". To this end we introduce the "complete shift operator":

Definition 3. An operator T^{β}, $\beta \in Z^n$ mapping the linear space L_A^n of sequences into itself, and defined by $g = T^{\beta} f$,

$$g(\alpha) = f(\alpha + \beta), \quad \alpha \in A,$$

is called a complete shift operator provided an extension of f to the set $A + \beta$ A is given. ▯

The complete shift operator may not be uniquely determined by the shift β only.

Theorem 4. Let $F_D(z)$ be the z_A-transform of $f \in L_A^n$ and let T^{β} be a complete shift operator $A \cap (A + \beta) \neq \emptyset$. Then

(3) $Z_A(T^{\beta} f) = z^{-\beta}\left(F_D(z) + \sum_{\alpha \in R_{\beta}} f(\alpha) z^{\alpha} - \sum_{\alpha \in S_{\beta}} f(\alpha) z^{\alpha}\right)$,

where $R_{\beta} = (A + \beta) \setminus A$, $S_{\beta} = A \setminus (A + \beta)$. ▯

Proof: The assumptions guarantee, that the sums are well-defined. Hence

$$Z_A(T^{\beta} f) = \sum_{\alpha \in A} f(\alpha+\beta) z^{\alpha} = z^{-\beta} \sum_{\gamma \in A+\beta} f(\gamma) z^{\gamma} =$$

$$= z^{-\beta}\left(\sum_{\gamma \in (A+\beta) \setminus A} + \sum_{\gamma \in A \cap (A+\beta)} \right) =$$

$$= z^{-\beta}\Big(\sum_{\gamma\in R_\beta} f(\gamma)z^\gamma + F_D(z) - \sum_{\gamma\in S_\beta} f(\gamma)z^\gamma\Big).$$

The complete shift operator evidently satisfies the homogenity requirement

$$(\lambda T^\beta)f = T^\beta(\lambda f)$$

for any $\lambda \in \mathbb{C}$. Under certain conditions the sum of two complete shift operators may be again an operator $T : L_A^n \to L_A^n$. This will happen when the two extensions coincide in the respective regions. More generally :

Theorem 5. A finite set of complete shift operators T^β, $\beta \in B \subset Z^n$ is closed under addition, i.e., $T = \sum_{\beta\in B} a(\beta)T^\beta$ maps L_A^n into itself for any complex constants $a(\beta)$, if and only if equation (1) has a unique solution f for any sequence $x \in L_A^n$. ∎

Proof : The main point here is, whether a sequence $f \in L_A^n$ can be extended to points $\alpha + \beta$, $\alpha^- + \beta^-$, $\beta \neq \beta^-$ so that $f(\alpha + \beta) = f(\alpha^- + \beta^-)$ whenever $\alpha + \beta = \alpha^- + \beta^-$; in other words whether there exist such values $f(\alpha + \beta)$, which extend the function f to the region A + B and satisfy the condition

$\sum a_\beta f(\alpha+\beta) = x(\alpha)$ for all $\alpha \in A$ and some $x \in L_A^n$.

But this is equation (1), which, in terms of the theorem might be written also in the form

$$Tf = x .$$

Application of Theorem 4. to the solution of equation (1) seems now to be straightforward provided the uniqueness of its solution can be assumed. Here a recently proved result can be used

Theorem 6. [1] Let in equation (1) the sets A, B be given. Then there exists a set $C \subset A + B$ such that for any coefficients $a(\beta) \neq 0$, $\beta \in B$ and an arbitrary function $g : C \to \mathbb{C}$ equation (1) has exactly one recursively computable solution $f : A + B \to \mathbb{C}$ such that $f(\alpha) = g(\alpha)$ for all $\alpha \in C$. If A is well-ordered with respect to an order relation $\leq$ and $\beta^0 = \max B$, then

$$C = (A + B) \setminus (A + \beta^0) . \qquad (4)$$ ∎

The initial set C (i.e. the set on which (initial) values of the solution f ought to be given) is here explicitly given in a special case only; in the general case its (more complicated) expression can be found in [1].

Summarizing, we have the following problem : When sets $A, B, C \subset Z^n$ with the above described meaning are given, then all equations (1) for arbitrary coefficients $a(\beta)$, arbitrary initial values $g(\alpha)$, $\alpha \in C$ and arbitrary input functions $x \in L_A$ have a unique solution. Does the z-transform of this solution exist? Is it possible to find this z-transform directly from the equation (1) and the given initial values? Surprisingly, the answer to the second question is in the negative even in some cases when the first question has a positive answer. The reason lies in Theorem 4. and can best be illustrated by an example.

Example 7. Solve by the z-transform the equation

$$f(\alpha_1-1,\alpha_2)+f(\alpha_1,\alpha_2-1) = x(\alpha), \qquad (5)$$

$$\alpha \in A_o = \{\alpha_1 \geq 0,\ \alpha_2 \geq 0\}$$

with initial conditions $f(-1, \alpha_2) = 0$ for $\alpha_2 \geq 0$ and with $x(\alpha) = 1$, $\alpha \in A_o$.

We may check that with the lexicographic order all assumptions of Theorem 6. are satisfied : the problem has a unique solution. Let its z-transform be denoted as F. To be able to use Theorem 4. we find the sets S_β, R_β for $\beta_1 = (-1,0)$, $\beta_2 = (0,-1)$ as follows :

$S_{(-1,0)} = S_{(0,-1)} = \emptyset$; $R_{(-1,0)} = C = \{\alpha: \alpha_1 = -1, \alpha_2 \geq 0\}$, $R_{(0,-1)} = \{\alpha: \alpha_1 \geq 0,\ \alpha_2 = -1\}$. Since $Z_A(x) = \dfrac{1}{(1 - z_1)(1 - z_2)}$, we obtain

$$F(z)(z_1 + z_2) = \frac{1}{(1 - z_1)(1 - z_2)} - \phi(z_1),$$

where $\phi(z_1) = \sum_{\alpha\in R_{(0,-1)}} f(\alpha)z_1^{\alpha_1}$ is an unknown function of one variable, which cannot be obtained from the given data and cannot be choosen arbitrarily. Hence, the direct application of the z-transform fails.

Let now the set $A^* = \{\alpha : \alpha_1 \geq 0\}$ (the right half-plane) instead of A_o be considered. Evidently, $A_o \subset A^*$ and with $C^* = \{(-1, \alpha_2), \alpha_2 \in Z\}$ we have also $C \subset C^*$. Take again the originally given equation (5). now with $x(\alpha) = \begin{cases} 1 \text{ for } \alpha \in A_o \\ 0 \text{ for } \alpha \in A^* \setminus A_o \end{cases}$ and $g(\alpha) = 0\ \forall\ \alpha \in A^*$. We find again $S_{(-1,0)} =$

$= S_{(0,-1)} = 0$, $R_{(-1,0)} = C^*$, but now $R_{(0,-1)} = \emptyset$. Therefore Theorem 4 yields

$$F^*(z)(z_1 + z_2) = \frac{1}{(1 - z_1)(1 - z_2)},$$

where F^* is the z_{A^*}-transform of the solution of the modified problem. Using some further results on the n-D z-transform, which in the sake of brevity will not be explained here, from the z_{A^*}-transform of a sequence $f \in L_{A^*}$, the z_A-transform of its restriction to the set $A \subset A^*$ can be obtained. Applying such procedure, we obtain from F^* the correct answer to the originally formulated problem as

$$F(z) = \frac{1}{(1 - z_1^2)(1 - z_2)},$$

which could readily be checked. ∎

In view of this example it would be desirable to delimit the class of equations (1), which can directly be solved by the z-transform, since mainly such equations describe n-D systems with meaningfull transfer functions. Such delimitation should be expressed in terms of the sets A and B independently of the coefficients $a(\beta)$. Due to the specialized version of Theorem 6. given here, we formulate only the specialized result for well-ordered sets A.

Theorem 8. The initial velue problem for equation (1) on a well-ordered set $A \subset Z^n$ can directly (i.e. without additional data) be solved by z_A-transform if and only if

$$\bigcap_{\beta \in B} (A + \beta) = A + \beta^o, \quad \beta^o = \max_{\leq} B . \quad ∎$$

Proof: A necessary and sufficient condition to use Theorem 4. for the solution of (1) consists of the following: for all $\beta \in B$ the sets S_β and R_β must be either empty or included in the initial set C; on the other hand all elements of the initial set C must belong to at least one of the sets S_β or R_β. It can be verified that $\bigcup_{\beta \in B} R_\beta = (A+B) \setminus A$, $\bigcup_{\beta \in B} S_\beta = A \setminus \bigcap_{\beta \in B} (A+\beta)$. With C as in Theorem 6 our condition reads :

$$[(A+B) \setminus A] \cup [A \cap (A+\beta)] = (A+B) \setminus (A+\beta^o),$$

which after a simplification gives the desired result.

This theorem contains as its special case the usually discussed QP filters for n = 2.

All the results presented so far are formulated for general n-D discrete systems. They enable to treat more general 2-D filters than hitherto considered and, at the same time, they contain some warning against haphazard use of the n-D z-transform. Similarly as in applications of other functional transform methods, the n-D z--transform cannot be used in proving the existence or nonexistence of a solution; this must be guaranteed beforehand. In this short overview a number of facts could not be presented; we had to specialize the assumptions. E.g., it can be proved that the assumption on well-ordering in Theorem 8.can be dropped, the indirect method illustrated in Example 7. can be used as a rather general procedure, based on theorems in the n-D z-transform, which were not included here. Theorem 6. remains true when variable coefficients are considered. In a somewhat imprecize language: Any partial difference equation (1) with properly chosen initial conditions has a recursively computable solution and the transfer function of the corresponding system can be defined. In some cases it does not follow from the equation itself, and an indirect procedure to find the transfer function is to be applied.

Similar conclusions hold true for systems of partial difference equations and for the state space description of n-D system, respectively.

REFERENCES

[1] Bosák M. and Gregor J., On Generalized Difference Equations, Apl. Math. /Prague/, in print.
[2] Bose N.K. /ed./, Multidimensional Systems Theory. Progress, Direction and Open Problems in Multidimensional Systems,D. Reidel Publ. Corp.. Dordrecht, 1985.
[3] Dudgeon D.E. and Mercerau R.M.,Multidimensional Digital Signal Processing, Prentice--Hall, INC., New Jersey, 1984.
[4] Huang T.S. /ed./, Two-Dimensional Digital Signal Processing I. Linear Filters, Springer Verlag, New York, 1981.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MODULAR IMPLEMENTATION OF M-D DIGITAL FILTERS USING BIT-SLICED M-D FILTER CHIPS

Vassilis Mertzios* and Anastasios Venetsanopoulos**

*Department of Electrical Engineering
Democritus University of Thrace
Xanthi, Greece

**Department of Electrical Engineering
University of Toronto
Toronto, Canada

1. INTRODUCTION

Recently there has been increasing interest in the use of two-dimensional (2-D) and multidimensional (M-D) digital filters for the processing of sampled M-D data. M-D linear filters find applications in many areas including those of image inhancement, restoration of linear degraded images as well as in the processing of geophysical and biomedical pictures. The current VLSI revolution leads to a reconsideration of the design and implementation criteria in digital signal processing. Current and future VLSI technology makes possible the use of low-cost, very efficient, high-speed special purpose hardware, for the implementation of digital filters. Therefore past considerations leading to moderate savings of dynamic elements and minimization of the quantization noise are being replaced by a new set of criteria, such as parallelism, pipeling, concurrency, modularity, flexibility and reduction of the data throughput-delay [1-3].

This paper proposes some modular structures with great parallelism for the implementation of the M-D digital filters. The used approach is based on the expression of an M-D high order polynomial, which represents the numerator or the denominator of an IIR M-D digital filter, in terms of low order M-D polynomials, which represent FIR M-D filters. Each one of the M-D polynomials is implemented in a modular manner on a chip with the bit-sliced technique. A recently designed chip [4] is used to implement a quarter-plane M-D second order digital filter. Thus M-D filters can be implemented with a standard hardware block, referred as the "Slice". A two-dimensional second order "Slice" was implemented on two chips using 4-micron NMOS VLSI technology [5,6].

Furthermore general realization methods may be used as a basis for the evaluation of structures, which involve the M-D standard chips as building blocks. By using the first, second and third 2-D direct forms [7,8] as well as the decomposition structure [9], it is possible to obtain structures processing 2-D lower order polynomials with fixed coefficients [7].

2. REALIZATION OF A M-D IIR DIGITAL FILTER IN TERMS OF LOW-ORDER M-D FIR FILTERS

Given an M-D rational transfer function:

$$H(Z)=\frac{n(Z)}{d(Z)}=\frac{\sum_{k_1=0}^{N_1}\cdots\sum_{k_m=0}^{N_m} n_{k_1,\ldots,k_m} z_1^{k_1}\ldots z_m^{k_m}}{1+\sum_{k_1=0}^{\hat{N}_1}\cdots\sum_{k_n=0}^{\hat{N}_m} d_{k_1,\ldots,k_m} z_1^{k_1}\ldots z_m^{k_m}}$$

$$k_1+\ldots+k_m\neq 0$$

express it as

$$H(Z)=\Phi\left[a(Z)\right] \tag{2}$$

$$n(Z)=\sum_{(k_1,\ldots,k_m)\in S_1}\sum a_{k_1,\ldots,k_m}(Z)\, z_1^{k_1(s_1+1)}\ldots z_m^{k_m(s_m+1)} \tag{3}$$

where

$$a_{k_1,\ldots,k_m}(Z)=\sum_{\ell_1=0}^{s_1}\cdots\sum_{\ell_m=0}^{s_m} a_{k_1,\ldots,k_m}(\ell_1,\ldots,\ell_m)\cdot z_1^{\ell_1}\ldots z_m^{\ell_m} \tag{4}$$

The set of the points $(k_1,\ldots,k_m)\in S_1$ is determined by

$$S_1=\left\{(k_1,\ldots,k_m)\;\middle|\; k_1=0,1,\ldots,\left[\frac{N_1-s_1}{s_1+1}\right],\left[\frac{N_1}{s_1+1}\right],\ldots,\right.$$

$$\left. k_m=0,1,\ldots,\left[\frac{N_m-s_m}{s_m+1}\right],\left[\frac{N_m}{s_m+1}\right]\right\} \tag{5}$$

and $[q]$ denotes the integer part of the positive number q.

The polynomials $a_{k_1,\ldots,k_m}(Z)$ in (3) are multiplied by powers of $z_1,\ldots,z_m$, which are multiples of $(s_1+1),\ldots,(s_m+1)$ respectively. Note that the maximum power appearing in $n(Z)$ is $z_1^{N_1}\ldots z_m^{N_m}$ while the maximum power appearing in $a_{k_1,\ldots,k_m}(Z)$ is $z_1^{s_1}\ldots z_m^{s_m}$. Equivalently this means that $n(Z)$ involves $(N_1+1)\ldots(N_m+1)$ coefficients $n_{k_1,\ldots,k_m}$, while each polynomial $a_{k_1,\ldots,k_m}(Z)$ involves $(s_1+1)\ldots(s_m+1)$ coefficients $a_{k_1,\ldots,k_m}(\ell_1,\ldots,\ell_m)$. From the above it is seen that the least number of low-order polynomials $a_{k_1,\ldots,k_m}(Z)$, (relatively to the orders of $n(Z)$, $a_{k_1,\ldots,k_m}(Z)$) appears in the case whe-

re $(N_1+1),\ldots,(N_m+1)$ are multiples of $(s_1+1),\ldots,(s_m+1)$ respectively. In this case it holds

$$\left[\frac{N_i-s_i}{s_i+1}\right] = \left[\frac{N_i}{s_i+1}\right] < \left[\frac{N_i+1}{s_i+1}\right] \quad ; \quad i=1,\ldots,m \qquad (6)$$

In the general case where N_i+1 is not a multiple of s_i+1, it holds

$$\left[\frac{N_i-s_i}{s_i+1}\right] < \left[\frac{N_i}{s_i+1}\right] = \left[\frac{N_i+1}{s_i+1}\right] \qquad (7)$$

The number of points being included in S_1 is

$$r_1 = p_1\, p_2 \cdots p_m \qquad (8a)$$

where

$$p_i = \left[\frac{N_i}{s_i+1}\right] + 1 \qquad (8b)$$

From the (3) and (4) it is seen that each coefficient $a_{k_1,\ldots,k_m}(\ell_1,\ldots,\ell_m)$ appearing in (4) is related with the coefficients of the given polynomial n(Z) by

$$a_{k_1,\ldots,k_m}(\ell_1,\ldots,\ell_m) = n_{k_1(s_1+1)+\ell_1,\ldots,k_m(s_m+1)+\ell_m} \qquad (9)$$

It is easily seen by considering (3),(9) that

$$a_{k_1,\ldots,p_i-1,\ldots,k_m}(\ell_1,\ldots,\ell_i,\ldots,\ell_m)=0; \quad i=1,\ldots,m \qquad (10)$$

if one or more of the following relations holds

$$(s_i+1)(p_i-1)+\ell_i > N_i \quad , \quad i=1,2,\ldots,m \qquad (11)$$

and it is nonzero otherwise. Relation (11) holds if N_i+1 is not multiple of s_i+1, i.e., if

$$(N_i+1) \bmod (s_i+1) \neq 0 \qquad (12)$$

The significance of (10) is that a number of the coefficients of certain standard low-order M-D polynomials $a_{k_1,\ldots,k_m}(Z)$ are zero.

The denominator of (1) can be written as

$$d(Z) = \sum_{k_1=0}^{\hat{N}_1}\cdots\sum_{k_m=0}^{\hat{N}_m} d_{k_1,\ldots,k_m} z_1^{k_1}\cdots z_m^{k_m} = 1+\bar{d}(Z) \qquad (13)$$

where $\bar{d}(Z)$ does not contain a constant term. The polynomial $\bar{d}(Z)$ can be expressed, in a way similar to (3) in the form

$$d(Z) = \sum_{(k_1,\ldots,k_m)\in S_2} a_{k_1,\ldots,k_m}(Z)\, z_1^{k_1(s_1+1)}\cdots z_m^{k_m(s_m+1)} \qquad (14)$$

where S_2 is a set of points $(k_1,\ldots,k_m)$, which is defined similarly to S_1.

Now consider the associated to (1) all-pole transfer function

$$H_D(Z) = \frac{1}{1+\bar{d}(Z)} \qquad (15)$$

which can be implemented by a configuration shown in Figure 1.

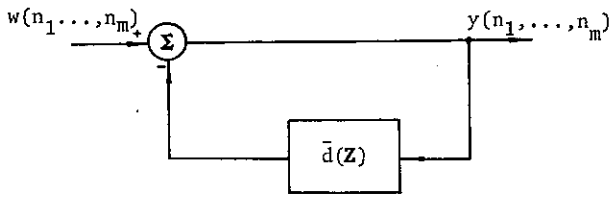


Figure 1

To implement an M-D IIR filter as it is described above, a specific building block a(Z) is needed. The optimum order and form of a(Z), for a specific application, is a matter of current investigation. A second order 2-D filter chip which may be used as the building block for the realization of a high order 2-D filter, was recently designed and implemented [5,6], using the bit-sliced technique.

Various realization methods may be considered as a basis for the evaluation of structures which involve the M-D standard chips as the building block. Specifically the direct realization forms I and II may be used. These realizations are natural extensions of their corresponding 2-D [7,8]. The direct form I results from the cascade configuration of the FIR filter n(Z) with the all-pole filter 1/d(Z)(Fig. 2a). The direct form II results from the cascade configuration of the above two filters in reverse order (Fig. 2b). For the evaluation of both the direct forms I,II the numeration and denominator polynomials are written in the form [6]

$$n(Z) = \sum_{k_m=0}^{N_m}\cdots\left[\sum_{k_1=0}^{N_1} n_{k_1,\ldots,k_m} z_1^{k_m}\right]\cdots z_m^{k_m} \qquad (16)$$

Another class of realizations are those which are based on the m-D decomposition theorem [9]-[12] . According to this latter theorem an arbitrary m-D rational transfer function can be expressed in terms of 1-D factors of the form $(z_i-z_{ij})^{\mu}$, $\mu = 1,-1$, and z_{ij} are constants. Special forms of the decomposition structure are the LU decomposition [11], the Jordan decomposition (JD) [9] , the Walsh-Hadamard decomposition (WHD) [10,12] and the singular value decomposition (SVD) [9] .

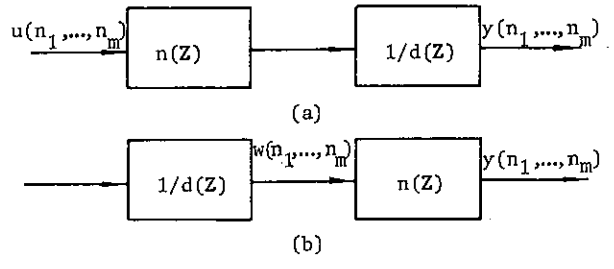


Figure 2

3. IMPLEMETATION OF AN M-D DIGITAL FILTER USING THE M-D FILTER CHIP

Using the expressions (3) and (14) we may write the given M-D transfer function in the form:

$$H(Z)=\frac{\sum_{(k_1,\ldots,k_m)\in S_1} \sum a_{k_1,\ldots,k_m}(Z) z_1^{k_1(s_1+1)} \ldots}{1+\sum_{(k_1,\ldots,k_m)\in S_2} \sum a_{k_1,\ldots,k_m}(Z) z_1^{k_1(s_1+1)} \ldots} \cdot \frac{z_m^{k_m(s_m+1)}}{z_m^{k_m(s_m+1)}} \qquad (16)$$

From the comparison of (1) and (16) it is seen that: (i) the gains of multipliers, i.e. the coefficients $n_{k_1,\ldots,k_m}$, $d_{k_1,\ldots,k_m}$ in (1) have been replaced by the standard polynomials $a_{k_1,\ldots,k_m}(Z)$, (ii) the delays $z_1,\ldots,z_m$ have been replaced by the delays $\hat{z}_1=z^{s_1+1},\ldots\hat{z}_m=z_m^{s_m+1}$ respectively, and (iii) for the implementation of (16) we always need in total $r=r_1+r_2$ standard M-D chips, namely $r_1=p_1\ldots p_m$ and $r_2=q_1,\ldots,q_m$ for the implementation of $n(Z)$ and $d(Z)$ respectively. The parameters q_i are defined similarly to (8b).
It is pointed out that the above hold independently of the realization structure. The main motivation in the use of the expression (16) is the substantial savings in the number of the needed chips.

4. EXAMPLE

From the analysis presented in section 2, it is clear that the proposed technique may be applied to both the IIR and FIR M-D digital filters. Here we will use the example given by Treiter and Shanks [13] (Fig. 3) which considers a 2-D planar filter that acts on the gridded potential field of one level and yields as output the field that would be measured at a distance h above the original recording plane. The impulse response prototype is given by

$$h_{ij} = \frac{1}{2}\,\frac{1.5}{(h^2+i^2+j^2)^{3/2}} \qquad (17)$$

where

$h = 1.5$

$i,j = -10, -9,\ldots,0,\ldots,9,10$

The size of $H=[h_{ij}]$ is $N_1 \times N_2 = 21 \times 21$ and the rank of the coefficient matrix of filter is $\rho=11$. Therefore eleven stages are adequate to fully realize the filter by applying the decomposition realization structures [9-12], [14].

At first we will consider the direct form realization (direct forms I and II coincide for the FIR filters) and the general decomposition structure [9] of the filter (17). In the sequel we will consider the impact of the use of a 2-D second order filter chip as a building block for the realization of (17) by the same structures. We will use as figures of merit the number of relays z_1, z_2, $\hat{z}_1=z_1^{s_1+1}=z_1^3$, $\hat{z}_2=z_2^{s_2+1}=z_2^3$ and the number of 1-D and second-order 2-D chips used in each case.

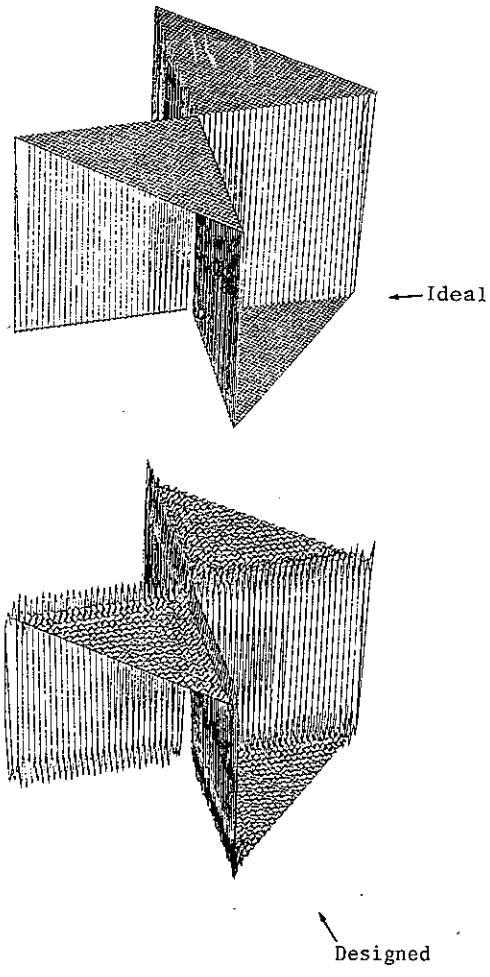


Figure 3

A) Direct form.

We need [5]

$N_D=N_1(N_2+1)$ elements $(z_1)+N_1$ elements (z_2)

or

$$\mu = (N_2+1)\left[\frac{N_1+1}{2}\right] = 242$$

second order 1-D chips in the place of the $N_1(N_2+1)=462$ elements (z_1).

Consider now the realization of the filter expressed in the form of (16); we need

$$r_1=p_1p_2=\left(\left[\frac{N_1}{s_1+1}\right]+1\right)\left(\left[\frac{N_2}{s_2+1}\right]+1\right) = 8\cdot 8 = 64$$

second order 2-D chips, in addition to a number

of delays $\hat{z}_1, \hat{z}_2$ given by

$\hat{N}_D = p_2(p_1-1)$ elements $(\hat{z}_1) + (p_2-1)$ elements $(\hat{z}_2)$
$= 56$ elemnts $(\hat{z}_1) + 7$ elements $(\hat{z}_2)$

B) Decomposition structure

We need just [5]

$$\mu = \left\lceil \frac{N_1+1}{2} \right\rceil + \left\lceil \frac{N_2+1}{2} \right\rceil = 11(11+11)=242$$

second order 1-D chips.

The decomposition structure of the filter in the form (16) also requires

$$r_1 = p_1 p_2 = 64$$

second order 2-D chips in addition to

$$\hat{\mu} = \min\{p_1, p_2\} \left\lceil \frac{p_1}{2} \right\rceil = 8 \cdot 4 = 32$$

second order 1-D chips and

$\hat{N}_D = \min\{p_1, p_2\}\ (p_2-1) = 8 \cdot 7 = 56$ elements $(\hat{z}_2)$.

5. CONCLUSIONS

A technique has been presented for the realization of an arbitrary M-D polynomial or rational transfer function (which represent FIR and IIR filters respectively) in terms of low-order M-D polynomials. These latter M-D polynomials may be implemented as FIR filters which are used as building blocks for the implementation of the general M-D filter. The proposed approach can be applied to any realization structure and leads to substantial savings in the number of chips and delays needed for the regular realizations. The application of this method for an advantageous modular block realization of M-D filters is presently under preparation.

REFERENCES

[1] Highly parallel computing, *Computer*, vol. 15, (1982).

[2] Kung, S.Y. and Annevalink, J., VLSI design for massively parallel signal processors, *Microprocessors and Microsystems*, vol. 7, no. 4, pp. 461-468, (1983).

[3] Peled, A. and Liu, B., A new hardware realization of digital filters, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 456-462, 1974.

[4] Jaggernauth, H., Loui, A.C.P. and Venetsanopoulos, A.N., Real-time image processing by distributed arithmetic implementation of two-dimensional digital filters, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1546-1555, (1985).

[5] Mertzios, B.G. and Venetzanopoulos, A.N., VLSI implementation of two-dimensional digital filters via two-dimensional filter chips, *IEEE Trans. Circuits Syst.*, vol.CAS-33, pp. 239-249, (1986).

[6] Lee, S. and Venetsanopoulos, A.N., A two-dimensional digital filter chip set for modular two-dimensional filter implementation, *in Proc. ICASSP, Tampa*, FL. pp. 26.8.1-26.8.4,(1985).

[7] Dudgeon, D.E. and Mersereau, R.M., *Multidimensional Digital Signal Processing*. Englewood Chiffs, NJ: Prentice-Hall, (1984).

[8] Bose, N.K., *Applied Multidimensional Systems Theory*. New York: Van Nostrand Reinhold, (1982).

[9] Venetsanopoulos,A.N. and Mertzios, B.G., A decomposition theorem and its implications to the design and realization of two-dimensional filters, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1562-1575, (1985).

[10] Mertzios, B.G. and Venetsanopoulos, A.N., Modular realization of multi-dimensional filters, *Signal Processing*, vol. 7, pp. 351-369, (1984).

[11] Nikias, C.L., Chrysafis, A.P. and Venetsanopoulos, A.N., The LU decomposition theorem and its implications to the realization of two-dimensional digital filters, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 694-711, (1985).

[12] Pitas, I.K. and Venetsanopoulos, A.N., Two-dimensional realization of digital filters by transform decomposition, *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 1029-1040, (1985).

[13] Treitel, S. amd Shanks, J.L., The design of multistage separable planar filters, *IEEE Trans. Geosci Electron*, vol. 12, pp. 242-244, (1976).

[14] Venetsanopoulos, A.N. and Nikias, C.L., Design and realization of multidimensional digital filters via matrix decomposition approaches, in *Advances in Geophysical Data Processing*, vol. 2, pp. 263-305, JAI Press Inc., (1985).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

2-D DIGITAL l_1-PSEUDOPASSIVE FILTERS

Marek DOMAŃSKI

Poznań Technical University,
Institute of Electronics and Telecommunication,
ul. Piotrowo 3a, 60-965 Poznań, Poland+

Both passivity of an analog network and classical pseudopassivity of a digital filter are related to the l2-norm of certain signals. Unfortunately, pseudopassivity of a digital filter, e.g., of a wave filter, does not guarantee BIBO-stability in the multidimensional case. That is just why we propose to design l1-pseudopassive two-dimensional recursive digital filters which are inherently BIBO-stable. The proposed structures of such filters are particularly useful for lowpass filtering of digital images. The theory developed for filter synthesis, enables us also to state a sufficient condition of BIBO-stability of digital networks.

1. INTRODUCTION

Many structures of digital filters derived from doubly-loaded lossless two-ports have been proposed during the last 15 years [1]. Most of such filters exhibit a property called "pseudopassivity" [2] which is related to the weighted l2-norm of the state-vector and the signals at the input and the output of the filter. Pseudopassivity of a 1-D digital filter implies BIBO-stability and suppression of parasitic oscillations, and under certain additional conditions, small sensitivity with respect to the filter coefficients [1],[3]-[7]. In pseudopassive filters, small roundoff noise at the output is also reported in the references [3],[7]-[9].

Generalization of pseudopassivity for multidimensional networks is straightforward [10], and many pseudopassive digital filters have been already considered in the references [1]. Moreover, some very useful properties of 1-D pseudopassive digital filters are proved to be preserved also in the multidimensional case [7][11][12][13]. Unfortunately, for the multidimensional filters, hitherto proposed pseudopassivity that is related to the l2-norm of the state vector does not guarantee BIBO-stability. Such a phenomenon in 2-D filters is related to the well-known effects of the nonessential singularities of the second kind at the boundary of the unit bidisc [14][15].

The aim of this paper is to propose two-dimensional (2-D) digital filter structures with guaranteed BIBO-stability and small sensitivity in the passbands. As a tool for synthesis, we use pseudopassivity related to the l1-norm of the state-vector.

2. l_p-PSEUDOPASSIVITY OF 2-D DIGITAL FILTERS

2.1. A general model of 2-D filters

A general structure of a spatially-invariant two-dimensional digital filter is shown in Fig. 1.

A filter consists of delay-elements and a delay-free digital network that can be considered as a digital (N+2)-port. So, the inputs and the outputs of the delay-free network are ordered in pairs. Usually, the additional input p and the additional output q (denoted by dashed lines in Fig. 1) are not realized in the network. They are introduced only to simplify the considerations on pseudopassivity.

The structure from Fig. 1 can be described by the local state-space model [16][17] with the $(N \times 1)$ state vector $\underline{x}$.

+The postal address: Politechnika Poznańska, Instytut Elektroniki i Telekomunikacji, ul. Piotrowo 3a, 60-965 Poznań, Poland.

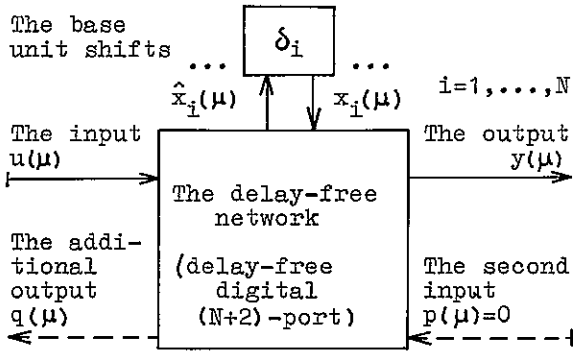


Fig. 1. The general structure of a spatially-variant filter.

We denote

μ - a 2-tuple of integers,

$\underline{x}(\mu)$ - the value of the state vector measured at a point μ ,

$\underline{x}(\mu) = [x_i(\mu)]$, $\hat{\underline{x}}(\mu) = [\hat{x}_i(\mu)]$,

$\hat{x}_i(\mu) = x_i(\mu+\delta_i)$, , i=1,...,N,

δ_i - a base forward shift, i=1,...,N.

Let us assume linearity of the system. Then, rearranging the state and output equations, we obtain

$$\hat{\underline{v}}(\mu) = \underline{S}\,\underline{v}(\mu) \qquad (1)$$

where

$\underline{v}$ - the augmented state vector,

$$\underline{v}(\mu) = \begin{bmatrix} u(\mu) \\ p(\mu) \\ x_1(\mu) \\ \vdots \\ x_N(\mu) \end{bmatrix}, \quad \hat{\underline{v}}(\mu) = \begin{bmatrix} q(\mu) \\ y(\mu) \\ \hat{x}_1(\mu) \\ \vdots \\ \hat{x}_N(\mu) \end{bmatrix},$$

$\underline{S}$ - the augmented state matrix which is the transfer matrix of the delay-free (N+2)-port.

At last, we assume that for the considered systems the signals $\underline{v}(\mu)$ and the corresponding $\hat{\underline{v}}(\mu)$ are members of a certain class of admissible signals. Such a class will be denoted as D.

2.2. Generalized pseudopassivity and pseudolosslessness

Let us introduce the norms $\|\underline{v}\|$ and $\|\hat{\underline{v}}\|$ of the vectors $\underline{v}$ and $\hat{\underline{v}}$, respectively. Thereby, we are able to define generalized pseudopower [18] absorbed in the delay-free network at a point μ as

$$p(\mu) = \|\underline{v}(\mu)\| - \|\hat{\underline{v}}(\mu)\| \ . \qquad (2)$$

Definition 1: The filter is pseudopassive (pseudolossless) if there is $p(\mu) \geqslant 0$ ($p(\mu)=0$) for all $\underline{v}(\mu) \in D$ and all μ.

So, pseudopassivity is related to a certain norm of the augmented state vector. We consider only the lp-norm

$$\|\underline{v}(\mu)\| = \sum_{i=1}^{N+2} g_i(\mu)\,|v_i(\mu)|^p, \qquad (3)$$

where $g_i > 0$ - the coefficient of the i-th port,

p - a positive integer.

Pseudopassivity related to the lp-norm will be called lp-pseudopassivity.

Proposition 1: A lp-pseudopassive digital filter is lp-stable [18], i.e.,

$$\sum_{\mu \in Z^2} |h(\mu)|^p < \infty, \qquad (4)$$

where Z^2 denotes the set of all 2-tuples of integers and $h(\mu)$ is the response to the unit impulse $\delta(\mu)$.

Proof:

Pseudopassivity yields

$$L = \sum_{\mu \in Z^2} \|u(\mu)\| \geqslant \sum_{\mu \in Z^2} \|y(\mu)\| = R \ .$$

But $u(\mu) = \delta(\mu)$ and $y(\mu) = h(\mu)$, and

$$L = g_1, \quad R = \sum_{\mu \in Z^2} g_2 |h(\mu)|^p.$$

So, $$\sum_{\mu \in Z^2} |h(\mu)|^p \leqslant g_1/g_2 < \infty \ .$$

Q.E.D.

The greater values of p are related to the weaker kinds of stability. So, it is interested to consider 2-D l_1-pseudopassive filters because they are l_1-stable, i.e., BIBO-stable [19] .

Note that the above considerations remain valid also for multidimensional filters, both linear and nonlinear. Nevertheless, we restrict further considerations to 2-D linear digital filters (both quadrantal and half-plane).

3. PROPERTIES OF l_1-PSEUDOPASSIVE 2-D DIGITAL FILTERS

Definition 2: A digital filter is structurally pseudopassive (structurally pseudolossless) in D if it is pseudopassive (pseudolossless) and there exists a neighbourhood of the nominal values of the filter coefficients such that the filter remains pseudopassive (pseudolossless) in D also for coefficient values from this neighbourhood.

Definition 3: A structurally pseudopassive digital filter is strictly structurally (s.s.) pseudopassive in D with respect to the given method of data rounding/truncation if for all $\underline{v} \in D$ we have

$p(\mu) \geqslant 0$ for each $\mu \in Z^2$
also for rounded/truncated coefficient values and arithmetic operation results.

Proposition 2: In a 2-D digital filter that is s.s. l1-pseudopassive with respect to a given method of data rounding/truncation, zero-input parasitic oscillations are suppressed at the output of the filter.

The proof utilizes a similar idea as that given in [20].

Proposition 3: The magnitude characteristic of a l1-pseudopassive filter is bounded, i.e.,

$$|H(e^{-j\omega_1}, e^{-j\omega_2})| \leqslant g_1/g_2 \qquad (5)$$

where $H(z_1,z_2)$ - the transfer function,
g_1, g_2 - the input and output port coefficients, respectively.

The proof is omitted for the sake of brevity.

If the upper bound in (5) is reached for a 2-D frequency

$\underline{\omega}_o = (\omega_{1o}, \omega_{2o})$,

the frequency is called the attenuation zero.

Proposition 4: A structurally l1-pseudopassive 2-D digital filter exhibits, in its attenuation zeros, zero magnitude sensitivity with respect to the coefficients that are independent on values of the port coefficients g_1 and g_2.

The proposition can be easily proved using the argumentation similar to that given by Orchard in [21].

At last, we should conclude that inherent BIBO-stability is the main advantage of 2-D l1-pseudopassive filters over l2-pseudopassive filters. The other basic properties of both classes of structures are similar.

4. STRUCTURES OF l_1-PSEUDOPASSIVE FILTERS

4.1. General remarks

Pseudopassivity of a linear filter is defined by the properties of the transfer matrix $\underline{S}$ of the delay-free (N+2)-port (see Fig. 1). We are going to consider those properties for two classes of admissible signals:
DR - the set of all signals with finite real values,
D+ - the set of all signals with finite positive real values.

4.2. Structures that are l1-pseudopassive in DR

The condition for l1-pseudopassivity is as follows

$$\sum_{i=1}^{N+2} g_i |s_{ij}| \leqslant g_j \ , \ j=1,\ldots,N+2 \qquad (6)$$

where s_{ij} - elements of the matrix $\underline{S}$ $(1 \leqslant i,j \leqslant N+2)$.

Unfortunately, there is no analogous condition for l1-pseudolosslessness.

So, we propose the structure that consists of simple blocks which are digital three-ports. In order to avoid delay-free loops, we have to assume that one port in a three-port is free of reflection. Then, the structure of a three-port can be easily obtained by inspection of its equations. An example [18] is given in Fig. 2.

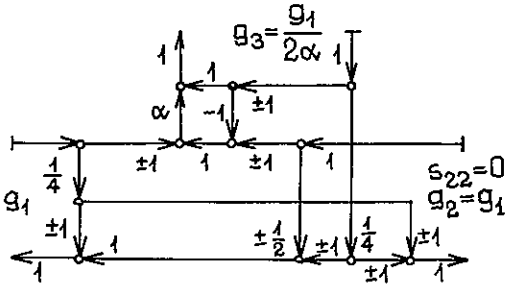


Fig. 2. A three-port that is l1-pseudopassive in DR for $|\alpha|<1$.

Unfortunately, there no general method of synthesis of filters consisting of such three-ports. The author tries to find an effective numerical method to design such filters.

4.3. Structures that are l1-pseudolossless in D+

In digital image processing, signal values are usually positive because they are proportional to the gray level of pixels. On the other hand, a filter with the transfer function

$$H(z_1,z_2) = \frac{\sum_{k=0}^{K}\sum_{l=0}^{L} a_{kl} z_1^k z_2^l}{1-\sum_{i=0}^{I}\sum_{j=0}^{J} b_{ij} z_1^i z_2^j} \qquad (7)$$

$(b_{00}=0, \ a_{kl}, b_{ij} \geqslant 0)$

has also output in D+ if its input is in D+. So, we restrict our further considerations to such filters which

are l1-pseudolossless and suitable for low-pass filtering of positive-valued signals.

There is the following condition of l1-pseudolosslessness in D+:

$$\sum_{i=1}^{N+2} g_i s_{ij} = g_j . \qquad (8)$$

This condition is fulfilled in the filters which are built up using the elementary structures shown in Fig. 3.

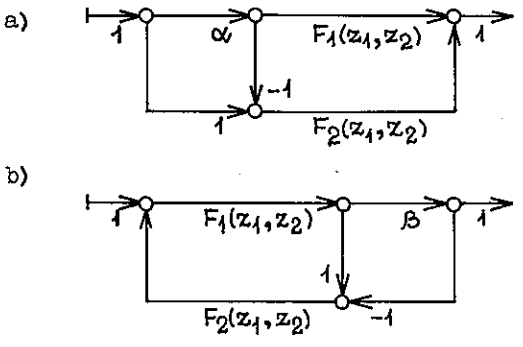


Fig. 3. The elementary structures of structurally l1-pseudolossless filters.

Such filters are structurally pseudolossless for $0<\alpha,\beta<1$. They are synthesized by matching the given transfer function, which is described by (7), and a transfer function given by Mason's formula for the graph transmission [22].

Example: A transfer function of the form

$$H(z_1,z_2) = \frac{\gamma[\alpha\beta+(1-\alpha)z_1+\alpha(1-\beta)z_2]}{1-(1-\gamma)[\alpha\beta z_2+(1-\alpha)z_1z_2+\alpha(1-\beta)z_2^2]}$$

is given. The corresponding digital network is shown in Fig. 4. Note that $\partial|H|/\partial\alpha=\partial|H|/\partial\beta=\partial|H|/\partial\gamma=0$ in the attenuation zero $\omega_0=(0,0)$. This implies small sensitivity in the passband.

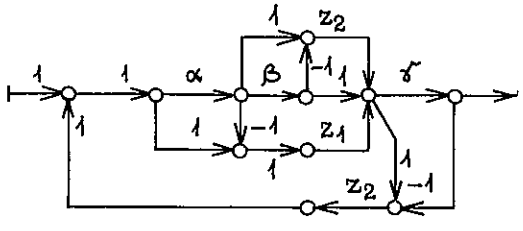


Fig. 4. A structurally l1-pseudolossless two-dimensional digital filter.

5. SUFFICIENT CONDITION FOR BIBO-STABILITY OF 2-D SYSTEMS

Proposition 5: If for each vertex of a signal flow graph the sum of the absolute values of the coefficients of all branches directed from the vertex does not exceed one the corresponding 2-D system is BIBO-stable.

This proposition is implied by l1-pseudopassivity of such a system.

REFERENCES

[1] Fettweis, A., Wave Digital Filters, in: Digital Signal Processing (VDE-Verlag, Berlin 1983) pp. 13-46.
[2] ---, IEEE Trans., CT-19 (1972) pp. 668-673.
[3] ---, IEEE Trans., ASSP-22 (1974) pp. 383-384.
[4] Fettweis, A. and Meerkoetter, K., IEEE Trans., CAS-22 (1975) pp.239.
[5] Renner K. and Gupta, S., IEEE Trans. CT-20 (1973) pp.555-567.
[6] Domański, M. and Piekarski, M., Voltage-Current DF with Zero Magnitude Sensitivity in Attenuation Zeros, in print.
[7] Domański, M., Synthesis of Two-Dimensional Pseudolossless Digital Filters, Doct. Thesis, Poznań Techn. Univ. (1983), in Polish.
[8] Renner, K. and Gupta, S., IEEE Trans., CAS-21 (1974) pp.305-310.
[9] Antoniou, A. and Rezk, M., IEEE Trans., CAS-27 (1980) pp.1184-1193.
[10] Fettweis, A., Proc. 1976 ECCTD (Genova, 1976) v.2, pp. 409-416.
[11] ---, IEEE Trans. CAS-25 1978 p.1060.
[12] Domański, M., 1984 ICASSP (San Diego).
[13] Linnenberg, G., On Discrete Multidimensional Signal Processing with Application of Wave Digital Filt., Doct. Thesis, Ruhr Univ. (Bochum, 1984), in German.
[14] Goodman, D., IEEE Trans., CAS-24 (1977) pp. 201-208.
[15] ---, Proc. IEEE, 66 (1978) pp.796-797.
[16] Chan, D., Asilomar Conf. Circuit Syst. Pacific Grove, (1977) pp.90-94.
[17] Aly, S. and Fahmy, M., IEEE Trans., CAS-27 (1980) pp.1175-1184.
[18] Domański, M., Generalized Pseudopassive Multidimensional Digital Filters, in: Proc. 8th Nat. Conf. Circuit Theory and Electr. Net. (Poznań, 1985) pp.44-48, in Polish.
[19] Jury, E.I., Proc. IEEE, 66 (1978) pp. 1018-1047.
[20] Fettweis, A., IEEE Trans., CAS-25 (1978) pp. 1060-1066.
[21] Orchard, H., Electron. Lett., 2 (1966) pp. 224-225.
[22] Zadeh, L.A. and Desoer, Ch.A., Linear System Theory (McGraw-Hill, 1963).

Tutorial on IMAGE CODING

M. Kunt
EPFL
Dept. of Electrical Engineering
Lausanne
Switzerland

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON MODIFICATION OF BLOCK TRUNCATION CODING APPROACH TO IMAGE COMPRESSION

E. Walach, D. Chevion, E. Karnin

IBM Israel Scientific Center
Technion City, Haifa 32000, Israel

Abstract - Block Truncation Coding is a technique for image compression in the spatial domain. An image is divided into small, mutually exclusive blocks, which are subsequently compressed utilizing an adaptive one bit quantizer. This paper suggests and analyses some modifications of the basic technique aimed at improving the image quality, and at the reduction of the overall bit-rate. Good quality images were obtained with a rate of 0.6 bit per picture element.

1. Introduction

Block Truncation Coding (BTC) technique for coding images in the spatial domain has been introduced by Delp and Mitchell in [1] . It is based on fragmenting the image into a number of small neighborhoods (of, say, 4x4 pels each) and, subsequently, obtaining a bilevel representation of each block of data. In other words, each block is represented by the two quantization levels and by the bit map which points out the places of the higher and the lower levels. The coding is performed in such a way that a certain optimization criterion will be fulfilled. In [1] , the BTC has been designed in such a manner that the reconstructed block preserves the first two moments of the original one.

Utilization of this approach provides a 32 bit representation of each 4x4 block of pels: 16 bits for the bit map and 16 bits for transmitting the quantization levels. Thus a 4:1 compression ratio is achieved. The quality of the reconstructed images proved to be quite good, making BTC attractive in terms of its robustness and high performance.

In recent contribution [2] some modifications of the basic technique have been proposed. The purpose of this manuscript is to extend this approach in a manner, which would improve both the quality of the reconstructed images and the compression ratio, while retaining the advantages of the basic structure of the BTC method. In order to achieve this goal, three distinct modifications are introduced. Two of them are aimed at improving the image quality, while the third one deals with increasing of the compression ratio:

1. A well known clustering algorithm for vector quantization is adopted in order to optimize, in the least squares sense, the bilevel representation of each block.
2. Further quality improvement is achieved by the introduction of a variation of a three-level block quantization scheme (instead of the bilevel one).
3. Compression ratio is increased by:
   a. coding of block averages
   b. splitting all the data blocks into classes and allocating different number of bits for each class. Since it is well known that the human eye resolution is proportional to the local variance, sparse bit-maps have been chosen for blocks with low internal variation.

   A heuristic image analysis has been performed in order to develop an optimal block classification procedure.

   Selective interpolation techniques have been adopted in order to filter out any "blockiness effects", which might have been introduced by the aforementioned procedure.

Combining all the modifications allowed us to achieve a good quality images with about 0.6 bits per pel (picture element) in contrast to the conventional BTC approach, which requires 2 bit/pel.

In subsequent sections (2-5) we will discuss in greater detail the 3 aforementioned modifications. In section 6 some simulation results will be presented.

2. Optimized Vector Quantization.

We have adopted the structure similar to that of the BTC quantizer i.e. we coded blocks of 4x4 pels utilizing two levels and a bit-map of 16 bits in order to allocate each pel to either one of these levels. However, we have found it advantageous to optimize the algorithm utilizing the concepts of the Lloyd-Max quantizer. In other words we wish to code each block, by two gray levels and by the corresponding bit-map, in such a way that the error will be minimized in the least squares sense.

In order to achieve this goal we have utilized the LBG clustering algorithm (see [2] - [3]). This is an iterative procedure, which, after a finite number of steps, converges to a sub-optimal solution. In order to reduce the

computational complexity, it is convenient to put some arbitrary limit on the allowable number of iterations of the LBG algorithm. If the initial clustering is chosen to be identical to that of BTC, and if only a single iteration is allowed, then the above mentioned approach yields the DBC (Differential Block Coding) algorithm (see [4] - [5]). An example of the DBC coded image is depicted in Fig. 2-d. Clearly, the quality is quite good. However, certain defects are noticeable. In particular, a close inspection reveals the presence of small indentations in the vicinity of sharp edges.

Next the same image has been coded utilizing the DBC code with the addition of a single iteration of the LBG algorithm. The result is depicted in Fig. 2-c. The irregularities, in the vicinity of the sharp edges, have been reduced and as a result the quality of the image has been enhanced (note, for instance, that the glint in the eyes has been preserved in Fig. 2-c, only).

3. Three level quantizer.

Despite the aforementioned improvement, in our experience, certain deterioration in the image quality will still be noticeable for certain cases, in which the block variance is extremely high. In relatively rare cases when such undesirable effect happened, it would be advantageous to utilize a three level quantizer. Naturally a 3 level quantizer yields a smaller error but of course one must transmit 3 quantization levels (instead of two), and replace the bit-map by ternary symbols. We have chosen a compromise (sub-optimal) three level quantizer, where one of the levels is fixed at the average of the other two. Hence, only two independent quantization levels have to be transmitted. Of course one still has to pay the "penalty" of a "ternary-map", which means that instead of 16 bits/block about 25 are required. As a result most of the benefits of the three level quantizer have been achieved with only insignificant increase in the information transmitted.

4. Bit Rate Reduction

So far we have discussed the issue of the potential improvement in the quality of BTC based algorithms. Next we will consider possible ways of reducing the number of bits required. There are two complementary facets to this problem: compression of the bit-map, and compression of the two block averages (quantization levels).

It turns out that good quality can be achieved with as few as 9 bits for the two block averages (instead of 16 mentioned above). Six bits are utilized for the high level, and the other 3 are used to quantize the difference between the two block averages, on a nonlinear scale. Further compression can be achieved even by a simple DPCM scheme.

In [6] and [7] , some ways of reducing the bit map from 16 to 12 or even 8 bits have been proposed. However, if a higher compression ratio is desired one has, adaptively, to skip the entire bit-map of the less significant blocks. This approach has been indicated in [8] , where the bit map has been set to "1" for blocks with a low difference between the high and low block averages. We believe that the effectiveness of this approach can be considerably enhanced if instead of two classes of blocks one would permit 4 or more classes. The classification would be based on the value of the difference between the block averages. Since the resolution of the human perception diminishes with the image contrast, the number of bits allocated to a block of each group would be changed. Twenty five bits would be allocated only to the "very high difference blocks" ("fourth class"). The "low difference" groups (of type 3, 2, and 1) will have 16, 8, and 0 bits respectively, and missing pels will be obtained by interpolation. In the last case (when no bit-map is transmitted), all pels will be set to a single block average.

Notice, that such block classification requires only a single bit/block overhead, in order to distinguish between the "no bit-map" blocks (where only a single average is required) and all the other blocks (where two quantization levels are transmitted) Since the block variance is known at the receiver's site, no additional information is required in order to distinguish between the various block classes.

The question arises: what kind of block classification procedure should be adopted? We believe that, strictly speaking, there is no generally optimal solution to this issue. Indeed, while it is intuitively clear that higher distortion is allowable for the areas with high local variance, the quantitative limits on unnoticeable distortion will depend on subjective evaluation pertinent to each specific application. However, practically useful criteria can be established. In our case, we have chosen to work with the curve of Fig. 1, which represents the allowable rmse (root mean square error) as a function of local variance.

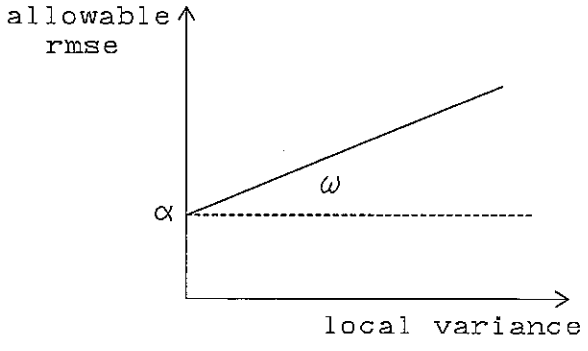


Figure 1. Allowable distortion limit.

At smooth areas relatively small error α is allowed. Distortion threshold increases proportionally to the local variance. The proportionality constant ω is chosen so

that good signal to noise ratio will be ensured for high contrast areas.

Based on our experience we have assumed $\alpha = 4$, and $\omega = 0.316$ (which corresponds to 10 db signal to noise ratio). Hence, for the block having local variance M the expected rmse e must fulfill

$$e \leq 4 + .316M. \tag{1}$$

Next we will estimate, for each type of blocks, the expected rmse. As a result we will be able to determine how to classify each data block so that the aforementioned distortion limits will be fulfilled. In order to achieve this goal some general (and therefore necessarily crude) statistical assumptions about the nature of the image data will be required. Hence, our analysis can be viewed as having a heuristic nature. However, the resulting block classification procedure does agree closely with simulation based conclusions.

Assume that, for each block, the gray levels of the pels are uniformly distributed in $\pm R$ range around the block average. The expected value of the difference between the "high" and "low" block averages, will be equal to R . The local variance (second moment) will be equal to $M = R/\sqrt{3}$. Denote by e the rmse error between the modified BTC representation of such a block and the original. This distortion will be caused by the quantization error e_q and (for blocks of second kind) interpolation error e_i The quantization error can be estimated as:

$$e_q = \frac{1}{\sqrt{3}} \frac{R}{n}, \tag{2}$$

where n stands for the number of quantization levels. The interpolation error depends heavily on the degree of correlation between the neighboring pels and on the type of the interpolation technique, which has been adopted. Based on our simulation results, we have assumed that, for the blocks of the second class, the interpolation and the quantization errors are equal and orthogonal to each other. Hence,

$$e = \sqrt{e_i^2 + e_q^2} = \sqrt{2}\, e_q. \tag{3}$$

Utilizing expressions (2) and (3) we can compute the expected rmse for the blocks of class 1-4 as: $R/\sqrt{3}$, $R/\sqrt{6}$, $R/\sqrt{12}$, and $R/\sqrt{27}$ respectively. Substituting these values into (1) yields the desirable block classification thresholds:

$$\begin{aligned} &\textit{class } 1 \textit{ for } 0 \leq R \leq 10 \\ &\textit{class } 2 \textit{ for } 10 < R \leq 18 \\ &\textit{class } 3 \textit{ for } 18 < R \leq 38 \\ &\textit{class } 4 \textit{ for } 38 < R \leq 405 \end{aligned} \tag{4}$$

Clearly, the proposed classes of blocks suffice in order to keep the reconstruction error below the allowable limits for the entire range of possible values of $0 \leq R \leq 255$. Despite some crude approximation which have been assumed in the course of the above derivation, the thresholds given by (4) closely agree with psychovisual characteristics of the human eye.

Of course, in practice, in order to control the compression ratio, one might prefer to perform some shift in the values of the thresholds. In such a way, for instance, it is possible to increase the compression ratio at the expense of some degree of deterioration in the image quality.

5. Selective Interpolation

The quality of the modified BTC coded image is quite good but some degree of "blockiness", due to bit-map reduction, is noticeable. However, this undesirable effect can be easily dealt with by means of selective interpolation technique. The interpolation will be applied exclusively to the blocks without bit-maps (class 1). Thus the "blockiness effect" will be removed without the concomitant lowpass filtering of the high resolution areas.

The interpolation (or smoothing) of each block (of the first class) is performed based on gray values of 20 pels surrounding the block and based on the average value of the block itself. We are utilizing the polynomial interpolation of the second degree, i.e. we are approximating the block by the function:

$$f(x,y) = \sum_{i=1}^{2}\sum_{j=1}^{2} a_{ij} x^i y^j. \tag{5}$$

The polynomial coefficients a_{ij} are chosen such that the mean square error at the neighboring pels will be minimized, keeping the block average constant. This is a straightforward case of constrained optimization, and it can be solved using, for instance, Lagrange multipliers. Once the optimal polynomial coefficients are found, each pel of the interpolated block can be obtained as a linear combination of the known values.

6. Simulation Results

In order to verify the proposed approach we have combined all the aforementioned modifications and applied them to various images. As an illustrative example consider the well known 256x256 head and shoulders image depicted in Fig. 3-a (Fig. 2-a). In Fig. 3-b classification of the various blocks is depicted. Each class is represented by a different color: white corresponds to blocks with no bit-maps, different shades of grey correspond to other classes. Clearly in about 80% of cases no bit-maps are required. As a result a high compression ratio of 0.6 bit/pel has been achieved. The reconstructed image is

depicted in Fig. 3-c. Some degree of "blockiness", due to bit-map reduction, is still noticeable. In order to mitigate this undesirable effect selective interpolation procedure is applied. The result is depicted in Fig. 3-d. In such a way an excellent image quality is achieved with 0.6 bits/pel.

Acknowledgements

The authors wish to express their deep gratitude to Mr. R. B. Hilgendorf and to Mr. U. Shvadron for their help in preparing this manuscript.

References

[1] E.J. Delp and O.R. Mitchell, **"Image Compression Using Block Truncation Coding"**, *IEEE Trans. on Communications, vol. COM-27,* September 1979.

[2] E. Walach, J. Bruck, D. Chevion, E. Karnin, D. Ramm, **"A Modified Block Truncation Coding Technique for Image Processing"**, *Proceedings of International Conference on Advances In Image Processing and Pattern Recognition, Pisa,* December 1985.

[3] Y. Linde, A. Buzo, and R.M. Gray, **"An Algorithm for Vector Quantizer Design"**, *IEEE Trans. on Communications, vol. COM-28, pp. 84-95,* January 1980.

[4] M. Kobayashi and T. Yamamoto, **"A Color Coding Scheme for Facsimile Signals"**, *Picture Coding Symposium, France,* July 3-5, 1984.

[5] M.D. Lema and O.R. Mitchell, **"Absolute Moment Block Truncation Coding and Its Application to Color Images"**, *IEEE Trans. on Communications, vol. COM-32,* October 1984.

[6] O.R. Mitchell and E.J. Delp, **"Multilevel Graphics Representation Using Block Truncation Coding"**, *Proc. IEEE, vol. 68,* July 1980.

[7] G.R. Arce and N.C. Gallagher, **"BTC Image Coding Using Median Filter Roots"**, *IEEE Trans. on Communications, vol. COM-31, pp. 784-793,* June 1983.

[8] Y. Yasuda, Y. Yamazaki, T. Kamae, and K. Kobayashi, **"Advances in Fax"**, *Proc. IEEE, vol. 73, pp. 706-730* April 1985.

Fig. 2. The result of applying BTC, DBC and LBG (2bit/pel) to the girl image 256x256x8. Starting from the top left clock-wise: a. the original, b. BTC, c. LBG (one iteration), d. DBC.

Fig. 3. The results of Modified BTC algorithm. Starting from the top left clock-wise: a. the original 8bit/pel, b. block classification, c. reconstructed image (0.6 bit/pel), d. reconstructed image after selective interpolation post-filter (0.6 bit/pel).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON FRACTAL BASED APPROACH TO IMAGE CODING

E. Walach, E. Karnin, D. Chevion

IBM Israel Scientific Center
Technion City, Haifa 32000, Israel

ABSTRACT - We describe a new approach to the issue of lossy data compression. The basic concept has been inspired by the theory of fractal geometry. The idea is to traverse the entire data string utilizing a fixed length "yardstick". The coding is achieved by transmitting, only, the sign bit (to distinguish between the ascent and the descent) and the horizontal distance covered by the "yardstick". All data values are estimated, at the receiver's site, based on this information.

We have applied this approach in the context of image compression, and the preliminary results seem to be very promising. Indeed, the proposed approach is very simple (both conceptually and from the point of view of computational complexity), and it seems to be well suited to the psycho-visual characteristics of the human eye.

The paper includes a brief description of the coding concept. Further a number of possible modifications and extensions are discussed. Finally a number of simulations are included in order to support the theoretical results. Good quality images are achieved with as low as .5 bit/pel.

1. Introduction and Basic Concepts

Modern fractal geometry has been introduced by B.B. Mandelbrot in the late seventies [1] , [2] . Since then it became a powerful tool in statistics, physics, texture analysis etc. (for extensive bibliography see [3]). In [4] a simple image compression algorithm, inspired by similar ideas, has been introduced. Subsequently we will discuss and extend various aspects of the proposed algorithm.

To the degree that one accepts the Mandelbrot's manifesto (see [3]): "there is a fractal face to the geometry of nature", it is feasible that this novel approach to the image compression (or in general data compression) issue might provide a number of advantages over the existing techniques. We see its potential strength in providing a simple solution to the need for an image compression algorithm which will, on one hand, take into consideration psycho-visual characteristics of the human eye, and, on the other hand, will be flexible enough to provide an easy control of the compression ratio as a function of the desired image quality.

The algorithm itself is akin to the measuring of the length of the coastline by setting dividers to a prescribed constant opening y , to be called the yardstick length, and walking the coastline, each new step starting, where the previous step left off. Naturally, in our case the "coastline" will be constructed by drawing a line through the grey levels of all the pels of the image under consideration.

We start from the first line of the image. Imagine that the grey levels of the pels define the edge line of some solid two-dimensional object. We put one end of our yardstick on the level defined by the grey level of the first pel in line. Then we let the yardstick fall until its other end will be "stopped" by our imaginary edge line. [1] Then we advance the yardstick to this new location and continue the process until the entire image will be "covered". Denote by t_i the horizontal distance covered by step i. Since we have assumed that the length of the yardstick is y,

$$0 \le t_i \le y. \tag{1}$$

Knowing the value of t_i. and the vertical location g_i of one end of the yardstick, one can determine the location of the other end as:

$$g_{i+1} = g_i + sign_i \times \sqrt{y^2 - t_i^2} \tag{2}$$

($sign_i$ distinguishes between the cases of ascent and descent). Hence, the entire process, which has been described above, can be uniquely characterized by the values of t_i ,and $sign_i$.

[1] For the sake of simplicity we treat our data as a "comb" of values located at each data sample, rather then a continues edge line. Hence, the "yardstick" can be stopped only at one of the data samples, and not in between them.

The aforementioned procedure can be utilized for compression purposes. The "yardstick travelling" will be performed at the transmitter's site. The values of t_i ,and $sign_i$. will be coded and transmitted to the receiver's site, where the "yardstick approximation" of the original will be reconstructed. The coded representation of each step of the algorithm will require at most [2] $1+\log_2(1+y)$ bits.

It is well known (see [3]) that if the aforementioned procedure is applied to a fractal having the dimension d , the entire process will require

$$fy^{-d} \tag{3}$$

steps, where f is a proportionality factor (which depends on the number of data samples). Hence the overall number of bits required will be

$$b = fy^{-d}(1+\log_2(1+y)), \tag{4}$$

Note that, strictly speaking, digital images cannot be characterized as fractals. Indeed, it is easy to create conditions (say, for a very large length of the "yardstick" y) such that expressions (3) and (4) will not be fulfilled. However, for a wide range of reasonable values of y equations (3) and (4) hold. Moreover, the fractal dimension d is closely related to the complexity of the image. It would be close to unity for the smooth areas, and small (relative to unity) for complex, difficult to compress images. For one dimensional Wiener processes d will be equal to 0.5.

It seems that the parameter d provides, what was badly missed in the existing studies for algorithms comparison: an objective means for classification of the image complexity.

The proposed compression algorithm seems to be quite consistent with the known characteristics of the human visual system. Indeed, it will automatically decrease the resolution in the smooth low frequency areas, while retaining a small quantization error. At the same time, at the frequency areas horizontal distances. t_i will be small, and thus the resolution will be increased.

It should be noted, also, that the computational complexity of the proposed technique is relatively low. In fact, no multiplications are required during the entire "yardstick measuring" procedure (calculations of equation (2) will be carried out by the corresponding look-up table).

[2] In practice the degree of compression can be further increased by application of one of the existing loss-less compression techniques.

The degree of compression will be controlled, according to (3), by a single design parameter: the yardstick length y. . Naturally, a decrease in the value of y will reduce the degree of compression, and simultaneously, improve the quality of the reconstructed image. Hence, in practice the value of y should be chosen based on the required degree of detail, whic in turn, will be determined by factors such as zoom, quality of the monitor, the distance between the viewer and the monitor etc. .

2. Extension to Two-Dimensional Arrays

So far we have treated the proposed algorithm as having one-dimensional structure i.e. the entire data set has been viewed as one long string. An enhanced performance can be expected if two-dimensional nature of image data would be efficiently utilized. Detailed analysis of this important issue goes beyond the scope of this brief contribution. However, subsequently, we will indicate a number of possible approaches.

1. Utilization of inter-line correlation:

   The lines of the image can be sub-sampled using the factor of, say, 1:2 or 1:4. The decimated data will be coded utilizing the aforementioned technique. Missing lines will be predicted based on interpolation of the known data. In order to improve the overall image quality high prediction errors (say above 20) will be subsequently corrected. Since such corrections will be relatively few (they will appear only at the vicinity of very sharp edges), high compression ratio will be maintained while at the same time subjectively important edge continuity will be improved.

2. Fractal based coverage of the image:

   The idea is to rearrange the image in the form of an one-dimensional array. However, instead of merely "stringing" the data line by line, it would be advantageous to adopt a structure, which will increase the correlation between the neighboring pels. One possible approach will be to traverse the image along the Peano curve (see, for instance, [3]), which interestingly enough, is by itself a fractal.

3. Two dimensional "triangulation" of the image:

   Instead of traversing an one-dimensional array, utilizing one-dimensional yardstick, two-dimensional array will be "covered" by equilateral triangles. First the boundary of the image will be compressed utilizing the one-dimensional technique. Next, a "triangulation" process will be performed. Each step will be started from a known section (side of a triangle). One end of the one-dimensional "yardstick" will be placed at the center, and appropriate fractal distance will be found at the direction perpendicular to that of the given section. Horizontal distance covered by the

yardstick will be transmitted to the receiver and the entire process will be repeated for a new section.

Note that equilateral triangles will "cover" the "surface" of the image. The corresponding projections on the horizontal plane will not be of the equilateral type. As a result the entire procedure is not as straightforward as its one-dimensional counterpart. Nevertheless, the intuitive appeal of the fractal approach is preserved: in smooth parts of the image horizontal projections will have large areas allowing high compression ratio, while in high frequency areas of the image the triangles will be smaller causing an automatic increase in the resolution.

3. Simulation Results

For simulation purposes we have chosen a well known 256x256 face image (see Fig. 1). In order to enhance the overall performance we have applied a nonlinear low-pass filter (similar to the one in [5]) both as pre- and post- filter.

Next we have applied the aforementioned "yardstick measuring" procedure, in order to verify the validity of expression (3). For a number of choices of y , the number of measuring steps has been found, and then least squares estimation of f and d has been obtained. We have computed f to be 99334 (or about 1.5 /pel). The fractal dimension d has been found to be 0.63. Using these two values the number of yardstick steps can be computed for an arbitrary choice of y . For y ranging from 8 to 32 we have found expression (3) to be precise within less then 3%.

Figures 2-4 illustrate the quality of reconstructed images for the choice of y equal to 7, 15 and 31 respectively. In order to enhance the overall performance, we have adopted a number of modifications, which have been introduced in [4] , including the modified trigger function (TF), and preferential treatment of sharp edges. In addition we have adopted 1:2 line sub-sampling with subsequent interpolation and error correction for the "missing lines". The entropy of the codes of these images has been 0.92, 0.7 and 0.6 bit/pel correspondingly. Excellent quality has been achieved for y = 15, and 0.7 bits/pel.

Further increase in the compression ratio can be achieved by sub-sampling the lines by the factor of 1:4. Figures 5 and 6 depict the results achieved with a different degree of prediction error correction. The corresponding degree of compression is 0.5 and 0.4 bit/pel respectively.

4. Conclusion

A new approach to the issue of image coding has been presented. It can be utilized for classification of the image complexity. Moreover, it can be used directly for the purpose of image compression. Easy control of trade-off between the resolution and quantization errors is obtained and as a result a good utilization of human vision characteristics is achieved. Even at the present preliminary stage, excellent quality images has been obtained with as low as 0.5 bits/pel.

Acknowledgements

The authors wish to express their deep gratitude to Dr. A. Bruckstein, Mr. R. B. Hilgendorf and to Mr. U. Shvadron for their inspiring discussions of the subject and help in preparing this manuscript.

REFERENCES

[1] B.B. Mandelbrot, **"Stochastic Models for the Earth's Relief, the Shape and the Fractal Dimension of the Coastlines, and the number-area rule for islands"**. *Pr. of the National Academy of Sciences, vol. 72,* 1975, pp. 3825-3828.

[2] B.B. Mandelbrot, **"Fractals: Form Chance and Dimension"**, *San Francisco: W.H. Freeman and Co.,* January 1977.

[3] B.B. Mandelbrot. **"The Fractal Geometry of Nature"**, *San Francisco: W.H. Freeman and Co.,* 1983.

[4] E. Walach and E. Karnin, **"A Fractal Based Approach to Image Compression"**, *Proc. of ICASSP 86,* April 1988.

[5] W.B. Pennebaker, J.L. Mitchell, K.S. Pennington and D. Anastassiou, **"A High Performance Freeze Frame Videoconferencing System"**, *Proc. of Globecom 83,* November 1983, pp. 16.5.1-16.5.8.

Fig. 1. Original of the 256x256 face image. Dimension d = 0.63.

Fig. 2. Compressed image for y = 7, entropy equals 0.92 bit/pel.

Fig. 3. Compressed image for y = 15, entropy equals 0.70 bit/pel.

Fig. 4. Compressed image for y = 31, entropy equals 0.60 bit/pel.

Fig. 5. Compressed image for 1:4 line sub-sampling with full prediction errors correction, y = 31, entropy equals 0.5 bit/ pel.

Fig. 6. Compressed image for 1:4 line sub-sampling with fragmentary prediction errors correction, y = 31, entropy equals 0.4 bit/pel.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A PYRAMID BASED IMAGE CODING

D. Chevion, E. Karnin, E. Walach, U. Shvadron

IBM Israel Scientific Center
Technion City, Haifa 32000, Israel

Abstract - We describe a very efficient, yet simple, approach to image compression, that utilizes ideas from the areas of progressive transmission, predictive coding and entropy coding. This technique is based on an effective form of scanning the data, such that at every stage of the algorithm four nearest neighbors (from previous layers) are known both at the transmitter's and the receiver's sites. The availability of this prior information is utilized in order to adaptively improve the prediction, quantization and lossless compression aspects of the algorithm.

1. Introduction

Previous works which took advantage of the notion of progressive transmission [1] , built the pyramid in a bottom up fashion, i.e., each layer was obtained by some kind of averaging of the layer below it. In [2] one climbs up the pyramid by convolving the pixels of a given layer with a Gaussian kernel, while in [3] simple block average is used (where a block consists of 2 pixels).

We propose a different approach, where the information passed to a higher level is merely a single pixel of the block, thus eliminating all the arithmetic operations. In other words, one does not construct the pyramid bottom up, but rather scans the pixels of the original image in what we call dot-interlaced scanning.

The first level of the pyramid is transmitted to the receiver utilizing, say, 6 bits /pel. For every pel of each subsequent level the four nearest neighbors, belonging to the previous layers, are known at the receiver's site (as opposed to the conventional raster scan where prior information is available from two sides only, namely left and top). Based on the known values of the neighboring pels a prediction value is computed for each pel of the image. Next, the prediction error is quantized and losslessly coded.

The availability of information, regarding the neighboring pels, is utilized in order to adopt the quantizer and lossless coding scheme to the specific character of each area. As a result, our particular form of pyramid turns to be quite effective in utilizing prior information in all the stages of the image compression algorithm. Hence the over-all performance is improved.

In order to enhance the performance of the proposed algorithm, it is useful to apply pre- and post-processing. We found it advantageous to "clean" the original image by a non-linear low pass filter, e.g., the one used in [4] , where the non-linearity helps keeping the transitions (edges) almost intact. In addition, it might be useful to apply a point transformation, which is a logarithmic like function that emphasizes the low grey levels. At the receiver site the reconstructed image will be transformed by the inverse function.

In the subsequent sections (2-4) we will discuss briefly various aspects of the image compression algorithm:

scanning and prediction;

activity measure and quantization;

lossless compression

Finally, in section 5 some simulation results will be presented.

2. Dot-interlaced pyramid scanning.

Let the image be a square of 2^L by 2^L pixels, then the 0-th layer contains a single element at (0,0), and layers 1 through $2L$ are constructed as follows. For each layer the number of pixels is doubled, by filling-in the centers of the squares, whose vertices are defined by the pixels at the higher (previous) layer. (Edge effects are treated by periodic continuation of the image in both horizontal and vertical directions.) For example, at layer 1 a pixel at $(2^{(L-1)}, 2^{(L-1)})$ is addressed, which is the center of (0,0), $(0,2^L)$, $(2^L,2^L)$, and $(2^L,0)$.

Similar scanning scheme has been utilized successfully in [5] , for the purpose of coding bilevel images. In this reference the known values of the neighboring pels provided a context for the lossless compression scheme. In our case, we are dealing with "lossy compression" of 8-bit deep grey-tone images. Accordingly, the prior level information is utilized differently: the grey values of the neighboring pels are utilized in order to compute the prediction at the given point.

Note that the predictor for each pixel is based on the interpolation of its neighbors (of the previous layer). In contrast to the raster-scan where a predictor is derived by extrapolation.

The choice of the appropriate interpolator will depend on a compromise between the desired efficiency and the allowable computational complexity. The simplest interpolator is the bilinear one, where merely the average of the 4 nearest neighbors is computed. Better results are obtained by two-dimensional, separable, cubic convolution on 16 points. Still better are some variations of the non-linear interpolator, which chooses the preferred orientation for interpolation.

3. Quantization Procedure.

Once the predictor is chosen, and prediction value is computed, the error signal is obtained as the difference between the actual pixel value and its predictor. Next the prediction error is quantized.

The hierarchy of the pyramid structure is exploited by designing different quantizers for different layers. Naturally, we use fine quantizers for the lower layers, and coarse quantizers for the higher ones, as was done in [2] .

Moreover, it is intuitively clear that it would be advantageous to have a spatially adaptive quantizer, which will be tuned to the different areas of the image. Indeed, in smooth areas even small noise might be noticeable. On the other hand in areas having large local variation even relatively large absolute error will be negligible (see [6]). Correspondingly we have defined, for each pel, an activity measure

$$AC = (\max G_i) - (\min G_i), \tag{1}$$

where G_i represents the known grey-levels at the four nearest neighboring pels. Note that activity is defined in such a manner that it can be determined both at the transmitter's and at the receiver's sites. Hence, it can be utilized for the control of the quantizer without necessitating transmission of any overhead information.

In practice a bank of non-linear quantization tables is stored both at the transmitter and at the receiver. In our experience it suffices to have 8 different quantizers: one for each layer of the pyramid. At the lowest layer the finest quantizer is utilized. Subsequently, with each new layer, the algorithm switches to a coarser quantizer. However, if the local activity measure AC is low then the algorithms moves, one or two places, to a table corresponding to a lower layer, and the number of quantization levels is increased. On the other hand if activity is high then the opposite phenomenon occurs: the algorithm switches to a coarse quantization table, and as a result the compression ratio is improved.

4. Lossless Compression of the Quantization Levels

In practical images the distribution of all the pels between the various quantization levels is highly non-uniform. Moreover, in the overwhelming majority of cases a low prediction error can be expected for the pels having a low activity measure. Hence a high degree of compression can be achieved by utilization of an appropriate form of lossless coding

In order to realize, in a relatively simple way, the potential for lossless compression we have decided to distinguish between the two cases:

frequent cases, in which the prediction error behaves "normally" i.e it is confined to a certain predefined range;

rare cases, in which the error falls outside the expected range.

Naturally, some overhead information will be required in order to allow the receiver to distinguish between these two cases. A priori such information may,by itself, require one bit for every pel of the image. However, the expected range ER will be defined in such a manner, say

$$ER = AC/2, \tag{2}$$

that only in a fraction of cases the actual error will exceed the prediction. Hence, the side information file (regarding the errors in expected range of the quantization levels) can be very efficiently compressed utilizing, for instance, Arithmetic Coding (see [7]).

Once the range of the error is known, it can be coded very efficiently. Indeed, inside the range the quantizer has only a limited number of levels. Frequently, no quantization levels appear inside the ER and as a result no additional bits are required in order to reconstruct the given pel.

If the pel under consideration has an error exceeding the range ER , then non-uniform length code is used. However, since such cases are relatively rare, only a small number of additional bits will be required.

To summarize our approach to the lossless compression: for a given layer we use a non-uniform quantizer such that the number of permissible quantization levels depends on some local activity measure. This idea was first introduced in [4] . However, since in our case the data is organized in the hierarchical manner, a significant improvement is possible. Rather than using the previously encoded error signals as an activity measure ([8] , [4]) we compute an activity indicator (1) based on the dispersion of the surrounding pixels (from the previous layer).

5. Analysis and Simulation Results

In the sections 2-4 we have described a complete approach to the issue of image compression including

scanning, prediction, adaptive quantization and lossless coding. It should be noted that the entire algorithm has been structured in such a way that the progressive transmission is possible. Indeed, at any layer, one might stop the process of prediction error correction. As a result an increased compression ration will be achieved at the expense of reduced resolution.

For instance, while dealing with "head and shoulder" images it might be useful to have a simplified version for the purpose of fast scanning of the portrait files. In our experience, good image recognition can be achieved even with the skipping of the last three layers.

Another area, in which the advantages of progressive transmission are very important, relates to coding of colored images. It is convenient to represent the colored images by their luminance and chrominance components, and to compress each component separately. It is well known that the resolution information is contained mainly in the luminance data (see for instance [9]). Hence, without any subjective quality deterioration, one can skip 2 or 3 last pyramid layers of the two chrominance components. Moreover, since changes in chrominance are usually accompanied by changes in luminance (see [9]), one can limit error correction of the chrominance components to the pels for which luminance activity is high. Using such techniques will reduce the chrominance information to about 10% of the overall data. As a result high compression of colored images is possible utilizing an algorithm which is essentially a slight modification of black and white coding scheme.

The aforementioned compression algorithm seems to be quite efficient both in terms of compression ratio and computational complexity, As an example consider the well known black and white Face image depicted in Fig.1. This 256x256 original is 8 bit deep. We have applied to it our Pyramid Compression Algorithm.

Fig. 2 illustrates the structure of the pyramid process. In the top left corner of the image the first pyramid layer is presented (1 to 8 decimation both in X and Y directions). Absolute prediction errors pertaining to the second level appear below, the third is on the right, and similarly prediction errors of all the other layers have been depicted. For the display purposes all errors have been amplified by the factor of 4. Note that in each new layer the number of pixels is doubled but the percentage of the significant errors is reduced. Naturally, the overall number of pels in all the layers is equal to the number of pels in the original image.

The reconstructed image has been presented in Fig. 3. The entropy of the data is 0.55 bits/pel. The quality of the image is quite good. Indeed, judged by the human perception the reconstructed image is almost indistinguishable from the original, and the RMS of the difference is about 2.5 (on a scale of 256 grey levels).

In order to check the sensitivity of the algorithm to the choice of the various parameters, we have repeated the compression process utilizing more coarse quantizers. The result is depicted in Fig. 4. The compression ratio has been increased i.e. data entropy has been reduced to 0.45 bits/pel. However only slight deterioration in the image quality is noticeable.

List of References

[1] K. L. Sloan and S. L. Tanimoto, **"Progressive Refinement of Raster Images"**, *IEEE Trans. on Computers, vol C-28, pp. 871-874,* November 1979.

[2] P. J. Burt and E. H. Adelson, **"The Laplacian Pyramid as a Compact Image Code"**, *IEEE Trans. on Communications, vol. COM-31, pp. 532-540,* April 1983.

[3] A. Sanz, C. Munoz and N. Garcia, **"Hierarchical Predictive Approach to Image Coding"**, *Proc. ICASSP-85, pp. 113-116,* March 1985.

[4] D. Anastassiou, J. L. Mitchell and W. B. Pennebaker **"A High Compression DPCM Based Scheme for Picture Coding"**, *Proc. ICC-83, pp. 453-457,* 1983.

[5] T. Endoh and Y. Yamazaki **"Progressive Coding Scheme for Interactive Image Communications"**, *Proc. ICC-84, pp. 1426-1433,* 1984.

[6] E. Walach, E. Karnin and D. Chevion **"On Modification of Block Truncation Coding Approach to Image Compression"**, *Proc. EUSIPCO-86,* 1986.

[7] J. Rissanen and G. Langdon, **"Universal Modeling and Coding"**, *IEEE Trans. on Information Theory, vol. IT-27, pp. 12-23,* January 1981.

[8] H. G. Mussman, P. Pirch and H. Grallert, **"Advances in Picture Coding"**, *Proceeding of the IEEE, vol 73, pp. 523-548* April 1985.

[9] J. O. Limb and C. B. Rubinstein, **"Plateau Coding of the Chrominance Component of Color Picture Signals"**, *IEEE Trans. on Communications, vol COM-22, pp. 812-820,* June 1974.

Fig. 1. The original 256x256x8 girl image.

Fig. 2. Prediction errors of the various pyramid layers.

Fig. 3. Reconstructed image obtained utilizing a fine quantizer: 0.55 bit/pel.

Fig. 4. Reconstructed image obtained utilizing a coarse quantizer: 0.55 bit/pel.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

LINEAR AND NONLINEAR IMAGE RESTORATION METHODS IN COMPARISON

T.J. Uhl

Bundeskriminalamt, KI 22 Bildverarbeitung
Wiesbaden, Germany

Four linear Wiener filter methods and the nonlinear Maximum Entropy algorithm are compared. The characteristics of the methods are shortly summarized. For two Wiener filter methods we show how the corresponding suboptimal filter parameter can be found automatically. Diagrams and appropriate image data show the efficiency of the different algorithms.
An important aspect is to show the behaviour of the methods if the image data is partially destroyed beyond noise and image blur. The superiority of the Maximum Entropy algorithm in that case is demonstrated.
The investigations are carried out with synthetic images as well as with real-world life images.

1. INTRODUCTION

During the last twenty years an enormous amount of research has been done in the field of image restoration /1/.
A great variety of different restoration conceptions have been designed and the corresponding algorithms have been developed and tested.
But in spite of all these efforts there remains one problem for the expert: How to select the appropriate method for his restoration case in hand out of this great pool.
In principle he can choose :
- linear and nonlinear methods
- methods working in the frequency domain or directly processing in the space domain
- statistical methods or methods working in a more heuristic manner
- various optimization criteria (mse-,sharpness, entropy, etc.) and different constraints (positiveness, smoothness, etc.)

Nevertheless one can state that among the linear methods the Wiener method is one of the most popular restoration procedures. Whereas among the nonlinear techniques the Maximum Entropy algorithm has gained a lot of attention in the recent past.
Compared to the deterministic procedures the nonlinear methods are a lot more computationally expensive. So there is the question, in what cases it is worthwhile to use these computationally heavy loaded algorithms instead of other cheaper methods.
Besides presenting our general experiences with these methods it is the main object of the following considerations to answer that problem.

2. THE METHODS UNDER CONSIDERATION

Although pictures are usually two-dimensional we use a one-dimensional notation and drop the arguments, for simplicity. In the following let be g_i ,f_i ,h_i , r_i (i=1,...,N) the vektors of the blurred image, of the original image, of the point spread function and of the noise, resp.. Then G, F, H and R define the corresponding spectra and H* denotes the complex conjugate of H. Further let EPS, ETA be two appropriate positive constants.
Now the Wiener filter method yields the approximation spectrum $\hat{F}$ of F by the equation $\hat{F} = K_W G$ and the filter K_W is given by

$$(1) \qquad K_W = H^* |I|^2 / (|H|^2 |I|^2 + |R|^2) \quad .$$

The usefullness of Wiener filtering in image restoration is undoubted. But of course in practice you generally do not know all the quantities needed for K_W. Especially I is unknown and $|I|^2$ can only be estimated at best. Often difficulties arise from the noise spectrum. So you have to look for appropriate alternatives to the original Wiener filtering.

In the following we define four linear Wiener filter methods which all use filters basing on the Wiener filter K_W.

2.1 Pseudowiener filter (PSWI)

If you have no knowledge about I , R the quantity $|I|^2/|R|^2$ in (1) can be replaced by a constant ETA. You obtain the filter

$$(2) \qquad K_{PSWI} = H^*/(|H|^2 + ETA) \quad .$$

Let $\bar{H}$ be a suitable value characterizing the mean value of $|H|^2$ without the zero-near values and neglecting the low frequency values. We find, if ETA >> $\bar{H}$ then K_{PSWI} is a low pass whereas if ETA << $\bar{H}$ the PSWI-filter method passes over to an inverse filtering. We therefore conclude to choose ETA near the value $\bar{H}$. Experiences have proven this choice to be rather successful; especially this choice can be done automatically and two to three automatic tests are usually enough to find an efficient parameter ETA.

2.2 Wiener filter with ideal spectrum approximation (WISA)

In general you do not know the ideal spectrum I. But if you are lucky you will have some knowledge about the shape of the power spectrum $|I|^2$. If we suppose its symmetry, we can simulate $|I|^2$ by a radial symmetric function V. Thus we get the filter

$$(3)\quad K = \begin{cases} H^*V/(|H|^2 V + ETA) & \text{if } |H|^2 V + ETA > EPS \\ H^*V/EPS & \text{else} \end{cases} .$$

The (automatic) choice for the constant ETA is led by the same arguments as above. We deduce that $\bar{H} \approx ETA/V$. If $|V| < 1$, which can always be achieved by a norming operation, then we have to choose ETA always smaller than $\bar{H}$ (one to two powers of ten).
For the tests we use a Gaussian vector with the standard deviation σ as parameter to generate V. This choice is due to the fact, that the power spectrum shape can be well approximated by bell-functions for a lot of pictures.

Here we state that experiences with this filter with suboptimal σ showed results which seemed to be a little bit sharper than the comparable restorations PSWI. Indeed, let us add to the known Wiener minimum mean square error (mmse) criterion the additional constraint of sharpness S :

$$S \equiv \sum_{i=1}^{N} u_i^2 |F(u_i)|^2 \quad , u_i \text{ frequencies.}$$

Then this additional constraint implies the filter (see /1/,p 207)

$$K = K(u) = H^*/(|H|^2 + |R|^2/|I|^2)\cdot(1/(1 + cu^2)) .$$

where c is the Lagrange multiplier. Obviously, we have a strong ressemblance between this filter and the filter in (3). Both filters attenuate the high frequencies by a bell like function (provided c>0) and they consist of nearly the same terms affecting the low frequencies.

2.3 Wiener filter with noise spectrum (WINO)

If there is additional knowledge about the noise you can replace (3) by the filter

$$(4)\quad K = \begin{cases} H^*V/(|H|^2 V + |R|^2) & \text{if } |H|^2 V + |R|^2 > EPS \\ H^*V/EPS & \text{else} \end{cases} .$$

Of course one can expect better results from that filter only if the noise spectrum has a distinct shape in higher frequencies.

2.4 Cannon filter (CANO)

To make a compromise between the low pass effect of the Wiener filter and the early singularity of a plain invers filtering you can build a filter by the geometric mean of the filters of these two methods. You obtain /3/

$$(5)\quad K = \begin{cases} H^*/(|H|^2(|H|^2 + ETA))^{1/2} & \text{if } |H|^2 > EPS \\ H^*/(EPS\ (1+ETA/EPS)^{1/2}) & \text{else} . \end{cases}$$

For practical reason the formula applies the Pseudowiener filter instead of the Wiener filter.

2.5 Maximum Entropy algorithm (ME)

This method maximizes the modified form S of the picture entropy /2/

$$S = -\sum_{j=1}^{N} f_j \ln(f_j/eA)$$

over the chi-squared

$$\chi^2 = \sum_{i=1}^{N} (g_i - r_i)^2/\sigma_i^2 \quad , \sigma_i^2 \text{ variances.}$$

Here A is a positive constant and is given by /2/

$$A = \sum_{i=1}^{N} g_i/N \quad .$$

This constant does not involve the optimization task.
The ME-restoration approach is nonlinear. As an advantage to the methods mentioned above it has the constraint of positiveness. Restored pictures are obtained by iteration. To get the incremental picture improvement we use an efficient search strategy in every iteration step /2/.

3. OBVIOUS DIFFERENCES

In this part we give a condensed presentation of obvious properties of the methods described above. The benchmarks used in Table 1 were cho-

method	computing costs	required apriori knowledge	preprocessing demands	number of parameter tests
PSWI	100	none	none	2-3
WISA	140	shape of the ideal spectrum	shape vector evaluation	4-6
WINO	170	shape of I and R	evaluation of 2 shape vectors	4-6
CANO	110	none	none	3-4
ME	$>10^3$	χ^2-statistics	from cheap to costly	2-3

Table 1 Some differences between the methods

sen from a practical point of view. With regard to fast Fourier Transform algorithms the computational load is nowadays not so important any longer.
Today other aspects such as time-consuming preprocessing demands or the necessary gaining of reliable apriori knowledge move into the foreground.
Another important question is how suboptimal filter parameters can be found effortless in or der to achieve reasonable restoration results.
Finally, let us make some remarks about the table. The computing cost is given in %, where 100% represent the computational effort necessary for one PSWI restoration. The rough numbers are based on pictures with the format N = 512x512 .
The demanded apriori knowledge concerns information about noise, variances, ideal spectrum.
The optical transfer function H is assumed to be known, of course.
The last column of the table gives the average number of necessary filter tests in order to get suitable filter parameters.

4. EXPERIMENTAL COMPARISON

Using a variety of synthetic pictures as well as operational data we have exhaustively tested the four linear methods and the ME-algorithm. Some of the results are presented.
In all cases we assume a convolutional blur and additive noise. In the test data we use two-sided motion blur or defocussion blur as image degradations. For noise we apply a uniform random noise. The Signal-to-Noise ratio S/N is characterized by the maximum relative error given in % .

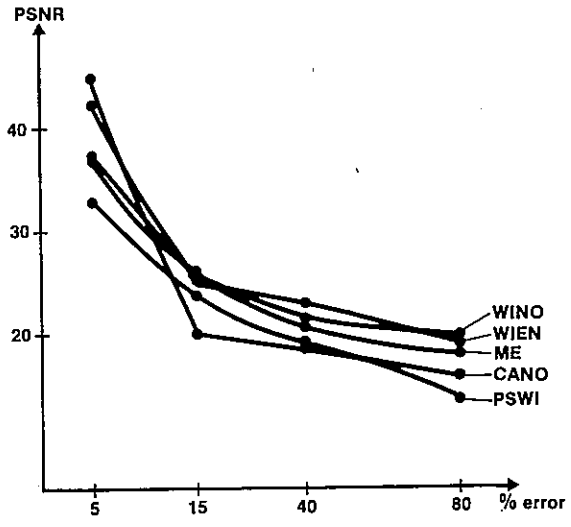


Fig. 1 PSNR of the methods in dependence of different S/N

4.1 Comparison with different S/N

We start with blurred text information. Next to the restoration the contrast of the pictures is enhanced by an appropriate video-lookup transformation (VLT).
The behaviour of the particular methods using different S/N is shown in Fig. 1. As quality measure between the undisturbed picture f and the reconstruction $\hat{f}$ we use the signal-to-noise ratio PSNR with

$$\text{PSNR} = -10 \lg \sum_{j=1}^{N} (f_j - \hat{f}_j)^2 / N^2 M^2 \quad ,M=255 \quad .$$

We study the behaviour of the methods with test pictures where sharp structures were embedded in a smooth changing neigbourhood. We blur the pictures by a Bessel filter and add noise with different maximum error. Fig. 2 shows that the sharp edge is satisfactory reconstructed. But we recognize spurious oscillations in the regions with slowly changing grey levels. This is even true fo the nonlinear method ME.

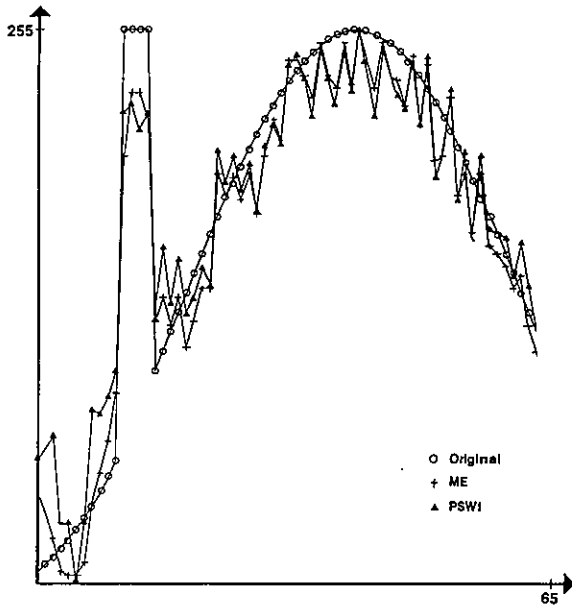


Fig. 2 PSWI and ME reconstruction in a smoothed grey level region with noise (5% maximum error)

4.2 Comparison at presence of noise and bursts of different intensity

If in addition to the noise we add bursts of different intensity we get significant better results with ME restoration than with the other methods. This can be seen in Fig. 3. The linear methods are not able to recover the bursty regions whereas the ME method succeeds in substituting the missing information. This is achieved by assigning a fixed great number to the burst points in the σ^2-error field .

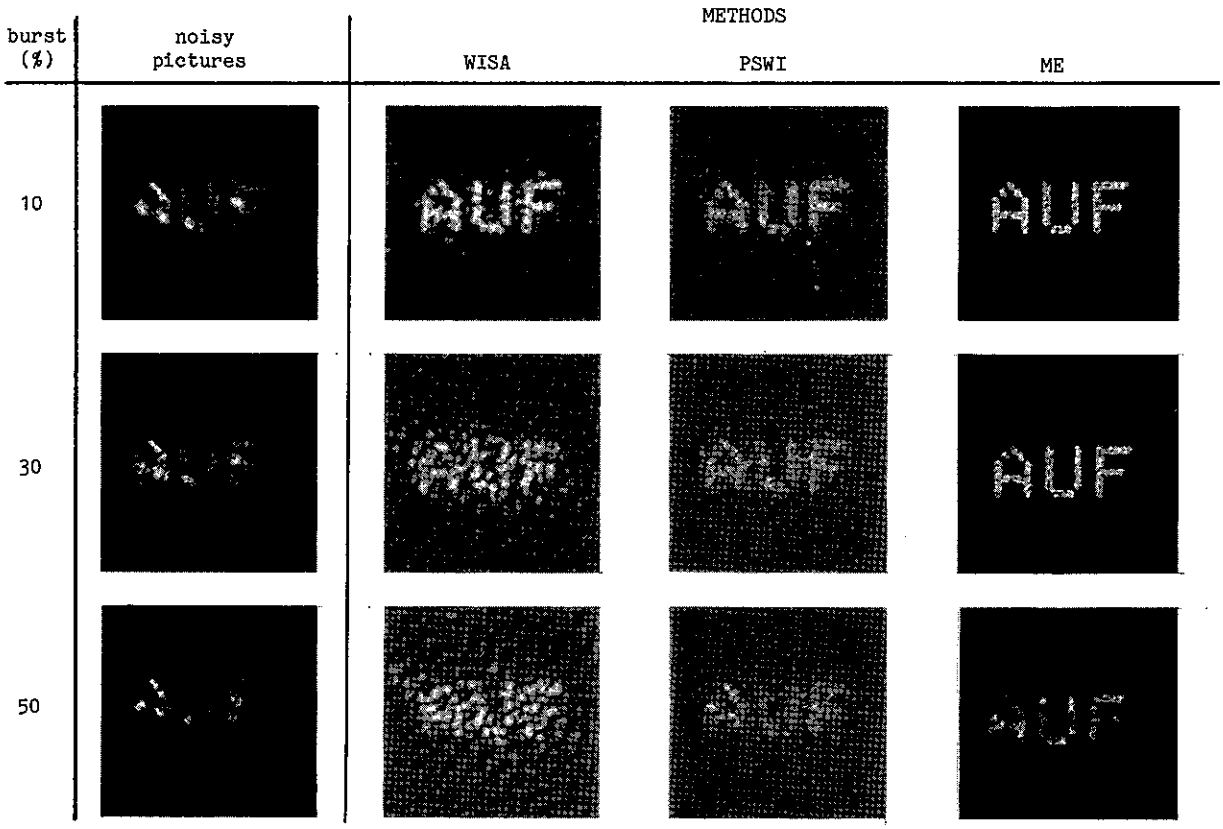


Fig. 3 Bursty pictures with additional noise (15% maximum relative error)

5. CONCLUSIONS

Some experiences with the methods examined can be summarized as follows:

- At the presence of additive noise no significant superiority of the ME-algorithm over the other methods could be realized. This is true, even if S/N is very low.
- In noisy smooth grey level regions all restoration methods - the linear as well as nonlinear methods - produced spurious oscillations.
- Using only a plain Gauss shaped function V the WISA restorations were only slightly better than the restorations of PSWI.
- The added expense of WINO did not produce any visible advantage to the other methods. Only if there is a distinct noise shape one may expect a positive effect.
- The results with CANO-filtering got significantly worse when S/N was low (> 10% maximum error).
- For the ME method we needed 20 iterations on the average. It cannot be assumed that the method converges to the best restoration result. So we have to look at every picture computed in the iteration sequence.
- The ME method was superior to the linear methods if there were bursts in the blurred picture. But the amount of burst must not exceed 50% in order to expect a reasonable restoration effect.
- Our tests did not indicate that the nonlinearity and the constraint of positveness of the ME method will produce remarkably better results when applied to operational pictures. The additional expenditure for ME seems to be justified only in those cases where we have bursts in the blurred picture.

REFERENCES

/1/ FRIEDEN,B.R.: Image enhancement and restoration; in Picture Processing and Digital Filtering; Ed. T.S. Huang, Springer Verlag, Berlin, Heidelberg, New York 1979, p 177-248.

/2/ BURCH,S.F.;GULL,S.F.;SKILLING,J.: Image restoration by a powerfull maximum entropy method; Computer Vision, Graphics and Image Processing; Vol. 23, 1983, p 113-128.

/3/ CANNON,T.M.;TRUSSEL,H.J.: Applications of digital image restoration to photographic evidence; 1980 Carnahan Conference on Crime Countermeasures, Proceedings, p 103-107.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Adaptive Maximum Entropy Coding

N. Merhav and D. Malah
Electrical Engineering Department
Technion - Israel Institute of Technology
Technion City, Haifa 32000, Israel

Abstract

In this paper we analyze and examine a recently proposed waveform coding scheme based on maximizing the entropy of the transmitted bit-stream. The theoretical motivation for using this scheme is the fact that maximum entropy is a necessary condition for optimality of any coding scheme. A practical motivation is its simplicity and amenability to fast implementation. For stationary signals, a detailed analysis of the coder/decoder characteristics is presented. For non-stationary signals we propose an adaptation scheme which tracks slow temporal variations of some statistical parameters. A gain adaptation mechanism cancels the idle channel noise which cannot be removed by an ordinary A.G.C. The new adaptive system is found to overperform ADPCM, particularly for not too - highly correlated ($\rho < 0.8$) non-stationary Gaussian processes.

1. Introduction

In this paper, a new adaptive predictive waveform coding scheme is developed and examined. The scheme is based upon maximizing the first order entropy of the transmitted bit stream. This concept is proposed by E. Angel and L. Daigle in [1], who presented some results of a non-adaptive version of the system (for 1 and 2 bits/sample), for coding speech signals [2] and images [1]. These authors assume the input signal to be a stationary Gaussian process with a known covariance function. For this class of signals we found their proposed scheme to over perform DPCM significantly in a wide range of the correlation coefficient value. However, for non-stationary signals, the fixed scheme is not suitable. Moreover, low energy regions of large dynamic range signals, are reconstructed with very high level of idle channel noise. The non-linear nature and the implicit dependence of the encoder characteristics upon the input signal gain and correlation coefficient value, causes considerable difficulties in the adaptation task. Nevertheless, in this paper we propose approximations of these characteristics by explicit functions of the gain and the correlation coefficient, which enable adaptation to slow variations of these parameters. The suggested gain adaptation algorithm cancels the idle channel noise, which can not be removed by an ordinary A.G.C. However, the computational complexity required for adapting the correlation coefficient is greater than in classical ADPCM. The proposed adaptive predictive maximum entropy system is particularly suitable for coding non-stationary Gaussian processes with slowly varying covariance functions. For speech signals, the resulting quality and intelligibility are equivalent to CVSD for the 1 bit/sample version and to ADPCM for the 2 bit/sample system. In addition to simulation results, the presentation of an adaptation scheme, and the method for cancelling the limit cycles at low input signal amplitudes, an important contribution of this paper is a detailed analysis and presentation of the coding and decoding characteristics, not given previously in [1,2].

2. The Maximum Entropy Concept

The maximum entropy (ME) criterion has been proposed by Angel and Daigle [1,2]. As we see it, the motivation for selecting this criterion, is the fact that maximum entropy is a necessary attribute of an optimum data compression system. This follows from a simple consideration: Suppose we had such an optimum system having entropy less than its rate (in bits/symbol), then one could further compress the data with no additional distortion (by entropy coding), and consequently reduce the rate. It follows that the original rate was not minimum for the given allowed distortion. From the convexity property of the rate-distortion function, it follows that the distortion was not minimum for that rate, in contrast to the above assumption.

It can be shown [5] that for conventional waveform coding schemes such as DPCM, there are considerable difficulties in optimizing the parameters (in the MMSE sense) for low rates, because the true equations for solving the optimum predictor coefficients are highly non-linear. On the other hand, since it satisfies a necessary condition for optimality, the ME approach has the potential of obtaining an improved solution compared to a solution based on linearizing the non-linear equations.

3. System Description

In this section we review the scheme proposed by Angle and Daigle [1,2]. In order to obtain a convenient analytic solution, these authors [1,2] concentrate on the maximization of the conditional first order entropy and assume that the source is a Gaussian process with a known covariance function. Fig.1 depicts the transmitter and receiver for the rate of 1 bit/sample. The transmitter (a) is quite similar to the first order DPCM transmitter. However, in contrast to the DPCM transmitter, which is designed to minimize the energy of the residual, the suggested predictor (also 1st order), is designed in such a way that the output symbols are equally likely, given the information currently available to the predictor, i.e., e_n and y_n. In other

words, the following condition is to be satisfied:

$$\Pr\{e_{n+1}=1\,|\,e_n,y_n\}=\Pr\{e_{n+1}=0\,|\,e_n,y_n\}=\frac{1}{2} \quad (1)$$

In this way the maximum conditional first order entropy is ensured and consequently, statistical independence between successive bits. The receiver (b) consists of a similar predictor (in order to reconstruct y_n), and in addition an estimator of the input ($\hat{x}_n$) which utilizes the information currently available at the receiver; namely, e_n, y_n, e_{n-1} and y_{n-1}:

$$\hat{x}_n = E\{x_n\,|\,e_n,e_{n-1},y_n,y_{n-1}\} \quad (2)$$

where $E\{(\cdot)\}$ denotes the expectation operator. Since y_n is determined from e_{n-1} and y_{n-1} by the predictor, it can be omitted and (2) can be rewritten also as:

$$\hat{x}_n = E\{x_n\,|\,e_n,e_{n-1},y_{n-1}\} \quad (3)$$

Since the prediction and the estimation functions are implicit (as we shall see later), it is proposed in [1,2] to store them in look-up tables (LUT's). This way there is no computational load at all but only memory accesses. Consequently, a very fast system can be implemented with quite modest memory requirements, and can handle high sampling rates as required, for example, in video processing.

The performance of the above scheme for image compression is described in [1] and the results for speech signals are given in [2], both in comparison with DPCM (non-adaptive), for 1 bit/sample and 2 bits/sample. No adaptive version of the above scheme is proposed in [1,2].

4. Prediction and Estimation Characteristics

In this section we describe in detail the prediction and estimation characteristics for a 1 bit/sample system. In the sequel we consider also the 2 bits/sample system.

Let $\{x_n\}$ be a zero mean Gaussian process with variance σ^2 and correlation coefficient $\rho \triangleq E(x_n x_{n+1})/\sigma^2$, and assume that these parameters are known.

4.1 The predictor

In order to ensure equal probabilities for the quantization levels at the transmitter, the predictor: $y_{n+1}=M(y_n,e_n)$ must be the median of the conditional density function $g(x_{n+1}|x_n \geq y_n)$ for $e_n=1$, or the median of the density function $g(x_{n+1}|x_n < y_n)$ for $e_n=0$. For the case $e_n=1$, it follows from the Bayes formula that:

$$g(x_{n+1}|x_n \geq y_n)=\frac{g(x_{n+1},x_n \geq y_n)}{P(x_n \geq y_n)} \quad (4)$$

The denominator of (4) is given by $Q(\frac{y_n}{\sigma})$, where

$$Q(x) \triangleq \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-t^2/2}\,dt \quad . \quad (5)$$

For the numerator of (4) we have:

$$g(x_{n+1},x_n \geq y_n)=\int_{y_n}^{\infty} g(x_{n+1},\vartheta)\,d\vartheta =$$

$$= g(x_{n+1})\int_{y_n}^{\infty}\frac{1}{\sqrt{2\pi(1-\rho^2)}\sigma}\exp\left[-\frac{(\vartheta-\rho x_{n+1})^2}{2\sigma^2(1-\rho^2)}\right]d\vartheta =$$

$$= \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{x_{n+1}^2}{2\sigma^2}\right)Q\left(\frac{y_n-\rho x_{n+1}}{\sigma\sqrt{1-\rho^2}}\right) \quad (6)$$

By putting (6) into (4) and integrating with respect to x_{n+1} from y_{n+1} to infinity, we obtain an expression for $Pr\{e_{n+1}=1\,|\,e_n,y_n\}$. Now, by (1) we have the following equation:

$$\Pr\{x_{n+1} \geq y_{n+1}\,|\,x_n \geq y_n\} \equiv$$

$$[\sqrt{2\pi}\sigma Q(\frac{y_n}{\sigma})]^{-1}\int_{y_{n+1}}^{\infty} e^{-\vartheta^2/2\sigma^2}Q\left[\frac{y_n-\rho\vartheta}{\sigma\sqrt{1-\rho^2}}\right]d\vartheta=\frac{1}{2} \quad (7)$$

In expression (7) the variables ρ,σ and y_n can be viewed as parameters and y_{n+1} can be viewed as the unknown. This equation can be solved by numeric techniques. For the case $e_n=0$ a similar equation is obtained by using symmetry considerations. The resulting predictor is non-linear as is demonstrated in Fig.2.

4.2 The Estimator

The estimator reconstructs the input by assigning the appropriate representation levels for the predictor's quantization decision levels. These levels are the centroids [6] of the ranges of the input samples given the information $\{e_n,y_{n-1},e_{n-1}\}$. For the case $e_n=e_{n-1}=1$ it can be shown (using integration by parts) that (3) satisfies:

$$\hat{x}_n=\sigma\sqrt{\frac{2}{\pi}}\frac{1}{Q(\frac{y_{n-1}}{\sigma})}\left\{e^{-y_n^2/2\sigma^2}Q\left[\frac{y_{n-1}-\rho y_n}{\sigma\sqrt{1-\rho^2}}\right]+\right.$$

$$\left.+\rho\,e^{-y_{n-1}^2/2\sigma^2}Q\left[\frac{y_n-\rho y_{n-1}}{\sigma\sqrt{1-\rho^2}}\right]\right\} \quad (8)$$

For other values of e_n and e_{n-1}, similar expressions are obtained by using symmetry properties of the Gaussian distribution.

Because of practical limitations explained below, a 2 bits/sample system is not obtained just as a simple extension of the 1 bit/sample system, since there are now 3 threshold values. Here, if each threshold is represented by 8 bits, and the error is quantized to 2 bits, then the predictor LUT is addressed by 26 bits. Consequently, the predictor LUT size needed is 192 Mbyte ! Clearly, one must limit the number of states. This could be done by dividing the support of the density functions $g(x_{n+1}|x_n \geq y_n)$ and $g(x_{n+1}|x_n < y_n)$ into four non-overlapping intervals, each having a probability of 1/4. This solution is of course suboptimal.

5. System Performance for Stationary processes

In this section we present some simulation results performed to measure the ME performance for stationary Gaussian processes at the rates of 1 and 2 bits/sample.

We now examine the dependence of the SNR on the value of the correlation coefficient - ρ. The SNR is defined as

$$\text{SNR} \triangleq 10\log_{10}[(\sum_{n=1}^{Nk}x_n^2)/(\sum_{n=1}^{Nk}(x_n-\hat{x}_n)^2)] \quad (9)$$

where N denotes the number of points per sequence and k - the number of sequences. The values of ρ used were $\rho=0.2, 0.5, 0.8, 0.9, 0.95, 0.98$. For each of these values, $k=50$ Gauss-Markov sequences were produced. Each sequence had $N=4096$ points. In addition, for each value of ρ the prediction and estimation LUT's were computed. The upper bound for the SNR of the reconstruction of a 1st order Gauss-Markov process from any representation by R bits/sample can be easily obtained from the rate-distortion function [3].

$$\text{SNR}[dB] \leq 6.02R-10\log_{10}(1-\rho^2) \quad (10)$$

In Fig.3 several graphs of SNR vs. ρ are presented. The figure compares the performances of ME, DPCM and the upper bound provided by (10) for the above rates.

The stepsizes for DPCM and LDM (Linear Delta Modulation) were selected empirically to minimize the MSE [5].

Several inferences are drawn from these curves:

1. The performances of all the systems examined are considerably far from the upper bound.
2. For a wide range of ρ, the ME system significantly overperforms DPCM and LDM, particularly at the rate of 1 bit/sample.

It is seen from Fig.3 that for a wide range of the correlation coefficient (zero to 0.8 or 0.85) the ME system has a higher SNR, by up to 5dB than LDM at 1 bit/sample, and about 3 dB above DPCM for 2 bits/sample.

6. The Adaptive Scheme

In order to make the system described earlier adaptive, one needs to estimate ρ and σ at each time instant, and to update the predictor and the estimator, accordingly, at both transmitter and receiver.

6.1 Gain Adaptation

It is suggested to estimate the parameter σ as follows:

$$S_n = \lambda S_{n-1} + (1-\lambda)\tilde{x}_n^2 \qquad (0 \le \lambda < 1) \qquad (11)$$

where: $\hat{\sigma}_n = \sqrt{S_n}$ is the estimate of σ at time n, $\tilde{x}_n$ - the reconstructed "normalized" signal (see Fig.4) and λ - a decay which determines the speed of adaptation.

In this way, the variance of the "normalized" signal remains roughly constant. At the receiver, $\tilde{x}_n$ is multiplied by σ_n ($\hat{x}_n = \tilde{x}_n \cdot \sigma_n$). The main problem observed in using this AGC mechanism, is a limit cycle effect which occurs when the input signal has a low energy. The limit cycle is characterized by high amplitude oscillations in the reconstructed waveform. Since the gain is adapted using the *reconstructed* signal, it turns out that the gain (σ_n) does not decay sufficiently in low energy intervals, and these oscillations remain large. To overcome the limit cycle problem it is first necessary to identify this event and to force the variable S_n to decrease, regardless of the value of the reconstructed signal. We have therefore modified (11) as follows:

$$S_n = \begin{cases} \lambda_1 S_{n-1} + (1-\lambda_1)\tilde{x}_n^2 & \text{, no limit cycle exists} \\ \lambda_2 S_{n-1} & \text{, limit cycle exists} \end{cases} \qquad (12)$$

where λ_1 and λ_2 are positive numbers smaller than 1. The identification of a limit cycle occurrence is based on the alternating sign of the prediction variable y_n. A limit cycle event is declared whenever the sign of y_n alternates at least three times successively. This mechanism was found to remove completely the limit cycle effect.

6.2 Adaptation of the Correlation Coefficient (ρ)

To adapt the system to variations in the correlation coefficient ρ, it is suggested to estimate ρ in the following way:

$$C_n = \lambda C_{n-1} + \tilde{x}_n \tilde{x}_{n-1} \qquad (13a)$$

$$S_n = \lambda S_{n-1} + \tilde{x}_n^2 \qquad (13b)$$

$$\hat{\rho}_n = C_n / S_n \qquad (13c)$$

where λ is the "forgetting" factor ($0 < \lambda \le 1$). It is necessary to limit the values of $\hat{\rho}_n$ such that $|\hat{\rho}_n|$ would not exceed unity.

The main problem now is how to use this estimate $(\hat{\rho}_n)$ to update the predictor and the estimator. We have seen (expression (8)) that the estimate $\hat{x}_n$ can be expressed "explicitly" in terms of ρ, σ, y_n, e_n, e_{n-1} and y_{n-1}, where y_n is related to y_{n-1} and e_{n-1} by the prediction function. However, for the predictor we do not have an explicit formula. Therefore, it is proposed to use a simple approximation of the predictor by an explicit function. This is easily done by defining the predictor as the conditional expected value (instead of the median value), e.g., for $e_n = 1$, we have:

$$y_{n+1} \cong E(x_{n+1} | x_n \ge y_n) = \frac{\rho\sigma}{\sqrt{2\pi}} \frac{\exp\left(-\frac{y_n^2}{2\sigma^2}\right)}{Q\left(\frac{y_n}{\sigma}\right)} \qquad (14)$$

This approximation turns out to be a very good one (particularly for large values of $|y_n|$). Simulation results did not reveal any significant differences between using the exact predictor or the above approximated predictor. Similar ideas can be used for the 2 bits/sample system [7]. We now have "explicit" formulas for both the prediction and estimation by which updated values of ρ and σ can easily be substituted.

The computational load required to adapt the system is heavier than in classical ADPCM because these formulas are quite complicated. A lookup table for the function $Q(\cdot)$ is needed as well. But, as it was shown above, if the input is not too highly correlated, that adaptive version of the ME method overperforms ADPCM for non-stationary Gaussian processes with slowly varying covariance functions. For speech signals, the adaptive ME scheme turns out to perform equivalently to ADPCM in terms of quality and intelligibility.

References

[1] E. Angel and L. Daigle, "A High Speed Maximum Entropy Encoder for Images", IEEE ICASSP, 1983, pp. 1236-1239.

[2] E. Angel, L. Daigle and M. Rodriguez, "A Maximum Entropy Encoder for Speech", IEEE ICASSP, 1983, pp. 1292-1295.

[3] T. Berger, "Rate Distortion Theory", Prentice-Hall, Cliffs N.J., 1971.

[4] N.S. Jayant and P. Noll, "Digital Coding of Waveforms", Englewood Cliffs, N.J., Prentice-Hall, 1984.

[5] L.R. Rabiner and R.W. Schaffer, "Digital Speech processing", Prentice-Hall Inc. Englewood, Cliffs, N.J., 1978.

[6] Y. Linde, A. Buzo and R.M. Gray, "An Algorithm for Vector Quantization Design", IEEE Trans. on Communication, Vol. COM-28, No.1, Jan. 80, pp. 84-95.

[7] N. Merhav, Adaptive Maximum Entropy Coding of Speech Signals, M.Sc. Dissertation, Technion - I.I.T., Haifa, Israel, Nov. 1985. (In Hebrew).

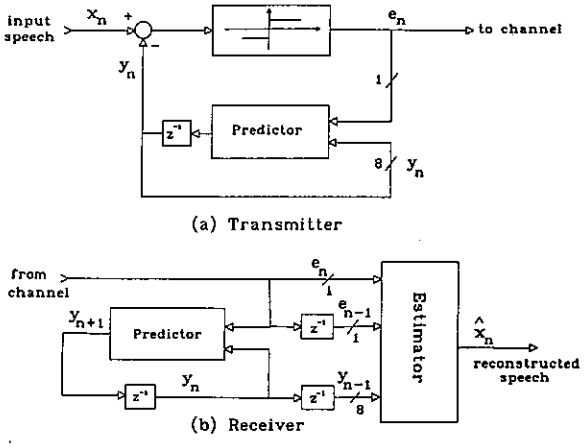


Fig. 1: - The compression scheme proposed in [1,2].

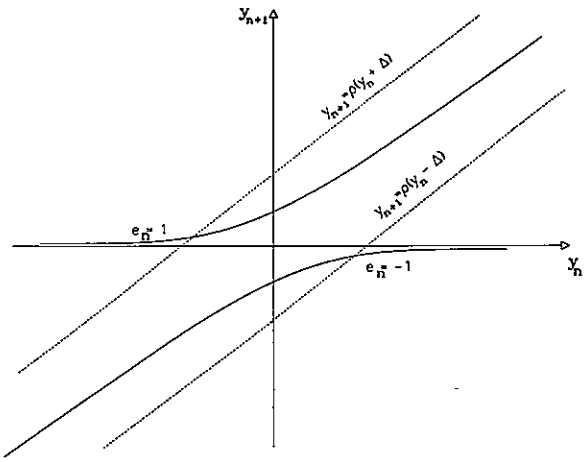


Fig. 2: - Prediction characteristics of ME and LDM (Linear Delta Modulator) for $\rho = 0.9$: solid line - ME predictor, dashed line - LDM predictor

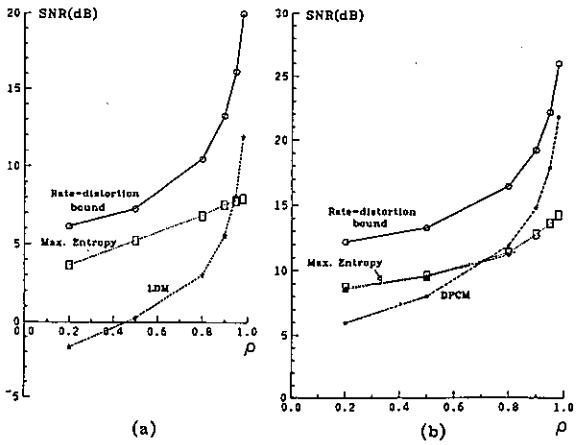


Fig. 3: - SNR versus ρ for the various systems: (a) R = 1 bit/sample (b) R = 2 bits/sample

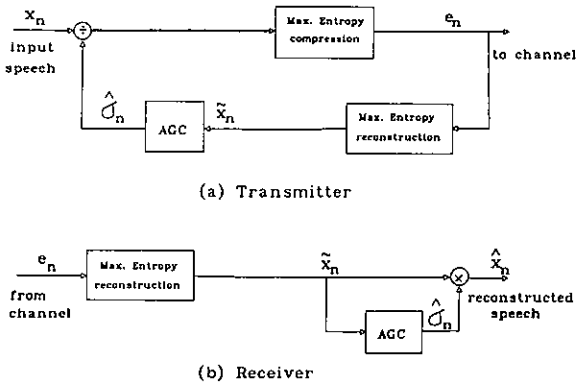


Fig. 4: - Transmitter and receiver with AGC.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

K1.7

A HYBRID IMAGE CODING SCHEME USING ADAPTIVE LOCAL RESOLUTION

H.J. Kirchhoff and Ph. W. Besslich

Section of Electrical Engineering
University of Bremen
D-2800 Bremen 33, F.R. of Germany

This hybrid coding method uses the transform coefficients of 2-D interleaved (subsampled) 4x4 subpictures to predict the three other 4x4 subpictures of an 8x8 (sub-)image. Whether or not the subsampled version is sufficient to represent the whole 8x8 image depends on the subpicture's spatial frequency contents. If necessary, additional information will be added. The scheme provides anisotropical resolution which is adaptively controlled.

1. INTRODUCTION

Data compression of image signals has been persued mainly using either predictive or transform coding [1-4]. To combine the merits of both these schemes a number of hybrid coding methods that employ both, transforms as well as prediction, have been proposed or implemented [5-7]. One of these hybrid coding methods takes a 1-D orthogonal transformation of the lines of an image block and subsequently applies predictive coding to the transform coefficients [6]. Another approach reversely applies DPCM to image data of a line and subsequently employs transform coding to the prediction error signal. The two methods have been shown to perform equally well [7], if applied appropriately.Both these hybrid intraframe coding schemes employ either prediction or transform coding to one spatial direction and the other coding method to the other direction.If interframe coding is the aim, the temporal direction may be exploited for DPCM coding, while the two spatial directions are 2-D transform coded [8].

Disadvantages of predictive coding is the use of only a few pixels in the prediction process. This fact, in connection with a sampling rate according to the highest spatial frequency, accounts for only a moderate compression to about 3 bits/pixel for non-adaptive DPCM. Transform coding on the other hand, more or less decorrelates the coefficients of larger image blocks. The sum of their variances remains constant, but for subpictures of very little detail information the total variance is contained in only a few significant coefficients. Hence, an efficient data reduction may be achieved. If, however, the subpicture contains a large amount of detail information, nearly all the coefficients become significant and must be retained to avoid low pass filter effects. Hence, for "busy" subpictures transform coding cannot be expected to be very effective. On average, non-adaptive transform coding requires 1.5 to 2 bits/pixel.

The reason for these limitations is the non-stationary nature of most images. In order to cope with this phenomenon certain qualities in the coding process may be controlled adaptively, for instance, by the spatial frequency content of the image region to be coded. Transform coding usually employs adaptive bit allocation to the coefficients, while prediction coefficients or/and quantization characteristic are controlled adaptively in DPCM. Data rates of adaptive DPCM are 1 to 1.5 bits/pixel for good quality pictures, while adaptive transform coders achieve 1 bit/pixel or even less.
The new hybrid coding scheme described here uses transform coefficients of an interleaved (subsampled) 2-D image block to predict the coefficients of a larger subimage the size (local resolution) of which is adaptively controlled. The scheme improves the efficiency of transform coding mainly for "busy" subpictures. To achieve this property, the merits of both transform and predictive coding are applied to an originally uniformly sampled image.

2. THEORETICAL MODEL

A rough outline of the coding procedure is given at the outset. Consider an image subdivided into 8x8 non-overlapping subpictures. Let the 64 pixels be rearranged into 4 interleaved 4x4 arrays formed by the pixels a_i, b_i, c_i and d_i, i=0,1,2,...,15, (cf. Fig. 2). Each of the subarrays corresponds to a subpicture sampled with half the original sampling rate in both directions. The 4 subarrays contain approximately the same low spatial frequency components. Each of these subsampled arrays undergoes a 2-D orthogonal transformation. Let the set of transform coefficients of blocks a_i, b_i, c_i and d_i be denoted by T_{ai}, T_{bi}, T_{ci} and T_{di}, respectively. The aim is now to retain only the coefficients of array a_i if the spatial frequency content allows to do so. Otherwise, the necessary detail information is added to the signal. For this purpose we calcu-

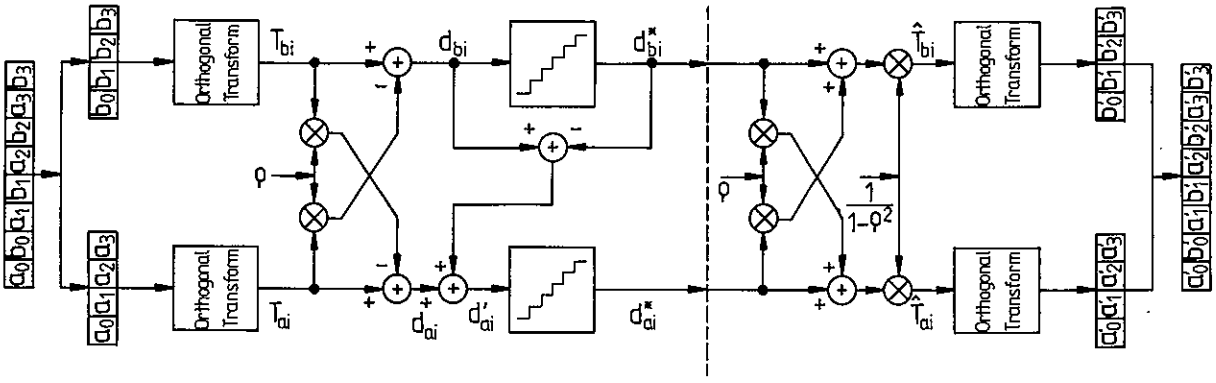


Figure 1

late the (weighted) difference of corresponding coefficients of the sub-blocks. If the correlation in the picture proves to be anisotropic, transmission of only one of these differences may be sufficient. Higher frequency coefficients are added to those subpictures only in which they are needed.

The general philosophy of the new hybrid coding scheme needs some more detailed consideration. Theoretically a signal source (image) that obeys a first order Markoff process can be completely decorrelated by a predictor of first order [9]. For real images, however, this model is applicable to the global image only. Locally, the Markoff model does not hold. Hence, the performance of coders based on this model often falls short of theoretical expectations. The new hybrid coding using adaptive local resolution is designed to cope with this situation.

To simplify the development of the coding procedure we will first explain it for the case of 1-D signals. Note, however, that the full advantage of the method becomes discernible only for the 2-D signals, e.g. for signals with anisotropic correlation. Let a 1-D signal consist of 8x1 pixels (a_0 b_0 a_1 b_1 a_2 b_2 a_3 b_3) as shown in Fig. 1. Each block is then divided into 2 sub-blocks (a_0 a_1 a_2 a_3) and (b_0 b_1 b_2 b_3), each one being a subsampled version of the signal. The 2 sub-blocks undergo now an orthogonal transformation (e.g. WHT or DCT). Predictive coding is then applied to the transform coefficients T_{ai} and T_{bi}, i=0,1,2,3, using the T_{ai} as a prediction value (estimate) of the T_{bi}

$$T_{ai} = \rho\, T_{bi} + d_{ai} \quad ,$$

and vice versa

$$T_{bi} = \rho\, T_{ai} + d_{bi}$$

where ρ is the correlation coefficient of the assumed Markoff model and the d_i are the prediction errors. Since the blocks a_i and b_i are shifted by exactly 1 pixel, the corresponding transform coefficients T_{ai} and T_{bi} form first order Markoff processes as well. In contrast to first order prediction, the set of transform coefficients may be made large enough to justify the assumption of first order Markoff signals. Hence, the variances of the differences d_{ai} and d_{bi} will be smaller than in the case of spatial prediction. The difference signals are quantized by the quantizers Q_a and Q_b. In order to avoid accumulation of quantization errors an error feedback is provided. Because Q_a is to quantize the sum of the difference signal d_a and the quantization error q_b, it needs more bits than Q_b to minimize the overall quantization error. For reconstruction the whole process is reversed (cf. Fig. 1). Note that the differences d_{ai} and d_{bi} account for a de-emphasis of low spatial frequencies that are contained in both the blocks a_i and b_i. In other words, the process of prediction causes an emphasis of high spatial frequencies. This process causes a more uniform distribution of spatial frequency components in the prediction error signal, and consequently, minimizes the correlation of its samples. If the blocks contain only very little detail information, it suffices to retain d_{ai}, because $d_{bi} \approx d_{ai}$. The decision whether to transmit d_{ai} and d_{bi} or else d_{ai} only is governed by the sum of squared transform coefficients. If this sum is below a threshold, only d_{ai} is transmitted. This process may also be viewed as an adaptive matching of spatial sampling, i.e. as a spatially variable resolution within the signal.

Actually this hybrid coding exploits redundancies of 2-D signals. Fig. 2 illustrates how an 8x8 subimage is subdivided into the 4 blocks a_i, b_i, c_i and d_i, (i=0,1,2,...,15). A 2-D orthogonal transform of each of the blocks is taken to obtain 4 sets of 16 transform coefficients T_a, T_b, T_c and T_d. Corresponding transform coefficients from the 4 sets serve as estimates in the 2-D prediction process. The use of two prediction coefficients ρ_v and ρ_h

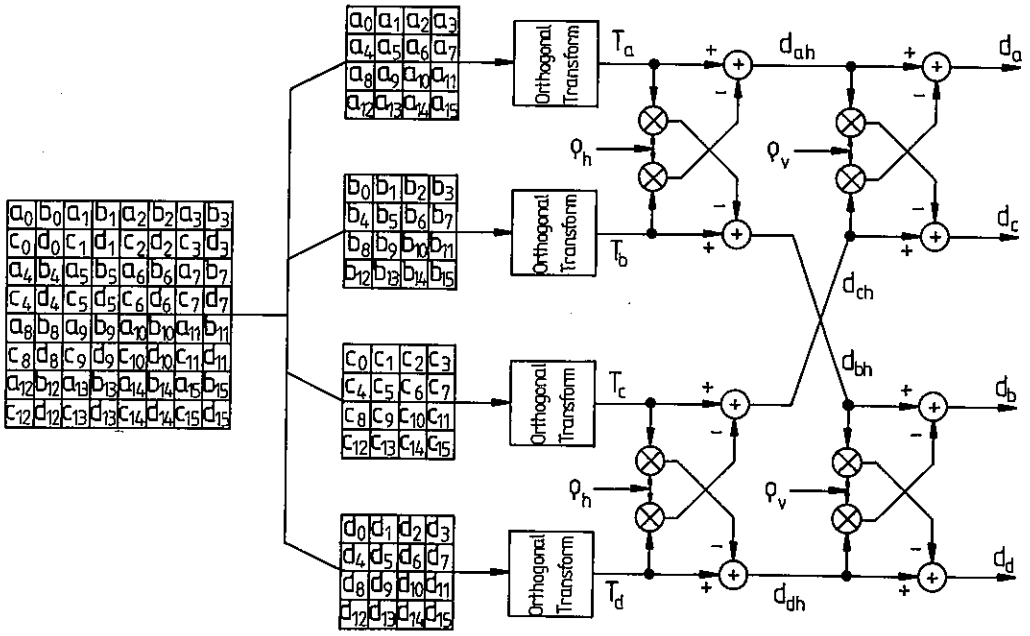


Figure 2

permits the system to take care of different correlation in vertical and in horizontal direction, respectively. Each element in the 4 sets of difference signals d_a, d_b, d_c and d_d uses the corresponding elements of vertically or horizontally neighboring elements for its prediction. Depending on whether the spatial frequency contents of the difference signals exceeds certain thresholds, only one set of difference signals (d_d) need be quantized and transmitted if the subimage is highly correlated in both directions. Alternatively d_a and d_b (or d_a and d_c) may be used if correlation is high in vertical or horizontal direction only. Finally all 4 sets may be needed for reconstruction of a subimage that is highly busy in all directions.

3. SIMULATION RESULTS AND CONCLUSION

The new hybrid coding scheme uses adaptively controlled resolution for both the spatial directions, and the associated overhead required is only 3/64 bits/pixel. Because the correlation in the picture proves to be anisotropic, the predictive coding applied to the transform coefficients improves the coding gain.

Simulation results with a data rate of 1 bit/pixel show pictures of excellent subjective impression. Signal-to-noise ratio for a test picture is about 20 dB, if based on variances of the reconstructed signal and the mean square error. Alternatively,it is 31 dB if based on the square of the maximum signal amplitude and the mean square error.

Fig. 3(a) shows the original image (512x512; 8 bits/pixel). Fig. 3(b) is the reconstructed image with only 1 bit/pixel. The different gray levels in Fig. 3(c) highlight the orientations of the image structure in different parts of the original picture (cf. Fig. 3(d)). The light gray level corresponds to regions of high correlation in horizontal direction. Similarly the dark gray level corresponds to that in vertical direction. The white and black regions, respectively, point to low and high correlations in both the directions.Thus, to reconstruct the image, 1, 2 or 4 sets of difference signals need to be transmitted depending on whether the region is black,gray or white.

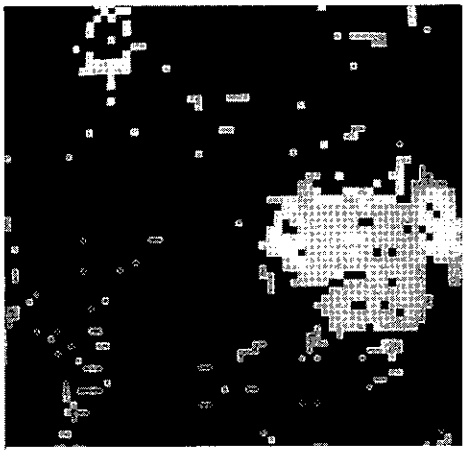
Preliminary studies have indicated that the scheme offers a solution for further reduction of bit rate. More detailed studies in this direction are under way.

The examples shown in Fig. 3 have been obtained using the 2-D Walsh-Hadamard transformation. Experiments using the cosine transformation indicate slightly improved performance.

REFERENCES

[1] Musmann, H.G., Predictive Image Coding, in: Pratt W.K. (ed.), Image Transmission Techniques (Academic Press,New York, 1979), pp. 73-112.
[2] Musmann, H.G., Pirsch, P. and Grallert, H.-J., Advances in picture coding, Proc. IEEE, vol. 73 (1985), pp. 523-548.
[3] Tescher, A.G., Transform Image Coding, in: Pratt W.K. (ed.), Image Transmission Techniques (Academic Press, New York, 1979), pp. 113-155.
[4] Clarke,R.J., Transform Coding of Images (Academic Press, London, 1985).
[5] Roese, J.A., Hybrid Transform/Predictive Image Coding, in: Pratt W.K. (ed.),Image Transmission Techniques (Academic Press, New York, 1979), pp. 157-187.
[6] Habibi, A., An adaptive strategy for hybrid image coding, IEEE Trans., COM-29 (1981), pp. 1736-1740.
[7] Ericsson,S., Fixed and adaptive predictors for hybrid predictive/transform coding, IEEE Trans., COM-33 (1985), pp.1291-1302.
[8] Kamangar, F.A. and Rao, K.R., Interfield hybrid coding of component color television signals, IEEE Trans., COM-29 (1981), pp. 1740-1753.
[9] Jayant, N.S. and Noll, P., Digital Coding of Waveforms (Prentice-Hall, Englewood Cliffs, N.J. 1984).

(a) Original 8 bit/pixel



(c) Areas of high (black), low (white) and anisotropic (gray) correlation

(b) Reconstruction 1 bit/pixel

(d) Reconstruction 1 bit/pixel

Figure 3

Tutorial on IMAGE FILTERING

G.H. Granlund
Linköping University
Dept. of Electrical Engineering
Linköping
Sweden

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW CONCEPT TO ENCODE THE OVERHEAD INFORMATION OF THRESHOLD TRANSFORM CODING SYSTEMS USING REPRESENTATIVE ROOT PATTERNS

U. Franke, R. Mester

Inst. f. Elektr. Nachrichtentechnik
RWTH Aachen
Melatener Str. 23
5100 Aachen, West-Germany

In image processing, threshold based transform coding systems enable very high data compression ratios. They require a proper method to inform the decoder which coefficients are transmitted for every particular subblock. A new classification scheme is described which can be used for a large class of threshold coding schemes and is easy implementable in hardware. Its superior performance compared to the usually used zigzag scan approach is demonstrated.

1. INTRODUCTION

Transform coding is one of the most important and efficient methods to code grey level as well as colour images. Adaptive coding systems, in particular, enable reconstructions with high subjective quality even at very low bit rates. Beside zonal coding schemes, threshold coding schemes are used increasingly due to their high data compression capabilities.

The basic idea of these coding schemes, which are adaptive by nature, is to transmit only those spectral coefficients that are subjectively relevant for the reconstruction of the considered subblock.
Usually, it is assumed that a coefficient is relevant in this sense, if its amplitude lies above a given threshold. Therefore, the selection of appropriate thresholds is important for the coder performance. CHEN /1/ for example uses a threshold which is constant for all coefficients, LOHSCHELLER /2/ and FRANKE /3/ measured visibility thresholds for luminance and chrominance DCT-coefficients, respectively.
A new algorithm to adapt the used thresholds to the contents of the actual subblock in order to achieve a constant subjective image quality is described by the authors in /4/.

The high adaptivity of threshold coding schemes requires a proper method to inform the decoder which coefficients are transmitted for every particular subblock.
For this reason, the procedure used to encode the binary events "coefficient y(i,j) transmitted / supressed" is important for every threshold coding scheme.

If the N*N binary informations are coded independently for each coefficient (blocksize N*N pels), a disproportional number of N*N bits would be necessary.

By this, some of the advantages offered by transforms with energy compaction properties like the discrete cosine transform (DCT) would be given away, since neither

1. the very low probability of occurance of many spectral coefficients nor
2. the statistical dependencies between the occurance of different supra-threshold coefficients

are exploited.

1.2 The Zigzag Scan

Often, a method proposed by TESCHER /5/ is used. The spectral coefficients are scanned along a zigzag path as indicated in figure 1 and a run-length coding scheme using a truncated Huffman code is applied to the runs of consecutive coefficients lying below the chosen threshold(s) in order to encode the adresses of the supra-threshold coefficients.

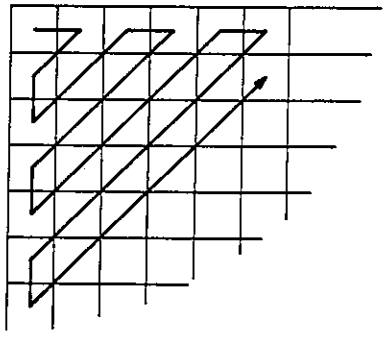


Fig. 1: Zigzag path proposed by Tescher

In addition to the required code bits an 'end of block' code (EOB) and a 'runlength prefix' code is necessary. The EOB code is used to indicate the receiver that the last significant coefficient of the subblock has been transmitted. The runlength prefix code is required to distinguish the runlength code from the amplitude code, since the also Huffman coded amplitudes and the necessary runlength-code are interlaced.

This approach only exploits the statistical deendencies of the coefficients along the zigzag path and neglects all other dependencies between the coefficients of the two-dimensional binary process which can be shown to exist. Furthermore, this scheme only partially takes advantage from the different probabilities of occurance of the different spectral coefficients.

Hence, this approach cannot give satisfaction from an information theoretical point of view and a further reduction of the required overhead information should be possible.

2. THE CONCEPT OF REPRESENTATIVE ROOT PATTERNS

2.1 The Approach

In order to get closer to the boundaries given by information theory, the N*N binary events (compare 1.) have to be regarded as a realization of a two-dimensional binary process and encoded accordingly.

Each of these realizations is a binary pattern of N*N coefficient flags. A flag is set TRUE, if the corresponding coefficient lies above its threshold and set FALSE otherwise. For this reason, these patterns shall be called supra-threshold coefficient patterns (STCPs).

Obviously, it is impossible for common block dimensions of 8*8 or 16*16 pels to generate an appropriate Huffman code table including all possible patterns, even though usually some coefficients are a priori excluded from transmission.

Therefore, a new classification scheme is proposed which overcomes this problem and takes advantage of all statistical dependencies within the considered subblock.
The approach is to define K classes, each led by a specially selected STCP which is called the representative root pattern (RRP). Each STCP has to be assigned to one of these classes according to the following rules:

1. to guarantee the required reconstruction quality, the selected RRP has to include all supra-threshold coefficient flags of the actual block
2. to achieve a minimum data rate, the selected RRP should contain as few coefficients as possible which actually lie below their thresholds.

The decoder has to be informed which RRP has been selected according to these rules. This can be done by transmitting some overhead bits implicitely denoting the coefficient values that will follow.

To illustrate this approach, a binary process of 2*2 elements according to a vector of 4 elements is considered. This leads to 16 possible combinations or binary patterns which are shown in figure 2. Each cross denotes a coefficient flag set TRUE.
In addition, it is assumed in this example that 3 classes are defined given by their RRPs, as indicated in the same figure in the first row.

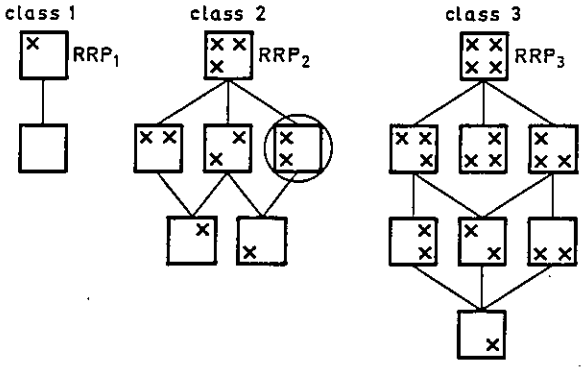


Fig. 2: A possible class assignment for 2x2 element patterns

For example, consider the encirceled pattern. This realization could be assigned to class 2 as well as class 3, if only the first rule would be given. But in order to achieve a low data rate it is obviously preferable to transmit only 3 than 4 coefficients. For this reason, rule 2 has been stated which assigns the considered pattern to class two.

2.2 The Optimization Problem

Within the proposed classification scheme, the total number of overhead bits is composed of two terms:
First, some bits are needed to inform the receiver which class has been chosen.
Second, additional bits are required since within a chosen class generally coefficients are transmitted which lie below their thresholds.
The mean number of additional bits shall be regarded as the costs of the proposed classification scheme.
The mean classification costs depend on:

1. the given RRPs (classification)
2. the costs for unnecessarily transmitted coefficients based on the given bit assignment
3. the probability of occurance of the different STCPs.

The facts mentioned above result in the optimization problem of deciding an appropriate number of representative root patterns and their optimal structures in order to minimize the mean classification costs.

This optimization problem requires a more precise definition of the second classification rule (compare 2.1):
In order to achieve a minimum data rate, the class minimizing the classification costs due to unnecessarily transmitted coefficients has to be selected.

This modified rule is the base for the optimization algorithm described below.

2.3 An Iterative Solution

The attempt to find the optimal RRPs by one of the known search algorithms must fail in practice, because the number of variations searching K classes out of 2^{N*N} possible patterns is nearly infinite for common dimensions of N.

For this reason, an new algorithm is proposed that treats this search as a dynamic problem. Starting with one class, it performs the step from K classes and the corresponding RRPs to K+1 classes by an iterative scheme described in figure 3.

To find a proper classification:

1. - Determine all RRP-candidates
   - Compute the cost table
2. - Define class one (CL_1) including all possible coefficients
3. - DO WHILE more classes are desired:
     - Search for a new class that minimizes the mean classification costs
4. - DO WHILE changes of classification occur:
5. - DO WHILE not all actual classes are considered:
     - Remove class CL_K
     - Search for a new class based on the remaining classification that minimizes the classification costs
     END DO
   END DO
 END DO
STOP

Fig. 3: Proposed iteration scheme

In the initialization step 1,the candidates for RRPs are determined by merging all STCPs found in the test material (performing a logical OR operation with corresponding coefficient flags) and repeatedly merging the resulting patterns until no new patterns are obtained. In order to speed up the iteration of the following steps, a cost table is computed containing the additional mean costs c(l,m) for assigning $STCP_1$ to RRP-candidate m.

The search for RRPs starts with defining class one containing all coefficients under consideration within the used coding scheme. This ensures, that all possible STCPs can be classified without errors, even though they do not occur in the used test material.

In the next step, given K classes a new class is searched which minimizes the mean value of the classification costs.
In step 5 one class is removed and a new one is determined on the base of the remaining K classes repetitivly. This step is repeated until the mean classification costs cannot further be reduced. This means that the best classification has been found using K+1 classes.

If desired, the iteration can continue with step 3 seeking for additional classes.

The proposed algorithm guarantees the decrease of the expectation value of the mean classification costs during each step. Its good behavior will be demonstrated in the next section by an example.
It will appear that the mean classification costs approaches the optimum value of zero with increasing number of classes, even though the proposed algorithm cannot guarantee this.

3. RESULTS

The solution gained by the iteration scheme described above highly depends on the threshold coding scheme under consideration.
The modification of one of the following coder parameters can result in a substantially different classification for a certain number of RRPs:

1. the costs for unnecessarily transmitted coefficients
2. the used thresholds
3. the number of coefficients admitted for transmission.

A luminance coding scheme based on the DCT with block dimension of 8x8 pels shall be taken as an example.

The considered system uses the visibility thresholds introduced by LOHSCHELLER /2/, which are adapted to the spectral sensitivity of the human eye.
Applying these thresholds, no more than 39 low frequent coefficients have to be transmitted in any case.

A variable length coding of the amplitudes of the DCT coefficients is assumed that requires one bit if a coefficient is transmitted unnecessarily.

The coding scheme characterized above has been applied to 12 representative natural images corresponding to about 50.000 realizations of STCPs. Among these patterns 4.860 different ones have been found. Due to memory constraints of the used computer, only the 512 most frequent patterns could be taken into account, still covering more than 86% of all realizations. The effects of this confinement will be discussed below.

Starting with the trivial case of one class, the RRPs corresponding to classifications with up to 27 classes have been determined using the iterative algorithm described in section 2.3. The results of the computations are summarized in figure 4.

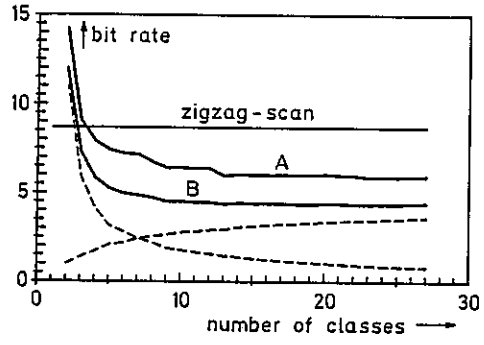


Fig. 4: Overhead bit rates as a function of used classes

The dashed lines represent the number of bits necessary to indicate the chosen RRP (if variable length coding is used) and the classification costs as a function of the total number of defined RRPs, respectively. As can be seen, a fixed length coding of the number of the chosen RRP is possible without significant increase of the mean bit rate.
By addition of both bit rates curve A is obtained, which shows the mean overhead bit rate if the 512 STCPs used for the iteration are classified.
Classification of all patterns found in the test material necessarily increases the bit rate by about 1.5 bit as shown by curve B.

Nevertheless, the proposed classification scheme decreases the overhead bit rate by more than 40% compared to the zigzag scan approach investigated for comparison and exceeds the lower bound given by the estimated entropy of the binary process by only 1.3 bit.

Obviously, from the viewpoint of practical realization only 8-16 RRPs are necessary to classify the STCPs efficiently. Figure 5 shows the 16 RRPs found with the described algorithm.

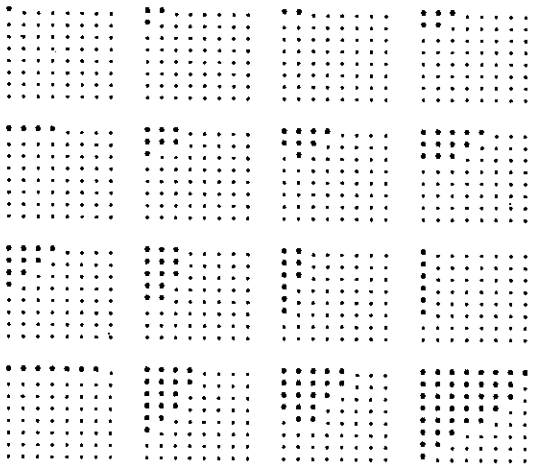


Fig. 5: Resulting RRPs for a classification with 16 classes

4. CONCLUSIONS

A new method has been proposed to inform the decoder of a threshold transform coding scheme which coefficients are transmitted. Its superior performance compared to the usually used zigzag scan approach is demonstrated. In addition to the easy implementation in hardware, the possibility of a transmission of the whole class information in advance of the coefficients offers two further advantages:
First, the compact form of the class information is much better suited to transmission error protection than the run-length approach.
Second, if image transmission with temporally increasing spatial resolution is required, an adaptation of the order of transmission of the coefficients is possible, improving the performance of such coding schemes.

Literature:

/1/ W.-H. Chen: Scene Adaptive Coder, IEEE COM-32, No. 3, March 1984
/2/ H. Lohscheller: Vision Adapted Progressive Image Transmission, Second European Signal Processing Conference EUSIPCO-83. pp. 191-194
/3/ U. Franke: Vision Adapted Transform Coding of the Colour Information of Natural Images, Proc. of the 10th International Symposium on Signal Processing and Applications GRETSI, Nizza 1985, pp. 835-840
/4/ R. Mester, U. Franke, Adaptive Threshold Control in Transform Coding Systems: a New Approach towards Stabilized Reconstruction Quality (in German), to be published
/5/ A.G. Tescher: A Dual Transform Coding Algorithm, Proc. NTC '79, pp. 53.4.1-53.4.4

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMAGE CODING BELOW 0.5 BITS PER PIXEL USING VECTOR QUANTIZATION†

R. Aravind and Allen Gersho

Department of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106

ABSTRACT

Reasonable quality image coding has been possible at rates larger than 0.5 bpp for quite some time. In order to achieve lower rates with Vector Quantization algorithms it is necessary to exploit the two-dimensional correlation between image blocks. In this paper a coding scheme is presented that exploits this correlation by applying feedback around the vector quantizer; good quality is achieved at a rate of 0.375 bpp.

1. INTRODUCTION

In conventional image coding with Vector Quantization (VQ), the image is partitioned into square blocks of pixels and each block (vector) is coded independently. Reasonable quality coding has been achieved at rates in the neighborhood of 1 bpp (bits per pixel) [1], [2]. However, natural images are correlated well beyond the size of a single block in both the horizontal and vertical directions. The capability of direct memoryless VQ is typically limited to bit-rates in the range of 0.5 to 1.5 bpp where the quality varies with the dimension (number of pixels in the block) and with the perceptually based techniques used to reduce blocking effects and edge degradations. To achieve effective coding at rates below 0.5 bpp it is necessary to exploit the two-dimensional correlation between image blocks. We have experimented with finite-state vector quantization to exploit interblock correlation and human visual perception to achieve low rates while maintaining satisfactory visual quality.

Finite-state vector quantization is a new source-coding technique which has recently been applied to speech-waveform coding, giving significant improvement over memoryless VQ at the same rates and the same level of complexity [3], [4]. A Finite-State Vector Quantizer (FSVQ) can be viewed as a finite collection of ordinary vector quantizers, each with its own codebook. Each successive source (input) vector is encoded using a codebook determined by the current machine *state*. The current state and the transmitted channel symbol determine the next state. The FSVQ-coder exploits the correlation between source vectors by choosing the current codebook based on past behavior of the input vector sequence.

In order to improve the quality, or alternatively the compression factor, we seek to exploit both the intensity correlation between adjacent blocks and the local continuity of edges. To this end we define the state as a two-dimensional vector, such that the two components of the current state together determine the codebook with which to encode the current input vector, based on perceptual characteristics. We also introduce a perceptually motivated distortion measure (dm) for codebook design and encoding. This paper presents a generalization of the FSVQ structure that was reported by the authors in [5]. We give results for images encoded at 0.375 bpp; we are currently examining other rates in the range 0.2 - 0.4 bpp.

†This work was supported by the National Science Foundation.

2. FSVQ STRUCTURES

Let the input and output vectors of the FSVQ both be members of the Euclidean space bR^k, and let bS represent the (finite) state-space. The state symbol $bs \in bS$ is in general also a vector. The alphabet of channel symbols is denoted bN, and N is the size of the alphabet. The FSVQ consists of a pair of finite-state machines, the encoder C at the transmitter and the decoder D at the receiver. At any time, the state of the encoder and decoder is the same. The encoder takes the input vector process and outputs a channel-symbol sequence. The decoder produces a sequence of reproduction vectors from the channel symbols. The state evolves according to a *next-state function*, available to both the encoder and decoder. The encoding and decoding of the the input sequence $\{bx_n\ ;\ n = 0, 1, 2 \cdots\}$ proceeds as

initial state: bs_0
current state: bs_n
encoder: $u_n = C(bx_n, bs_n)$
decoder: $by_n = D(bs_n, u_n)$
next-state function: $bs_{n+1} = f(by_n, by_{n-L}, by_{n-L-1}, \cdots)$
next state: bs_{n+1}.

Here $\{u_n\}$ is the sequence of channel symbols, and $\{by_n\}$ is the sequence of reproduction vectors, which is available to both the encoder and decoder. Since the state update depends only on current and previous output vectors, the decoder can track the encoder state given the initial state and the sequence of channel symbols. The next-state function could employ an arbitrary number of past output vectors as arguments. In this work we will use three vectors to determine the (two-dimensional) state bs_n.

The collection $rC_{bs} = \{D(bs, u);\ u \in bN\}$ is the codebook of possible reproduction vectors, or the *reconstruction* codebook associated with the state bs. Every such codebook has N codevectors; the rate of the FSVQ is given by $r = \log_2 N$ bits per vector.

In order to search the codebook to find the best representation of an input vector it is necessary to define a distortion measure (dm). The objective fidelity measure used to quantify the overall FSVQ performance can be different from this dm. We now introduce a dm, d, which assigns a nonnegative cost $d(bx,by)$ for the reproduction of the input vector bx with the output vector by. We assume that the encoder operates on a minimum-distortion rule: given the current state bs and the input vector bx, it selects the channel symbol as

$$C(bx, bs) = \min_{u \in bN}{}^{-1}\ d(bx, D(bs, u)). \qquad (1)$$

The dm d itself can depend on the state bs.

3. FSVQ DESIGN

The design of the FSVQ involves first the definition of the states, and then the design of the various reconstruction codebooks. We will therefore first clearly define the states and then discuss codebook design. In this work the dimension k equals 16, corresponding to blocks of size 4×4.

3.1. The Definition of the States

The next-state function f is central to the definition and propagation of the states. In [5] a perceptually-based block classifier was used to form f. The components of the two-dimensional state were the two classes of a pair of vectors, calculated independently of one-another; more details can be found in [5]. In this paper we develop this idea further.

Consider the three neighboring 4×4 blocks A, B and C in Fig. 1. For the moment we take these blocks from input images; however, in the operation of the FSVQ the same function f, defined here, will be applied to past *output* 4×4 blocks so that the receiver can track the state.

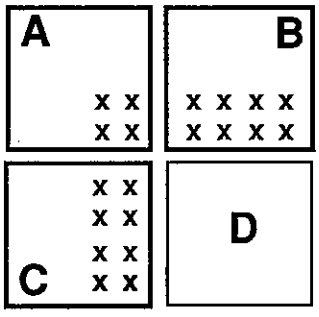


Fig. 1: The definition of the state

The position of the blocks in the (input) image is as shown in the figure. We desire a mapping between these blocks and a two-dimensional (state) vector whose components take a finite set of values. We use the term *class* ,as in [5] for the components of the state variable. The the next-state function, f, exploits both the adjacent-pixel correlation and the local edge-continuity across blocks.

We operate only on those pixels shown marked in the three blocks. The marked pixels (the *bottom two* rows) in block B are first passed through edge detectors. If no edge is detected, we simply calculate and quantize the mean value (which determines the *shade* class) of the two bottom rows of pixels. If an edge is present, it is classified according to orientation and the sign of the intensity change across it. The allowed edge-classes for block B are vertical B-W and W-B, and diagonal B-W and W-B. In case more that one edge-type is detected, the class is determined by the *dominant* edge (the edge with the strongest intensity transition). We do not recognize a horizontal edge for block B, instead declaring a shade class (by quantizing the mean of the bottommost row) when a horizontal edge occurs in the bottom rows of block B.

For block C, we process the two *rightmost columns* of pixels in a similar fashion to block B, allowing *horizontal* B-W and W-B edge-classes, and diagonal B-W and W-B edge-classes, and the shade classes. We convert a vertical edge into a shade class by quantizing the mean of the rightmost column. As in [5], we restrict diagonal edge-classes to 45° and 135°.

When both blocks B and C yield shade classes, we check for edges *between* blocks A and B and between blocks A and C. If we detect edges between these blocks, we apply a correction to the two classes calculated from blocks B and C; we will not go into detail here into this portion of the algorithm. The final output (ordered) pair of classes forms the state.

We have thus defined eight edge classes (4 × 2). We also chose eight as the number of shade classes, for a total of 16 classes for 4×4 blocks. However, it must be remembered that the component class obtained from block B is not permitted to be either of the two horizontal edge-classes. Similarly, the component class from block C is not allowed to be a vertical edge-class. Hence the maximum number of classes for either component is 14, and this implies that the maximum number of states is 196 (=14 × 14). The actual number of system states could be smaller because this figure allows for *all* possible combinations of shade classes, some of which can be eliminated because the mapping will not yield, for example, a very bright shade class for block B and a very dark shade class for block C.

3.2. The Reconstruction Codebooks

We follow the procedure presented in [5] to design the reconstruction codebooks. This process first partitions the training set, generating a subset for every state we desire a codebook, and it next designs the codebooks by applying a memoryless design technique to each subset.

The training set consists of vectors derived by blocking ten 512 × 512 images. These images were chosen in order to make the codebooks as general as possible and included landscapes, human faces and aerial photographs. To partition the training set, we apply the next-state function f described in section 3.1 to *every* triplet (A, B, C) of vectors in the training set. Each such triplet is thus mapped into a state bs that determines (and labels) the subset to which the training vector D (below vector B) is assigned. This procedure is repeated for all members of the training set. At the end of the partitioning, all vectors in a given subset follow or occur after the same state label. We can therefore argue that a codebook designed on a subset will be effective for encoding vectors which follow the state that labels the subset.

Since the reconstructed image will consist of vectors from the various codebooks, a critical design issue is to find perceptually effective codebooks. The best-known technique for codebook design is the LBG algorithm [6], which designs codebooks for a specific d.m. So far we have not placed any restrictions on the d.m.

It is known [2] that the direct application of the LBG algorithm with the MSE-d.m. yields codebooks containing very few vectors of high detail (i.e. with visible edges). Images encoded with such codebooks suffer severely from the so-called *staircase-effect*. For rates above 0.5 bpp, techniques described in [2] can be employed to redress this deficiency. However, at 0.375 bpp and k = 16, there are only 64 codevectors in each codebook, and this number is too small for these techniques to work. We have employed a new weighted-MSE (WMSE) d.m. and some ad-hoc tricks to partially solve this problem.

The WMSE-d.m. is defined in [5] through the edge-based classification of vectors:

$$d(bx,by) = \begin{cases} \|bx-by\|^2 & \text{if } Z(bx) = Z(by) \\ w\|bx-by\|^2 & \text{otherwise} \end{cases} \qquad (2)$$

Here w is a positive weight (we used 1.25 or 1.5) and Z denotes the edge-based classifier so that $Z(bx)$ is one of the eight edge classes or is the single shade class (no further classification of the shade class is performed). *All* the pixels of the vector are

involved in this classification, and not just its bottom rows or rightmost columns.

This d.m. attempts to reproduce edge-orientations at the cost of larger intensity errors than the usual MSE-d.m. When applied to a training set, it appears to form more meaningful clusters than the MSE-d.m., but, as explained in [5], it is impossible to define an optimal centroid for it. The ad-hoc technique of calculating the usual MSE-centroid at a given LBG-iteration, but checking for the edge-class of this centroid before using it as the codevector for the next iteration [5], works reasonably well. However, improved approaches to codebook design still form an active research topic.

3.3. The Next-State Function

Once all the reconstruction codebooks are obtained, the next state function f can be applied to any triplet of codevectors taken from any of the codebooks. Figure 2 shows the propagation of the state based on three past output vectors having the same spatial relationship as the vectors A, B, and C in Fig.1. We assume that the image is scanned left-to-right and top-to-bottom. The integer L denotes the index difference between two vertically adjacent vectors.

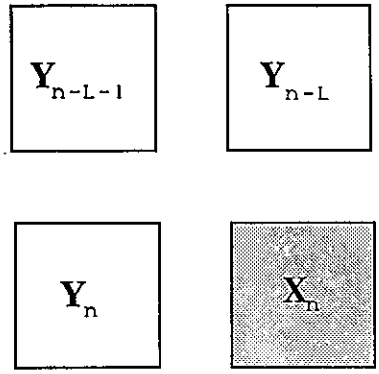


Fig. 2: The propagation of the state based on previous outputs

Thus we have

$$bs_{n+1} = f(by_n, by_{n-L}, by_{n-L-1}) \quad (3)$$

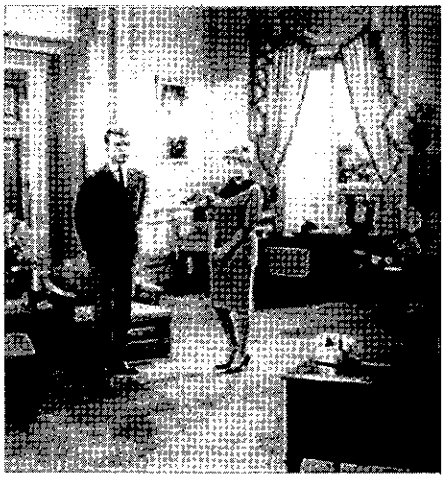
where by_{n-1} is the output vector to the left of the current input bx_n, by_{n-L} is the output above the input, and by_{n-L-1} is the output above by_{n-1}.

The outputs of the next-state function can either be stored in a state-transition table or computed on-line. The computation involved in finding the state is negligible in comparison with codebook search.

The FSVQ design is now complete. We bear in mind, though, that what we have described here is an *open-loop* design procedure. *Closed-loop* design implies the optimization of the codebooks for the FSVQ-structure. Though algorithms have been proposed for closed-loop design [3], there is no known *convergent* algorithm for this problem. Further, the perceptual issues involved in such design algorithms have not yet been resolved.

4. SIMULATIONS

We present results of 16-dimensional FSVQ-coders at 0.375 bpp ($N = 64$). In all our experiments we initialize the state with input vectors from the top row and leftmost column of the image to be encoded. In Fig. 3 we show the results for two different test images, both of which are outside the training set.



(a) the original

(b) the encoded image

(c) the original

(d) the encoded image

Fig. 3: Encoding at 0.375 bpp.

The coder has trouble tracking rapid changes in the input images, though it can handle isolated edges well, and we are currently pursuing ways to minimize the "blurring".

We tried to improve subjective performance by varying the d.m. with the state. It was noticed that the unrestricted use of the WMSE-d.m. sometimes caused large intensity errors, and led to severe blocking of output vectors. We therefore used the WMSE only when the state was composed of two shade classes, which meant that the surroundings were encoded with small error (it can be seen that shade regions do very well). In all other situations, even when only one of the component classes was an edge-class, we employed the MSE. Switching the d.m. did reduce some of the blocking, and more investigations are in progress.

Rates lower than 0.375 bpp do not appear likely to work at 16 dimensions. The current research effort is to design 25-dimensional systems (5×5 blocks) at 0.24, 0.28 and 0.32 bpp. One of the problems encountered at 25 dimensions is that a single block often contains more detail than can be modeled by a single edge. The task of good codebook design is even more complicated at this block size.

5. CONCLUSIONS

By combining the powerful pattern-matching idea of vector quantization with memory, we are able to preserve critical perceptual features even at low rates. One price is that memory requirements are quite high. The FSVQ presented here achieves the performance of memoryless VQ-based coders at approximately half the rates in [2].

REFERENCES

[1] R. Baker and R. Gray, "Differential Vector Quantization of Achromatic Imagery," *Proc. Picture Coding Symp.*, pp. 105-106, March 1983.

[2] B. Ramamurthi and A. Gersho, "Image Vector Quantization with a perceptually-based Block Classifier," *Proc. ICASSP*, pp. 32.10.1-32.10.4, March 1984.

[3] J. Foster, R. Gray, and M. Dunham, "Finite-state Vector Quantization for Waveform Coding," *IEEE Trans. Inform. Thy.*, pp. 348-355, May 1985.

[4] A. Haoui and D. Messerschmitt, "Predictive Vector Quantization," *Proc. ICASSP*, pp. 10.10.1-10.10.4, March 1984.

[5] R. Aravind and A. Gersho, "Low-rate Image Coding with Finite-state Vector Quantization," *Proc. ICASSP*, April 1986.

[6] Y. Linde, A. Buzo, and R. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Commun.*, pp. 84-95, January 1980.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DIGITAL COLOR IMAGE RESTORATION

P.H. Westerink, J. Biemond and P.H.L. de Bruin

Information Theory Group, Department of Electrical Engineering,
Delft University of Technology, P.O. Box 5031, 2600 GA Delft, the Netherlands

In this paper various aspects of digital color image restoration are discussed. At first attention is focused on the important problem of blur identification from the noisy blurred color image itself by means of spectral and cepstral techniques. Two types of blur will be investigated: uniform motion blur and defocusing blur, which is an example of a space-variant wavelength-dependent type of blur. The restoration will be performed in the frequency domain by means of a constrained least-squares filter. In order to measure the filter performance, two performance measures for color images are introduced, one of which is based on the properties of the human visual system. Finally, several experimental results on color images will be given. A comparison is made between reconstruction in the RGB or YIQ domain using test images and the newly introduced performance measures. Further, some experiments are described on the identification and restoration of photographically blurred color images.

1. INTRODUCTION

Observed or recorded images are not only corrupted by random observation noise, but also often degraded by the imaging medium. Defocusing, imaging with camera motion and imaging through atmospheric turbulence are examples of such degradations. In image restoration we try to remove these kinds of degradations by developing algorithms based on an adequate mathematical description of the degrading phenomena.

The image restoration problem has been thoroughly investigated, see for example Andrews and Hunt [1]. However, most of the applications have been limited to monochrome images. Some early work on color image restoration can be found in [2] and [3]. A color image is defined by three components, usually the tristimulus values red, green and blue (RGB). Each of these components is in itself a monochrome image. These color components can be linearly transformed into other component systems with higher energy packing capability. As an example, the Y (luminance) and I,Q (chrominance) component system is widely known as the basic color system for the NTSC television system.

In this paper attention will be focused on various aspects of digital color image restoration, such as blur identification, restoration both in RGB and YIQ domain and filter performance.

2. IMAGE FORMATION

Let a 3-D object or scene be imaged onto a 2-D image plane by means of an image formation and recording system (say a camera). If the image formation system can be modeled as a linear system, then in the discrete case, the following equation gives an adequate description of the image formation and recording process

$$g(i,j) = \sum_{(k,\ell)\in S_1} \sum h(i,j;k,\ell) f(k,\ell) + n(i,j), \qquad (i,j) \in S_2. \qquad (1)$$

Here $g(i,j)$ is the recorded discrete image, $h(i,j;k,\ell)$ is the 2-D impulse response or point-spread function (PSF) with support S_1 modeling the blur and $f(k,\ell)$ is the input image, i.e. the image which would be obtained in the case the PSF is a unit impulse function and the noise is negligible. The noise contribution $n(i,j)$ from the image formation and recording process is modeled as an additive process, uncorrelated with the image data.

If the PSF acts uniformly across the image and object field and consequently depends only on the relative difference i-k and j-ℓ, equation (1) reduces to the familiar convolution form

$$g(i,j) = \sum_{(k,\ell)\in S_1} \sum h(i-k,j-\ell) f(k,\ell) + n(i,j), \qquad (i,j) \in S_2. \qquad (2)$$

The above definitions are valid for monochrome images. For color images, each color component is in itself a monochrome image and can be modeled individually by eq. (1) or (2).

In the following we will discuss the PSF for two types of blur of particular interest in the color image restoration area: uniform motion blur and defocusing blur, which is an example of a space-variant wavelength dependent type of blur.

2.1. Motion blur

Motion blur is caused by a relative movement between an object (or scene) and the image formation system. We can distinguish many kinds of motion blur. The blur can be due to a translation or a rotation or a combination of the two, while the velocity of the motion can be varying.

Here we consider the simple case of translation at constant velocity. When the motion is at a constant velocity v along a horizontal line during the exposure interval [0,T], the distortion is one-dimensional and the continuous PSF is space-invariant with the following form (with continuous variables (x,y)):

$$h(x,y) = h(x) = \frac{1}{vT}, \quad 0 \leq x \leq vT; \quad y=0$$
$$= 0 \ , \text{ otherwise.} \tag{3}$$

The discrete equivalent of eq. (3) is given by

$$h(k,\ell) = h(k) = \frac{1}{L+1}, \quad 0 \leq k \leq L; \quad \ell=0,$$
$$= 0 \quad , \text{ otherwise} \tag{4}$$

where L is called the length of the motion blur. The PSF in eq. (4) is a parametric expression with one unknown parameter being the length L.

In the frequency domain eq. (4) is described by a discrete sinc-shaped transfer function with equidistant zero-crossings, perpendicular to the direction of the motion. For the identification of the blur, in Section 3 gratefully use will be made of this property.

2.2. Defocusing

It is possible to obtain a very precise model of the PSF for defocusing. This model can be derived from the diffraction theory, where the waveform character of light is considered [4]. A model of this kind is nevertheless complex and in practice hard to derive for a particular case of defocusing. However, when the sample frequency in the image plane is not too high, as in practical cases, then all effects described by diffraction theory can quite well be approximated by applying geometrical optics.

According to geometrical optics for a circular aperture an object point is imaged as a circle. The light intensity distribution is uniform within the circle and zero elsewhere. From fig. 1 we can derive for monochromatic light the radius r of this so-called circle of confusion (COC), determined by the severity of the defocusing.

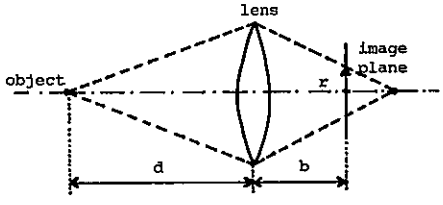


Fig. 1. Geometric description of defocusing.

This radius r is given by

$$r = |b.r_\ell(1/b+1/d-1/f)|, \tag{5}$$

where f denotes the focal distance of the lens, r_ℓ the effective lens radius and where b and d denote the image plane- and object distance to the lens, respectively. This function is shown in fig. 2 as a function of the object distance d, to demonstrate the effect that at all distances but one the object points are out of focus ($r \neq 0$). Due to a certain resolution limit δ, however, all object points within the range (d^-,d^+) appear to be in focus ($r<\delta$); this range is refered to as depth of field.

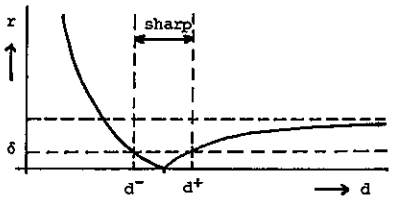


Fig. 2. Radius of the COC (r) versus object distance (d).

For identification and restoration purposes we simplify the model of the PSF by assuming that defocusing is a *locally* space-invariant parametric type of blur, the only parameter being the radius of the COC r. For the continuous case the PSF is

$$h(x,y) = \frac{1}{\pi.r^2}, \quad x^2+y^2 \leq r^2,$$
$$= 0 \quad , \text{ otherwise,} \tag{6}$$

and for the discrete case, its sampled equivalent is given by

$$h(k,\ell) = c, \quad k^2+\ell^2 \leq r^2,$$
$$= 0, \quad \text{otherwise.} \tag{7}$$

The constant c in eq. (7) is chosen in such way that the sum over all elements of the PSF h(k,ℓ) equals 1, to satisfy the physical assumption that the image formation system does not generate or absorbe energy.

For different wavelengths, however, the radius of the COC will not be the same due to the wavelength dependency of the refractive index of the lens; this is called chromatic aberration. The three components of a color image, R, G and B, each originate from a different frequency band of the imaged scene. When estimating the radius of the COC we must therefore be aware that this radius is wavelength dependent, resulting in a possible different PSF for defocusing for the three color components.

3. BLUR IDENTIFICATION

The first step towards the restoration of a degraded color image is to be able to identify the kind of degradation that particular image has suffered. Restricting ourself to motion blur or defocusing blur it appears that the transfer functions of both eq. (4) and (7) have an oscillatory character with typical zero-crossings patterns. It is therefore advantageous to identify the blur (or blur parameters) in the spectral or cepstral domain under the assumption of a locally space-invariant blur, in order to justify a frequency-domain approach. Blur identification techniques

in the spatial domain are described in [5,6].

3.1. Averaged log spectrum

If we assume the effect of the observation noise n(i,j) in eq. (2) to be negligible on the estimation of the parameters of the blur, and if we subdivide the degraded image g(i,j) into k subimages (which may overlap), assuming the support of the PSF to be small compared to the size of the subimages, we can write the averaged log spectrum as [1]

$$\frac{1}{k}\sum_{i=1}^{k}\log|G_i(u,v)| = \frac{1}{k}\sum_{i=1}^{k}\log|F_i(u,v)| + \log|H(u,v)|, \quad (8)$$

Here (u,v) are the discrete frequency variables and G, F and H are the DFT's of g, f and h. The first term at the right hand side will be smoothed due to the averaging. However, the zero-crossings in the transfer function of the blur will cause distinct negative peaks due to the taking of the logarithm. We can discriminate between motion blur and defocusing blur by considering the zero-crossings pattern: equidistant parallel lines for motion blur and non-equidistant concentric circles for defocusing blur.

For the case of defocusing blur we estimate the location of the first zero-crossing. Of course this is a rough approximation due to sampling, division into subimages, presence of noise, space-varying blur, etc. For motion blur another technique is more applicable, which is explained in the next sub-section.

3.2. Cepstrum for motion blur

For the identification of motion blur the use of the cepstrum is a very suitable method. The 2-D cepstrum of g(i,j) is defined as

$$c_g(p,q) = F^{-1}\{\log|F\{g(i,j)\}|\}, \quad (9)$$

where F denotes the discrete Fourier transform and F^{-1} its inverse.

To make use of the advantageous properties of the averaged log spectrum, the cepstrum of the degraded image is calculated by taking the inverse Fourier transform of eq. (8). Because the spectrum of the original image is averaged it will be smooth and therefore the largest contribution to the cepstrum will come from the transfer function H(u,v) of the blur. The zero-crossings in H(u,v) will give large (negative) values after taking the logarithm. These values, which dominate the averaged log spectrum, are equidistant and will result in very clear impulses when the inverse Fourier transform is applied. The location of the first impulse indicates the length and orientation of the motion blur.

4. RESTORATION

4.1. Constrained least-squares filter

After the blur in a degraded image has been identified, the resulting PSF can be used for restoration. As a restoration filter we apply the constrained least-squares filter in the frequency domain, which requires only a-priori knowledge of the PSF of the (locally space-invariant) blur and the noise variance. This filter is given by [1]:

$$\hat{F}(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2+\gamma|C(u,v)|^2}\cdot G(u,v), \quad (10)$$

where * denotes complex conjugate. The parameter γ is dependent of the noise variance. The linear operator C(u,v) is chosen to be the Fourier transform of the Laplacian operator, to restrict the solution $\hat{f}(i,j)$ (the restored image) to be smooth.

4.2. Boundary value problem

Because of a filter implementation in the frequency domain using the 2-D discrete Fourier transform (DFT), implicitly use is made of an image structure which is periodic both in horizontal and in vertical direction. If left- and right-hand side boundaries or lower- and upper boundaries are of different gray levels, frequency leakage occurs. Leakage components, that are close to the zeroes of the transfer function of the blur, will be amplified by the deconvolution action of the restoration filter. This will introduce so called ringing due to the boundaries. To reduce this effect the image may be extended with certain boundary values to smooth the sudden change in gray level. For that purpose a linear or third-order polynomial interpolation is used between the left- and right-hand side boundaries, and the lower- and upper boundaries. It was found that both techniques suppressed the ringing effects sufficiently [7].

5. EXPERIMENTAL RESULTS

5.1. Results for test images

In the case of color test images we have the original undegraded image available and therefore we are able to measure the filter performance by comparing the original, the degraded and the restored images. As an extension of the well-known improvement in signal-to-noise measure for monochromatic images we define

$$\eta_1 = 10^{10}\log\frac{\Sigma\Sigma\{(g_R-f_R)^2+(g_G-f_G)^2+(g_B-f_B)^2\}}{\Sigma\Sigma\{(\hat{f}_R-f_R)^2+(\hat{f}_G-f_G)^2+(\hat{f}_B-f_B)^2\}}\ \text{dB}, \quad (11)$$

where g, f and $\hat{f}$ denote the degraded, original and restored image and where R, G and B indicate the color components red, green and blue. This measure, however, slightly favors noise suppression above sharpness and therefore for low SNR we introduce a frequency-weighted version of η_1 given by

$$\eta_2 = 10^{10}\log\frac{\Sigma\Sigma\{H_{VS}[|G_R-F_R|^2+|G_G-F_G|^2+|G_B-F_B|^2]\}}{\Sigma\Sigma\{H_{VS}[|\hat{F}_R-F_R|^2+|\hat{F}_G-F_G|^2+|\hat{F}_B-F_B|^2]\}}\ \text{dB}. \quad (12)$$

Here G, F and $\hat{F}$ are the DFT's of g, f and $\hat{f}$, respectively. H_{VS} is the transfer function of the human visual system. Errors in the mid-fre-

quency band where the human eye is most sensitive are weighted more severely, while in areas of the spectrum where the eye is less sensitive larger errors are allowed. In fig. 3 it is shown for a particular case (defocusing with r^2=20) how both performance measures behave as a function of the SNR. As can be seen these two measures only differ for low SNR, while both are increasing in the high SNR area. It was found that in cases of low SNR the performance measure η_2 is more in agreement with a subjective evaluation than η_1 [7].

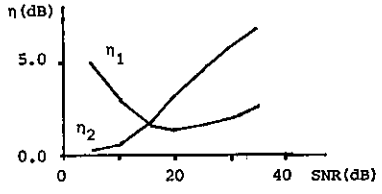


Fig. 3. Performance measures as a function of the SNR

First an identification and restoration simulation was carried out on a color test image ("Girl") of size 256x256 with 8 bit gray levels per color component. The original image is artificially blurred with space-invariant horizontal motion blur with L=9, and white Gaussian noise is added with a signal-to-noise ratio SNR=40 dB. Because of the one-dimensionality of the blur, the averaging in the log spectrum is done on a 1-D line basis over 50 image lines. Fig. 4a shows the 1-D averaged log spectrum and fig. 4b the corresponding cepstrum, where for display purposes the cepstrum is clipped at zero and made positive. From the first peak in fig. 4b we identify the length of the motion blur at L=9.

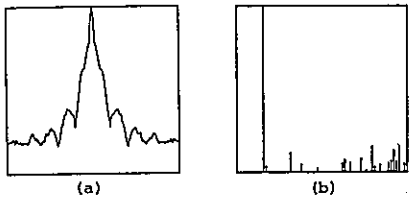


Fig. 4. 1-D spectrum (a) and cepstrum (b) for motion blur with L=9.

In general a color image is defined by its three primary components R, G and B. Alternate to this representation is the NTSC YIQ representation. In this space, the I and Q chrominance components are narrow band, so it may only be necessary to process the Y luminance component when restoring images. To compare results between RGB- and Y-restoration both experiments were carried out on the degraded color test image, using the identified PSF. As a result of processing the R, G and B components η_1=10.9 dB, while processing only the Y component resulted in 7.9 dB. Both visual inspection and performance measurement yield the same quality ranking. The actual degraded and restored color images will be shown at the conference presentation (see note). More numerical results and conclusions can be found in [8].

5.2. Results for photographically blurred images

Finally some identification and restoration experiments were carried out on photographically blurred images, for that purpose kindly placed at our disposal by KODAK Research Labs, Rochester, New York. As in the previous sub-section we restrict ourself to the identification results, while the actual restored color images will be shown at the conference (see note).

The first experiment contained a 256x128 train-image, locally blurred with horizontal motion blur. Because of the one-dimensionality of the blur the 1-D cepstrum was calculated over the horizontal (blurred) image lines. This way the length of the motion blur was estimated for all three color components R, G and B at L=7. The identified PSF's were used for the restoration of each of the color components.

The last experiment concerned an imaged girl's face of size 256x256 where defocusing blur was introduced by a wrong setting of the lens of a camera. To estimate the blur parameter, (the radius r of the COC), the averaged log spectrum was computed and the first zero-crossing of the transfer function of the blur was located. This yielded a radius for the red component of r=3.7 and for blue and green components r=3.4. Because of these small differences in r the PSF for defocusing with r=3.4 is used for the restoration of each of the color components R, G and B.

Of special interest and subject of current investigation is a refinement of identification and restoration techniques for defocusing blur in color images.

NOTE

The authors will provide a set of color pictures to readers who are particularly interested in the results.

REFERENCES

[1] Andrews, H.C. and Hunt, B.R., Digital Image Restoration, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1977.

[2] Bescos, J., Glaser, I. and Sawchuk, A.A., "Restoration of Color Images Degraded by Chromatic Aberrations", Applied Optics, vol. 19, no. 22, pp. 3869-3876, november 1980.

[3] Mancill, C.E., Digital Color Image Restoration, Ph.D. Thesis, Univ. of South Calif., USCIPI report 630, august 1975.

[4] Goodman, J.W., Introduction to Fourier Optics, Mc-Graw Hill, USA, 1968.

[5] Biemond, J. and Putten, F.G. v.d., "Image Restoration Using a Parallel Identification and Filtering Procedure", Proc. ICASSP, vol. 2, pp. 660-663, Tampa, Florida, 1985.

[6] Biemond, J., Putten, F.G. v.d. and Woods, J., "A Parallel Identification Procedure for Images with Noncausal Symmetric Blurs", Proc. ICASSP, vol. 2, pp. 1489-1492, Tokyo, Japan, 1986.

[7] Westerink, P.H., "Digitale rekonstruktie van kleurenbeelden in het frekwentiedomein", M.Sc. Thesis, Delft Univ. of Techn., Dep. of El. Eng., Inf. Th. Group, august 1985.

[8] Angwin, D.L., Kaufman, H., Woods, J.W., Westerink, P.H. and Biemond, J., "Effects of Processing Space Upon Color Image Restoration", Proc. Ann. Conf. on Inf.Sc. and Syst., Princeton Univ. Princeton, New Jersey, march 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A HYBRID IDENTIFICATION SCHEME FOR IMAGE AND BLUR PARAMETERS+

F. van der Putten, J. Biemond, J.W. Woods*

Information Theory Group, Department of Electrical Engineering,
Delft University of Technology, P.O. Box 5031, 2600 GA Delft, the Netherlands

ABSTRACT

In this paper a parallel identification and restoration procedure of noncausal image blurs is presented. It is shown that the blur identification problem can be specified as a parallel set of one-dimensional complex Autoregressive Moving Average (ARMA) identification problems. We will also briefly discuss the problem of obtaining a useful noncausal set of MA (blur) parameters from the identified minimum-phase set. Several identification and restoration results on real image data are given as examples.

1. INTRODUCTION

The first step towards the restoration of a degraded image is to be able to identify the kind of degradation the image has suffered. In real life situations the degrading system is not known but can at best be inferred from the underlying physical process, such as motion degradation or defocussing. Then the PSF can be parameterized from the a priori knowledge about the source of degradation, but the values of the parameters themselves will be unknown and must be estimated from the degraded image itself. The purpose of this paper is to complement the hybrid identification technique, described in [1], with the parallel Kalman filter, described in [2]. Note that this is an extension of our work in [3], where a parallel identification/restoration procedure was described for the special case of 1-D motion blur. We start in Section 2 with some basic model developments and show how we decompose the 1-D state space model for the sequence of noisy blurred image vectors into a set of nearly uncorrelated subsystems in the Fourier domain suitable for the derivation of a parallel bank of Kalman filters, described in Section 3. Since for the Kalman filter the image model parameters as well as the blur parameters must be known we give in Section 4 a brief description of the hybrid identification procedure and shall pay some attention to blur symmetrizing aspects. An extensive description of some experimental results on real image data will be given in Section 5.

2. IMAGE REPRESENTATION

Basic model development

It is assumed that the image can be represented by a zero mean homogeneous discrete MxN random field. The image can then be modeled by the semi-causal type of model

$$x(m,n) = \sum_{p,q\in W} a(p,q)\,x(m-p,n-q)+u(m,n) \qquad (1)$$

with support

$$W = \{p,q:0 \le p \le P,-Q \le q \le Q \wedge (p,q)\neq(0,0)\}$$

and u(m,n) is a noise input term which describes a white-noise process in the "m" variable and a colored noise process with nearest-neighbor support in the "n" variable [4].
If the imaging system is linear and spatially invariant and we assume the noise to be an additive process at the output of the system, then the observed or recorded image can be modelled by the following 2-D convolution summation

$$y(m,n) = \sum_{k=-K}^{K}\sum_{\ell=-L}^{L} g(k,\ell)\,x(m-k,n-\ell)+w(m,n) \qquad (2)$$

where w(m,n) is a white noise process uncorrelated with the data and g(k,ℓ) is the point spread function (PSF) of the blur, assumed to be symmetrical both in k and ℓ.

Model Decomposition

If we both circularize model (1) *and* data (2) and take the row-DFT over n=0,...,N-1 then we arrive at a set of nearly uncorrelated scalar subsystems in the Fourier domain [1]

$$A_0(j)X(m,j) = -\sum_{p=1}^{P} A_p(j)X(m-p,j)+U(m,j) \qquad (3a)$$

$$Y(m,j) = \sum_{k=0}^{K_0} G_k(j)X(m-k,j) + W(m,j) \qquad (3b)$$

where K_0=2K and $A_p(j)$, $G_k(j)$ are the DFT's of the defining sequences a(p,q) for fixed p resp. g(k,ℓ) for fixed k. (j is the

frequency variable).

3. PARALLEL KALMAN FILTERING IN FOURIER DOMAIN

The N nearly uncorrelated scalar subsystems defined in (3) can be transformed into N first-order vector dynamical models suitable for the derivation of N low-order Kalman filters [2], as outlined in fig. 1.

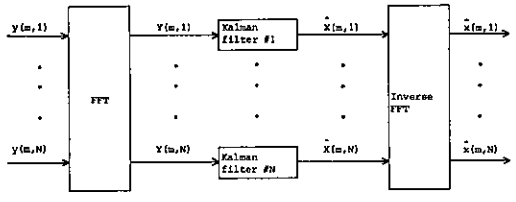


Fig. 1. Parallel Filter scheme

Considerable reduction of the total number of frequency components (channels) to be processed can be obtained by using only those channels with sufficient energy [5]. Note that this holds as well for identification.

4. PARALLEL IDENTIFICATION PROCEDURE

ARMA-model description

By eliminating X(m,j) from (3) and assuming the effect of the observation noise to be negligable on the estimation of the parameters we arrive at the following ARMA-model

$$Y(m,j) = -\sum_{p=1}^{P} D_p(j)Y(m-p,j) + V(m,j) + \sum_{k=1}^{K_0} N_k(j)V(m-k,j) \qquad (4)$$

where $D_p(j) \triangleq \frac{A_p(j)}{A_0(j)}$, $N_k(j) \triangleq \frac{G_k(j)}{G_0(j)}$ and

$V(m,j) \triangleq \frac{G_0(j)}{A_0(j)} U(m,j)$ for $A_0(j), G_0(j) \neq 0$.

The identification task is now reduced to the parallel identification of ARMA column sequences.

ARMA-model identification

Since we assumed hardly any observation noise, the Kalman filter derived from (3) will have a high deconvolving action. So very accurate knowledge about the PSF (the MA-part) is needed for good restoration results. There exists a variety of ways of doing ARMA identification. Many of these methods require the solution of a large set of non-linear equations. Recursive techniques are relatively simple but need long data runs to meet the desired accuracy [6]. Therefore it is desirable to find an estimation procedure which offers the potential of being relatively fast and estimating the MA-part with great accuracy. One such method was proposed by Graupe, Krause and Moore (GKM-method) [7,8]. They use a high-order AR-approximation as an intermediate step for the ARMA parameter estimation. This entirely linear method has the additional advantage that a number of fast AR-parameter estimation algorithms are available. From the definition of $D_p(j)$ and $N_k(j)$ we see that once these parameters are estimated we need also estimates for $A_0(j)$ and $G_0(j)$ to complete our identification scheme. If we apply an inverse DFT on $\sigma_v^2(j)$ then the resulting autocorrelation sequence can easily be used for the estimation of $a(0,q)$, $g(0,\ell)$ and σ_u^2, which are space domain parameters. [1].

Blur symmetrizing aspects

The estimated PSF has minimum phase, i.e. all zeros of the corresponding polynomials are located inside the unit circle, regardless of what the actual model was. If we aysume blur symmetry we know that if z_0 is a zero inside the unit circle, its conjugate reciprocal will be a zero outside. Thus by flipping out one of this pair of zeros we symmetrize our blur function. Special attention must be paid to a PSF with single order zeros on the unit circle. Due to the truncation of the infinite order polynomial in the GKM-method they will be located close to the unit circle. They can be shifted back onto the unit circle by noting that such zeros will cause a jump of π in the phase function and if we change the sign of the modulus function at those locations, we have an estimate for the symmetrical PSF with zeros *on* the unit circle. We can summarize our symmetrizing algorithm as follows:

Step 1: Determine the "unwrapped phase function" [9] $\phi(\omega)$ and the modulus function using a 64-point DFT.

Step 2: Calculate the first derivative $\phi'(\omega)$ and second derivative $\phi''(\omega)$. The local maxima and minima in $\phi''(\omega)$ mark the starting resp. ending points of a phase jump.

Step 3: Scan the frequency band from $\omega=0$ to $\omega=\pi$ and search for each maximum at location $\omega=\omega_{max}$ the next local minimum at $\omega=\omega_{min}$. For each pair of maxima and minima determine the width of the jump $\omega_{max}-\omega_{min}$ and the height of the jump $\phi(\omega_{max})-\phi(\omega_{min})$.

Step 4: The width of the jumps depends on the distance of the zeros to the unit circle. This allows us to set a threshold to discriminate if a zero is close enough to the unit circle. As a threshold we used width=5 points.

Step 5: If the height of the jump is in between $\frac{1}{2}\pi$ and π we will conclude that we found a single-order zero.

Step 6: If the test in step 5 is passed, then change the sign of the symmetric transform PSF at the point between ω_{max} and ω_{min} where $\phi'(\omega)$ has a local maximum otherwise leave the sign unchanged.

Because the parameters $G_k(j)$ are, for every channel 'j', made symmetrical in the k-direction seperately, there could be erroneous zero locations in some channels. However, if we smooth $G_k(j)$ in the j-direction, according to the assumption that the PSF is limited to a small area in space, the zero locations are tracked very well in going from one channel to another.

5. EXPERIMENTAL RESULTS

We shall present some practical identification and restoration results on artificially blurred real image data at different SNRs to show the feasibility of the proposed method to be used in combination with imaging systems of different quality. Since we blurred the images ourselves we can compare the results obtained with the identification/restoration procedure with those of the restoration with "known parameters", where we used as "known image model" the following parameters estimated from the undisturbed image

$$\text{"known image model"}: \begin{bmatrix} -0.0614 & 0.1740 & -0.0614 \\ 0.2440 & -0.7018 & 0.2440 \\ -0.4605 & 1.0000 & -0.4605 \end{bmatrix}.$$

As a test image we used the 256x256 "Cameraman", each pixel quantized in 8 bits. In artificially blurring images there is always the problem of the pixel values outside the image boundaries. We solved this problem by blurring the image circulantly and cutting out the 244x244 center part of it. We made it approximately circulant by using a third order polynomial for the interpolation from right to left boundary such that the row length is again 256 points.
We will present one model for motion blur and two models for defocussing. The identification results are only given at a SNR of 60 dB (those at 50 and 40 dB are similar). The filter improvement is measured with the well known MSE measure

$$\text{Improvement(dB)} = 10 \log \frac{\text{MSE(blurred image)}}{\text{MSE(reconstructed image)}}$$

Experiment 1: Uniform motion blur

We blurred the cameraman with the following PSF $\frac{1}{9}[1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$ (see fig. 2). The estimated image model and PSF (at 60 dB) are given by

$$\text{"estimated image model"}: \begin{bmatrix} -0.0497 & 0.1654 & -0.0497 \\ 0.1951 & -0.6896 & 0.1951 \\ -0.4169 & 1.0000 & -0.4169 \end{bmatrix},$$

PSF:

[0.1110, 0.1109, 0.1092, 0.1124, 0.1131, 0.1124, 0.1092, 0.1109, 0.1110]

Fig. 3 shows us the reconstruction result where we used the "known image model" and input PSF as a input for the Kalman filter. We see some "ringing" in the coat of the cameraman. The reconstruction of the 256x256 circulantly blurred image was ringing-free so we may conclude from this that this effect is caused by the fact that equation 3b is only *approximately* true for the blurred image with interpolated boundaries. When we used the estimated image model and PSF we see no noticeable difference in the reconstruction result (fig. 4) because of the high accuracy in the estimate of the PSF.

TABLE 1. The effect of observation noise

	Improvement in dB		
SNR	60 dB	50 dB	40 dB
Known parameters	11.8	12.0	.9.0
Estimated parameters	11.8	11.7	8.8

When we investigate the effect of the observation noise (Table 1) we see that at a SNR of 50 dB there is even a higher filter improvement, when we use known parameters, than at 60 dB. This is mainly because the filter has a lower deconvolving action and so the fact that our image is only approximately circulant has less effect and so there is less "ringing". We see that even at a SNR of 40 dB the identification/restoration procedure performs very well for this example.

Experiment 2: Pill-box model

One of the ways of modelling defocussing is by a so-called Pill-box. We blurred the cameraman with the following PSF:

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

The identification results are given by

$$\text{"estimated image model"}: \begin{bmatrix} -0.0363 & 0.1678 & -0.0363 \\ 0.2086 & -0.7021 & 0.2086 \\ -0.4276 & 1.0000 & -0.4276 \end{bmatrix},$$

$$\text{"estimated PSF"}: \begin{bmatrix} 0.1099 & 0.1113 & 0.1099 \\ 0.1115 & 0.1149 & 0.1115 \\ 0.1099 & 0.1113 & 0.1099 \end{bmatrix}.$$

Experiment 3: Gaussian PSF

The last PSF we investigated is a truncated 3x3 Gaussian pulse given by the following approximation

$$\text{"input PSF"}: \begin{bmatrix} 0.0449 & 0.1121 & 0.0449 \\ 0.1221 & 0.3318 & 0.1221 \\ 0.0449 & 0.1221 & 0.0449 \end{bmatrix}.$$

The estimation results are:

"estimated image model": $\begin{bmatrix} -0.0220 & 0.0542 & -0.0220 \\ 0.1702 & -0.5592 & 0.1702 \\ -0.4185 & 1.0000 & -0.4185 \end{bmatrix}$,

"estimated PSF": $\begin{bmatrix} 0.0483 & 0.1309 & 0.0483 \\ 0.1155 & 0.3138 & 0.1155 \\ 0.0483 & 0.1309 & 0.0483 \end{bmatrix}$.

6. DISCUSSION

We see in all results that although the image model is often not accurately estimated, the filter results are still good since only the accuracy of the PSF matters.
With the method as proposed here it seems to be possible to do identification and restoration up to a SNR of 40 dB. The CPU time required for the complete procedure depends on the size of the blur but is in the order of minutes (on a VAX 11/750 + AP500 Array Processor). Of special interest and subject of further investigation is the use of frequency interpolation techniques for those channels in which the SNR is too low for proper identification in order to improve the results at lower SNR.

7. REFERENCES

[1] J. Biemond, F.G. van der Putten, J.W. Woods, "A parallel Identification Procedure for Images with Noncausal Symmetric Blurs", Proc. 1986 IEEE Int. Conf. on ASSP, Tokyo, Japan, April 1986, pp. 1489-1492.
[2] J. Biemond, J. Rieske, J.J. Gerbrands, "A Fast Kalman Filter for Images Degraded by Both Blur and Noise", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-31, No. 5, Oct. 1983, pp.1248-1256.
[3] J. Biemond, F.G. van der Putten, "Image Restoration Using a Parallel Identification and Filtering Procedure", Proc. 1985 IEEE Int. Conf. on ASSP, Tampa, Florida, March 1985.
[4] A.K. Jain, "Advances in Mathematical Models for Image Processing", Proc. IEEE, Vol. 69, May 1981, pp. 502-528.
[5] T. Katayama, "Restoration of Images Degraded by Motion Blur and Noise", IEEE Trans. on Automatic Control, Vol. AC-27, No. 5, Oct. 1982, pp. 1024-1033.
[6] L. Ljung and T. Söderström, "Theory and Practice of Recursive Identification", The MIT Press 1983.
[7] D. Graupe, D.J. Krause, J.B. Moore, "Identification of Autoregressive Moving Average Parameters of Time Series", IEEE Trans. on Aut. Control, Feb. 1975, pp. 104-106.
[8] L.B. Jackson, "Simple Effective MA and ARMA Techniques", Proc. 1983, IEEE Int. Conf. on ASSP, Boston, April 1983, pp. 1426-1429.
[9] José Tribolet, "A New Phase Unwrapping Algorithm", IEEE Trans. on ASSP, Vol. 25, No. 2, April 1977, pp. 170-177.

+ This work was supported in part by the Netherlands Organisation for the Advancement of Pure Research (ZWO) and in part by NSF Grant No. ECS 8313889.

* On leave from Rensselaer Polytechnic Institute, Troy, New York.

Fig. 2. Uniform motion blur at SNR = 60 dB

Fig. 3. Restoration result with "known parameters"

Fig. 4. Restoration result with "estimated parameters"

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ITERATIVE IMAGE RESTORATION WITH RINGING REDUCTION

R.L. Lagendijk, J. Biemond and D.E. Boekee

Information Theory Group, Department of Electrical Engineering
Delft University of Technology, P.O. Box 5031, 2600 GA Delft, the Netherlands

Linear space-invariant image restoration filters often introduce ringing effects near sharp intensity transitions. In this paper it is shown that this is caused by the noise-suppression action of the filter near the zero-crossings of the transfer function of the blurring system. Furthermore, an iterative restoration algorithm is described which includes adaptivity and the usage of a priori knowledge as two possible ringing reduction techniques.

1. INTRODUCTION

In many practical situations, image degradations can be modelled by a linear blur (e.g. motion, defocussing, atmospheric turbulence) and an additive signal-uncorrelated noise term. If the observed image is represented by an MxN array of real picture elements $g(i,j)$ $(1\leq i\leq M,\ 1\leq j\leq N)$, then in the space-invariant case it can be described by the following 2-dimensional convolution summation [1]:

$$g(i,j)= \sum_{(k,\ell)\in W_d} \sum d(k,\ell)f(i-k,j-\ell) + n(i,j)$$

$$= d(i,j) * f(i,j) + n(i,j). \qquad (1)$$

Here $f(i,j)$ is the original image; $n(i,j)$ the additive noise term and $d(i,j)$ the point-spread function (PSF) of the imaging system introducing the blur. It is assumed that the support W_d of this PSF is much smaller than the size of the image.

Generally, the knowledge about the noise term is very limited: $n(i,j)$ results from a stochastic zero-mean process with variance σ_n^2, and is uncorrelated with the original image. Due to this restricted knowledge, the original image cannot be restored exactly from the blurred version $g(i,j)$. In image restoration we try to remove the degradations caused by the blur and noise in such a way that an improved image is obtained which is an acceptable approximation of the original image $f(i,j)$. Since the image restoration problem is ill-posed or ill-conditioned [1]-[3], we have to prevent the amplification of noise, and in particular the amplification of high-frequency noise. To this goal a smoothness requirement is imposed on the solution of the restoration problem, yielding several kinds of restoration filters (e.g. Wiener filter, Constrained least-squares filter) [1]. Unfortunately, in many restored images the very disturbing effect of ringing (also referred to as superwhites and superblacks, or over- and undershoots) can be observed in the vicinity of sharp intensity transitions. Particularly the restoration of images degraded by linear motion blur often introduces severe ringing effects.

In this paper we focus on the question as to how ringing is created in linear space-invariant restoration filters. Next we describe two methods for the reduction of ringing effects. Finally, we propose an iterative restoration algorithm which incorporates both ringing reduction methods.

2. ON THE ORIGIN OF RINGING

We consider a 1-D linear space-invariant restoration filter with PSF $h(i)$. In order to characterize the errors in the filtered signal $\hat{f}(i)$, we first decompose $\hat{f}(i)$ as follows:

$$\hat{f}(i)= h(i) * g(i)$$

$$=[d^{-1}(i)*d(i)]*h(i) * [d(i)*f(i)+n(i)]$$

$$=[d(i)*h(i)]*f(i) + d^{-1}(i)*[d(i)*h(i)]*n(i), \qquad (2)$$

where $d^{-1}(i)$ denotes the exact inverse sequence of the PSF $d(i)$ of the blurring system (i.e. $d^{-1}(i)*d(i) = \delta(i)$, where $\delta(i) = 1$ if $i=0$ and $\delta(i) = 0$ if $i\neq 0$). Next the error sequence $e(i)$ is introduced to characterize the deviation of $h(i)$ from the exact inverse sequence $d^{-1}(i)$:

$$d(i) * h(i) = \delta(i) - e(i). \qquad (3)$$

Applying the DFT on eq. (3) [4], we get

$$D(u) \cdot H(u) = 1 - E(u). \qquad (4)$$

The better the sequence $h(i)$ resembles $d^{-1}(i)$, the more $e(i)$ approaches a zero-sequence. Substituting eq. (3) into eq. (2) yields

$$\hat{f}(i) = f(i) - e(i)*f(i) + d^{-1}(i)*[\delta(i) - e(i)]*n(i), \qquad (5)$$

or in the Fourier domain:

$$\hat{F}(u) = F(u) - E(u)F(u) + \frac{N(u)}{D(u)} \cdot (1-E(u)). \qquad (6)$$

From the equations (5) and (6) we observe that a restored signal $\hat{f}(i)$ is composed of three separate parts, each having a clearly different interpretation. Naturally, the original $f(i)$, which we try to compute, represents the major contribution to $\hat{f}(i)$. This original is, however, contaminated by two additive terms, leading to the inevitable difference between $f(i)$ and $\hat{f}(i)$. In the first place the additive noise $n(i)$ in the blurred signal $g(i)$ is contained in the restored signal as the filtered noise $d^{-1}(i)*(\delta(i)-e(i))*n(i)$. Given the noise and $d^{-1}(i)$, only $e(i)$ can prevent noise amplification. Secondly, the deviation of the PSF $h(i)$ of the restoration filter from the inverse sequence $d^{-1}(i)$ introduces the term $e(i)*f(i)$, which has to be subtracted from $f(i)$. Observe that this term depends only on $e(i)$ and $f(i)$, and not on $n(i)$. Therefore, $e(i)*f(i)$ will be related to the local structure in the signal $f(i)$ itself. It is this convolutional action which is the origin of the phenomenon of ringing, as will be shown by considering the properties of the error sequence $e(i)$ and its spectrum $E(u)$.

If the transfer function $D(u)$ of the blurring system approaches zero ($|D(u)| \to 0$), equation (6) shows that $E(u)$ must take the value +1.0 to prevent the appearance of enormously magnified noise $\frac{N(u)}{D(u)}$ in the restored signal $\hat{F}(u)$. Stated differently, near a zero-crossing of $D(u)$, $E(u)$ must form a bandpass filter, with bandwidth depending on the signal-to-noise ratio (SNR). A very high SNR allows a small bandwidth, whereas a low SNR requires a large bandwidth.

We now focus on linear motion blur as a worst case situation. The PSF of this space-invariant blur is defined by:

$$d(i) = \frac{1}{L}, \quad -\frac{L-1}{2} \leq i \leq \frac{L-1}{2}$$

$$= 0, \quad \text{elsewhere.} \qquad (7)$$

Applying the Fourier transform on $d(i)$ yields:

$$D(\omega) = \frac{1}{L} e^{j\omega\left(\frac{L-1}{2}\right)} \frac{1-e^{-j\omega L}}{1-e^{-j\omega}}, \quad 0 \leq \omega < 2\pi. \qquad (8)$$

The $(L-1)$ zeros of $D(\omega)$ are located at the frequencies $\omega_j = \frac{2\pi}{L} j$ $(j=1,2,\ldots,L-1)$. The discrete equivalent $D(u)$ of $D(\omega)$ may of course have no exact zeros as a consequence of sampling in the frequency domain, but the zero-crossing will still be located around the frequencies ω_j.

From the previous discussion we are now able to sketch the typical shape of $|E(\omega)|$ (fig. 1). The exact $E(\omega)$ obviously depends on the filter under consideration and the SNR-dependent filter parameter(s), and may also depend on additional knowledge such as the power spectrum of the original or the noise.

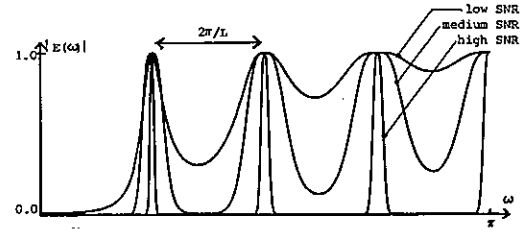


Fig. 1. Typical shape of $|E(\omega)|$ for various SNR (Motion blur with L=8).

Generally, the peaks in $|E(\omega)|$ are more distinct for relatively high SNR. Although the expression for $E(\omega)$ is usually quite easy to derive for the filters under consideration, a closed form for the error sequence $e(i)$ is seldom easy to find. However, as a consequence of the peaked nature of $|E(\omega)|$, the error sequence $e(i)$ will be dominated by positive impulses at the locations $i=\pm k.L$ $(k=0,1,2\ldots)$, as has been sketched in figure 2.

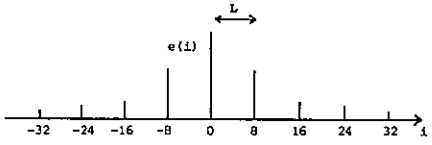


Fig. 2. Dominant impulses in the error sequence $e(i)$. (Motion blur with L=8).

As a result of these dominant impulses in $e(i)$, ringing is created in the restored signal $\hat{f}(i)$. In flat regions of $f(i)$ the convolution $-e(i)*f(i)$ has hardly any (visible) effect, but particularly near sharp intensity transitions (such as edges and bright point-sources) this convolution will cause the appearance of negative version of the intensity transition at distances $\pm k.L$ $(k=1,2,\ldots)$. Since the impulses at $\pm L$ have usually the largest value, ringing will be most severe at distance $\pm L$.

The severity of the ringing can be measured by the size of the dominant impulses $e(+L)$ and $e(-L)$. The larger the bandwidth of the bandpasses in $E(\omega)$, the larger the sizes of the impulses $e(+L)$ and $e(-L)$ are. Observe, however, that $e(+L)$ and $e(-L)$ are only dominant in the error sequence $e(i)$ if $|E(\omega)|$ has a distinctly peaked character. Since the bandwidth of a bandpass in $E(\omega)$ depends on the SNR, the severity of the ringing is obviously SNR-dependent.

To end this section we present three examples illustrating the ringing effect. In all examples the distortion is linear motion blur with L=8.

Example 1: Pseudo-inverse filter [1]
From the definition of the pseudo-inverse filter it follows that

$$E(u) = \begin{cases} 1, & \text{if } D(u) = 0 \\ 0, & D(u) \neq 0 \end{cases} \qquad (9)$$

Figure 3 shows the (SNR-independent) error spectrum $|E(u)|$ and sequence e(i), introducing hardly any ringing.

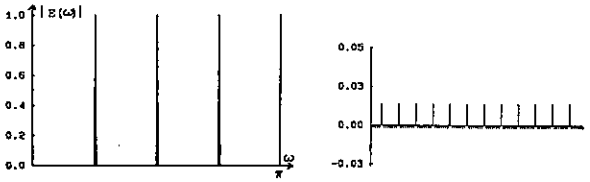


Figure 3. Error spectrum $|E(u)|$ and error sequence e(i) for the pseudo-inverse filter

Example 2: Constrained least-squares filter [1],[4]
By the definition of the constrained least-squares filter, and by using eq. (4) we obtain

$$E(u) = \left(1 + \frac{|D(u)|^2}{\alpha |L(u)|^2}\right)^{-1}, \qquad (10)$$

where L(u) is a regularizing operator, reflecting desired properties of the restored signal (e.g. we used the Laplacian operator to constrain the high-frequency content of $\hat{f}(i)$). Figure 4 shows $|E(u)|$ and e(i) for several values of the SNR-dependent parameter α. For $\alpha \to 0$ the constrained least-squares filter approaches the pseudo-inverse filter.

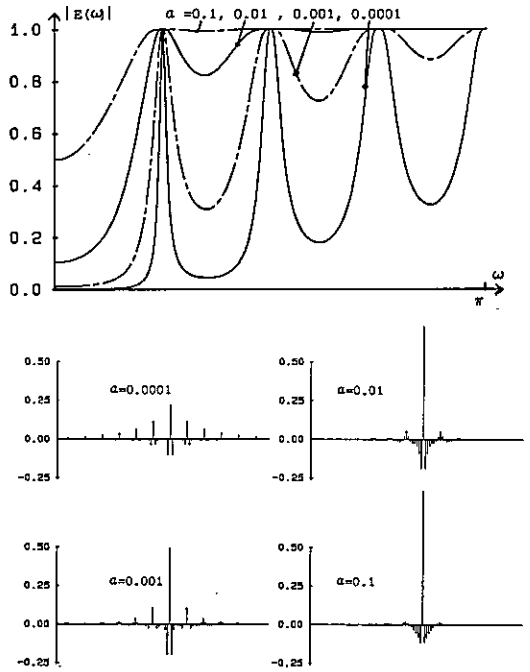


Figure 4. Error spectrum and error sequence for the constrained least-squares filter (α=0.0001, 0.001, 0.01, 0.1)

Example 3: Iterative filter [5]-[7]
We consider the iterative approximation of the pseudo-inverse filter estimate by the following algorithm:

$$\hat{f}_{k+1}(i) = \hat{f}_k(i) + \beta d(-i) * (g(i) - d(i) * \hat{f}_k(i)) . \qquad (11)$$

The iteration step-dependent $E_k(u)$ is easily obtained:

$$E_k(u) = (1 - \beta |D(u)|^2)^k . \qquad (12)$$

The number of applied iterations depends on the SNR. If the SNR increases, the parameter k may also increase, yielding a better approximation of the pseudo-inverse filter.

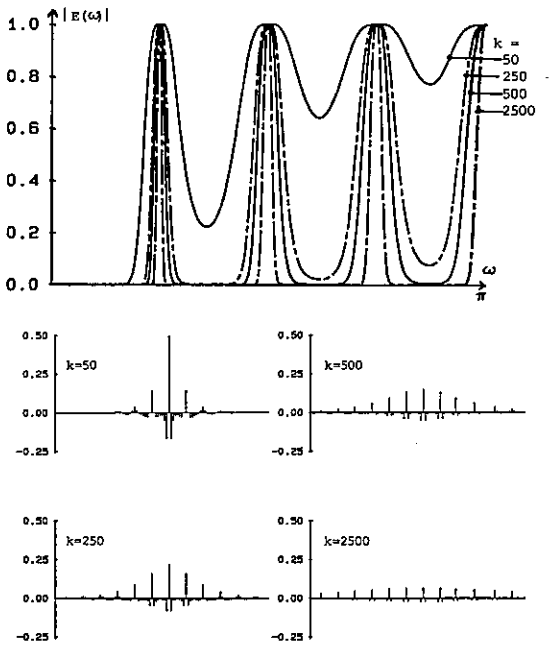


Figure 5. Error spectrum and sequence for eq.(11) (β= 0.8, k = 50, 250, 500, 2500) .

3. TWO METHODS TO REDUCE RINGING

The use of a priori knowledge about the original image in restoration algorithms is one method to suppress the arising of ringing. Usually, this knowledge is represented by deterministic constraints C_i. Unfortunately, relatively simple linear space-invariant restoration filters cannot utilize this kind of knowledge, because the constraints are often nonlinear and/or space-variant. In recent years fruitful use has been made of the theory of the projections onto convex sets in developing iterative restoration algorithms, which can incorporate many kinds of a priori knowledge [5]-[10]. It has been shown that if the a priori knowledge, represented by m convex sets, gives an accurate description of the original data, the iterative restoration approach yields very good results, simultaneously suppressing noise magnification and ringing effects. Iterative restoration algorithms based

on the projections onto convex sets are particularly suitable in processing X-ray fluorescence signals (positivity constraint), text images (locally bounding the intensities) and astronomical images (positivity, finite extent constraint).

However, when we have to remove blur from more complicated signals, such as real-life scenes, the use of deterministic constraints often turns out to be not powerful enough to suppress ringing, and the restored images benefit but little by employing deterministic constraints. To reduce ringing effects in this kind of image an adaptive algorithm has to be used, which filters the image in accordance to local properties, and in particular in accordance to local intensity transitions. In the previous section we have seen that ringing mainly arises in the vicinity of sharp intensity transitions as a result of suppressing the noise. Hence ringing can be reduced by less suppressing the noise near edges and bright point sources. This is also in agreement with various psychophysical experiments, which have shown that a human observer prefers sharp edges in images, and that at the same time sharp edges considerably reduce the visibility of noise (noise masking).

In the next section we describe an adaptive iterative image restoration algorithm that incorporates the two described methods to reduce ringing. A more extensive description of this algorithm (including derivation, proof of convergence, etc) is given in [8].

4. AN ADAPTIVE ITERATIVE RESTORATION ALGORITHM

We require the restored image $\hat{f}$ to satisfy the following three conditions:

(i) $$\|g(i,j) - d(i,j) * \hat{f}(i,j)\|_R^2 \leq \varepsilon^2. \quad (13)$$

Here R is a weight matrix with weight coefficient r_{ij} for the picture element (i,j). With R we can locally regulate whether the local restoration process is dominated by pseudo-inverse filtering (r_{ij} large: near edges), noise smoothing (r_{ij} small: flat regions) or any intermediate kind of filtering.

(ii) $$\|\ell(i,j) * \hat{f}(i,j)\|_S^2 \leq E^2. \quad (14)$$

Usually, $\ell(i,j)$ represents a high-pass filter, hence eq. (14) imposes a smoothness requirement on a restored image. The weight matrix S locally regulates this requirement, depending on the local properties.

(iii) $\hat{f}$ satisfies a deterministic constraint C, which represents certain a priori knowledge about the original image f. The projection onto the closed convex set described by C is denoted by P [10].

The adaptive iterative restoration algorithm is derived by first combining both sets described by eq. (13) and (14) into a single quadrature formula:

$$\Phi(\hat{f}) = \|g(i,j) - d(i,j) * \hat{f}(i,j)\|_R^2 + \alpha \|\ell(i,j) * \hat{f}(i,j)\|_S^2, \quad (15)$$

where $\alpha = (\varepsilon/E)^2$. Next, $\Phi(\hat{f})$ is minimized subject to the deterministic constraint C, yielding our adaptive iterative image restoration algorithm [8]:

$$\hat{f}_{k+1}(i,j) = P[\{\delta(i,j) - \alpha\beta\ell(-i,-j) * (s_{ij}.\ell(i,j))\} * \hat{f}_k(i,j) + \beta d(-i,-j) * r_{ij}.\{g(i,j) - d(i,j) * \hat{f}_k(i,j)\}]. \quad (16)$$

Equation (16) clearly incorporates both ringing reduction methods described in the previous section. The a priori knowledge about the original image is utilized by projecting the data onto the convex set, described by the constraint C, in every iteration step. The weight matrices R and S locally regulate the action of the filter. For example, in the vicinity of sharp intensity transitions the coefficients s_{ij} take small values, thus preventing smoothing of the data by the term $(\delta(i,j) - \alpha\beta\ell(-i,-j) * (s_{ij}\ell(i,j))$, while the relatively large coefficients r_{ij} allow large corrections $(g(i,j) - d(i,j) * \hat{f}_k(i,j))$ for the next iteration result.

5. CONCLUSIONS AND DISCUSSION

We have described the origin of ringing in linear space-invariant restoration filters, and proposed an iterative image restoration algorithm which incorporates the usage of a priori knowledge and adaptivity as two possible ringing reduction techniques. In order to show the ringing reduction capabilities of our algorithm we will present several experimental results on noisy blurred images during the conference.

Current research concentrates on locally optimizing the filter parameters, and on the development of other generally applicable deterministic constraints, both aspects aiming at further reduction of ringing effects.

REFERENCES

[1] Andrews, H.C. and B.R. Hunt, *Digital Image Restoration*, Prentice-Hall, Inc, New Jersey, 1977.

[2] Sanz, J.L.C. and T.S. Huang, Unified Hilbert Space Approach to Iterative least-squares Linear Signal Restoration, *JOSA* 73, 1983.

[3] Tikhonov, A.N. and V.Y. Arsenin, *Solutions of Ill-Posed Problems*, Wiley, Washington, 1977.

[4] Gonzalez, R.C. and P. Wintz, *Digital Image Processing*, Addison Wesley, Reading Mass., 1977.

[5] Schafer, R.W., R.M. Mersereau and M.A. Richards, Constrained Iterative Restoration Algorithms, *Proc. IEEE* 69, 1981.

[6] Frieden, B.R., Image Enhancement and Restoration, in *Picture Processing and Digital Filtering*, T.S. Huang (ed), Springer, Berlin, 1975.

[7] Trussell, H.J., Convergence Criteria for Iterative Restoration Methods, *ICASSP'83*, Boston.

[8] Biemond, J. and R.L. Lagendijk, Regularized Iterative Image Restoration in a weighted Hilbert Space, *ICASSP'86*, Tokyo.

[9] Trussell, H.J. and M.R. Civanlar, The Feasible Solution in Signal Restoration, *IEEE trans. ASSP* 32, 1984.

[10] Youla, D.C. and H. Webb, Image Restoration by the Method of Convex Projections, *IEEE trans. MI* 1, 1982.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SUM OF ABSOLUTE DIFFERENCE VALUES SMOOTHING: EVALUATION AND APPLICATION

A. de ALBUQUERQUE ARAÚJO

Departamento de Engenharia Elétrica
Universidade Federal da Paraíba
58.100 - Campina Grande - PB., Brasil

Spatial-domain noise smoothing techniques have been widely used in image enhancement tasks. An edge-preserving smoothing method presented in (1) plus seven other stablished algorithms have their performance evaluation illustrated. Application of these methods as preprocessing schemes in edge-detection and segmentation tasks is reported.

1. INTRODUCTION

Image enhancement usually concerns contrast stretching, smoothing, sharpening and highlighting of specific features. An offshoot of the process is edge detection and segmentation, which is very useful in automatic classification and description. The application of enhancement techniques, in general, improves human viewing ability and is expected to facilitate further processing and to improve analysis results.

This work deals with spatial-domain techniques which perform some type of local operation, such as averaging, through a mask to the picture. The gray level of the pixel at the center of this mask is replaced by the gray level average of the pixels inside the mask. The main characteristics of the spatial smoothing algorithms are the reduction of noise corruption and the preservation of important features such as edges. They are sometimes refered to as edge-preserving smoothing approaches. All can be iterated to improve their performance.

The algorithms tested and compared in this work are:

sum of abs. dif. values smooth. 5x5 - SADVS (1),
unweighted neighbor avearaging 3x3 (2),
median filtering 3x3 (2)
k-nearest neighbor averaging 3x3 - KNN6 (3)
gradient inverse weight. smooth, 3x3 - GIWS(4),
most homog. neighborhood smooth, 5x5 - MHNS(5),
slope facet model smoothing 5x5 (6), and
extremum sharpening 3x3 - XSHARP (7)

2. EVALUATION

The synthetical image used to evaluate the performance of the algorithms is of size 128x128, consisting of a circle of radius 50. The system resolution allows 256 gray levels. In a first step, Gaussian noise (0;20.0) with null mean and standard deviation 20.0 is added to the original image, which is then processed by the algorithms (four iterations).

Two performance criteria were considered: a) noise smoothing efficiency, expressed by the reduction of noise standard deviation (SD), and b) fidelity to the original noise free image, expressed by the mean square error (MSE). In Table 1, column 1 shows the results obtained for the noise standard deviations of the filtered images (forth iteration) computed from a flat area of size 21x21. It can be seen that averaging and median filtering are the most efficient ones in terms of smoothing noise, followed by SADVS. GIWS presents the worst performance.

TABLE 1 - CIRCLE

Methods	SD	MSE1	MSE2
AVERAGE	2.76	87.27	78.90
MEDIAN	3.23	29.71	77.16
KNN6	3.94	37.72	79.99
GIWS	8.46	74.22	108.13
MHNS	4.65	39.72	66.03
FACET	5.45	37.36	60.86
SADVS	3.69	27.28	76.47
ORIGINAL	0.0	0.0	58.59
+ NOISE	20.0	385.82	153.25

The results obtained for the mean square error are shown in Table 1, column 2. SADVS is now the best performer, due to its noise smoothing with edge retention property. Averaging presents the worst results, reflecting the fact that although it reduces well the effects of random noise it also affects considerably the original information, usually in the form of blurring.

In order to test the edge-sharpening capacity of the algorithms in presence of noise, Gaussian noise (0;10,0) is added to a blurred circle image. Results of the MSE after processing the image once by each filter are shown in Table 1, column 3.

In addition to the above performance criteria, scan line profiles of the filtered images were

used to visualize the algorithms capacity of edge retention and edge sharpening.

It can be observed (Fig. 1) that all algorithms but averaging are very good in preserving edges with KNN6 being slightly inferior to the others.

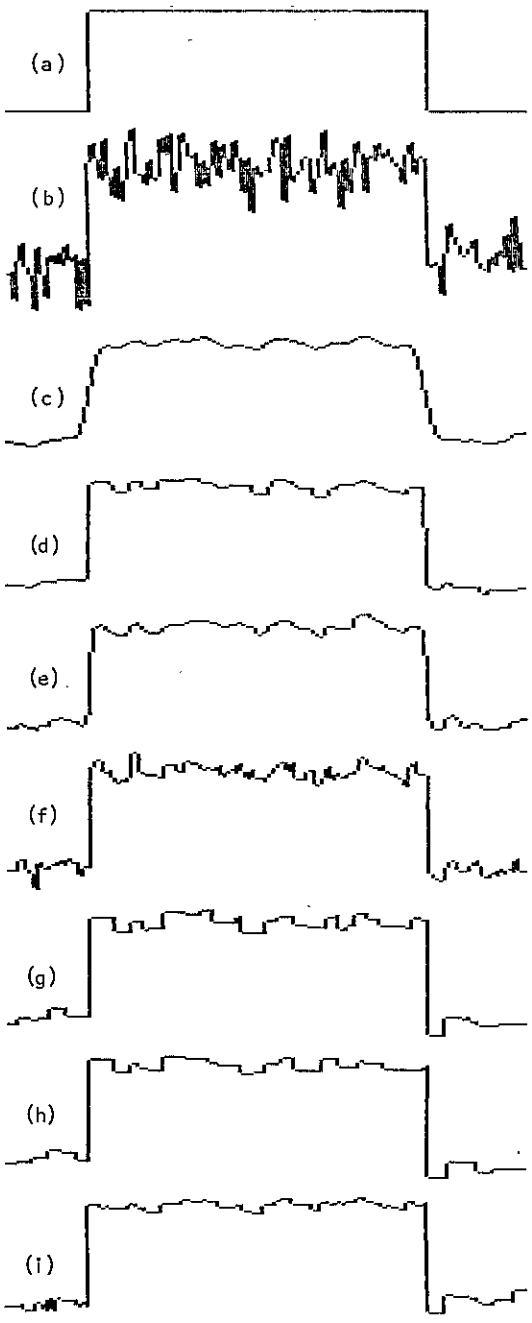


Figure 1. Profiles of scan line 45: (a) original, (b) with noise, (c)-(i) images filtered by AVER., MEDIAN, KNN6, GIWS, MHNS, FACET, and SADVS.

It can be also observed (Fig. 2) that MHNS sharpens edges the best, due to its directional subregion average. However MHNS does create distortions and introduces artifacts. FACET sharpens edges well followed by SADVS, with the advantage that they are relatively free of created artifacts.

4. APPLICATION

Determining the boundaries of objects is one of the major first steps in extracting the information in a picture. Two approaches are generally employed: a) locating discontinuities in intensity (edges points) and connecting them, and b) finding regions of fairly uniform intensity by growing regions or thresholding at a calculated level.

In this experiment the image to be segmented was preprocessed once by algorithms MHNS, FACET, SADVS, and XSHARP (Fig. 3, line 1). The test image is a girl picture (GIRL) of size 128x128 with 256 gray levels. Two different approaches to segmentation enhancement were attempted: a) an edge detection approach which employs gradient operators plus thresholding and thinning of edge vectors; and b) a region detection approach which uses a split-and-merge scheme based on recursive subdivision of the image into quadrants.

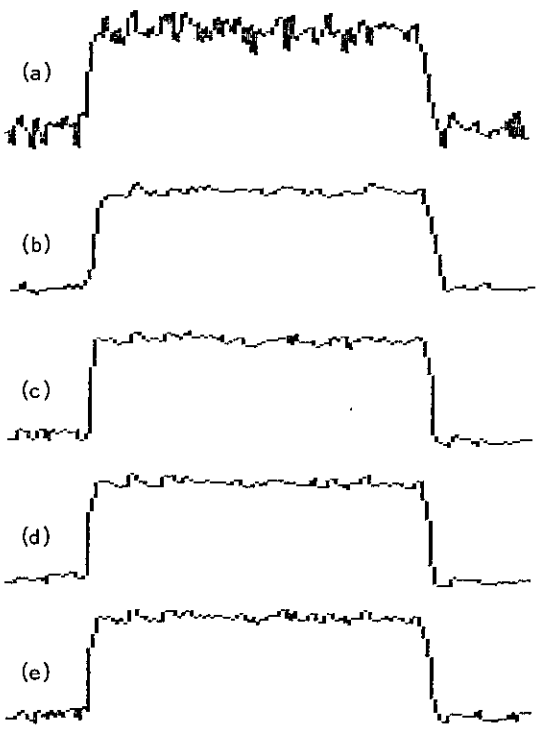


Figure 2. Profiles of scan line 45: (a) blurred circle with noise, (b)-(e) images filtered by MEDIAN, MHNS, FACET, and SADVS.

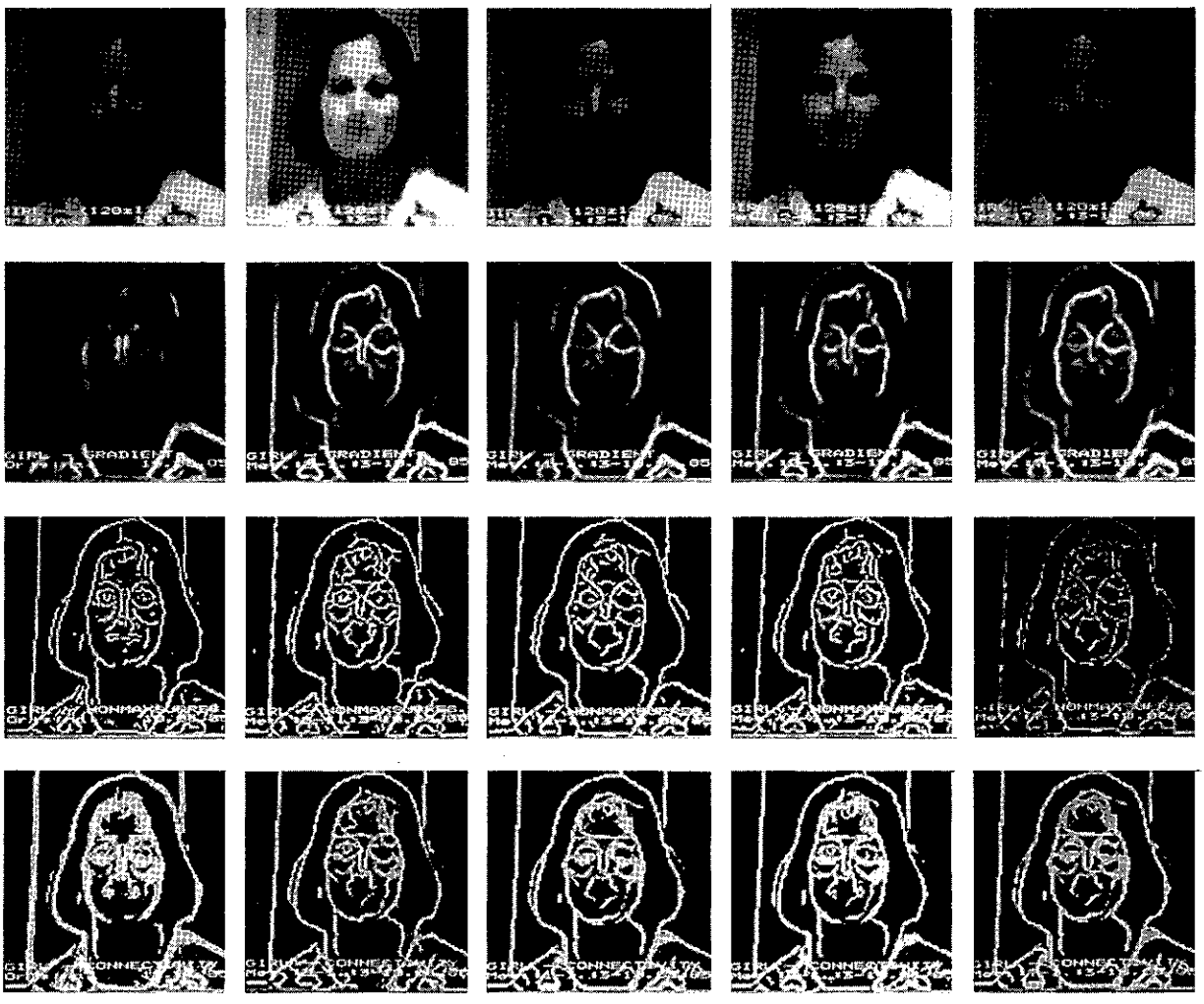
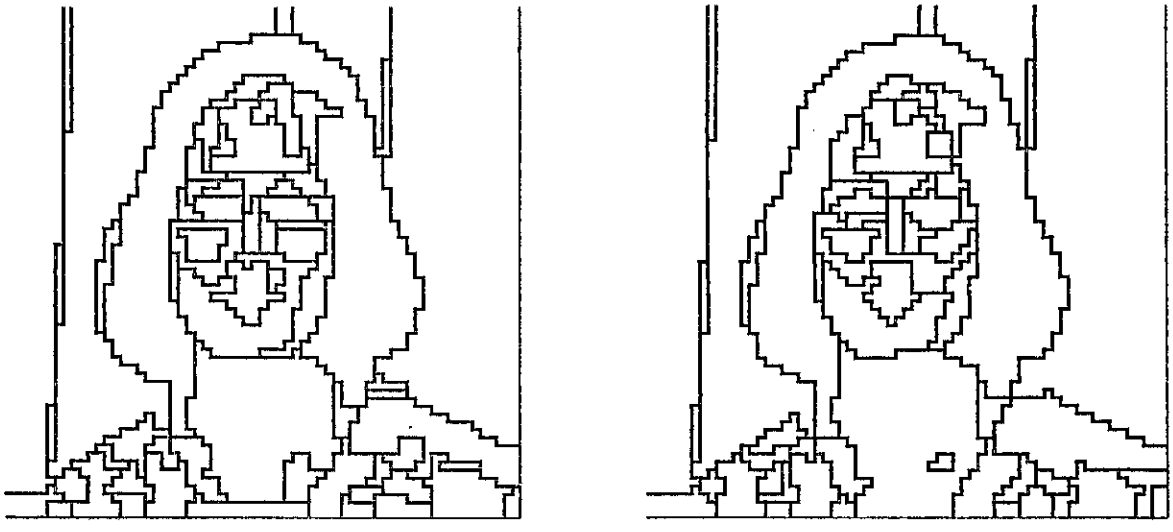


Figure 3.

Figure 4.



4.1. Edge Detection

Parallel gradient operators yield valuable directional information, which is useful in edge detection in digital pictures. Some algorithms have been devised to process the edge vectors derived from the gradient. The resulting output consists of thinned edges which may be tracked very quickly.

In this experiment, two methods were used to obtain a "cleaned" edge map of potencial boundary points on which to do further processing. Both algorithms locate edges by applying a gradient-type edge-detection operator (GRAD) to the image, followed by thresholding (THRES) to eliminate weak edges. To delete redundant responses to a single boundary, the first method uses then a local connectivity test (CON) (8), while the other one uses a thinning algorithm based on nonmaximum suppression (NMS) (9).

Preprocessing by the filters brings out edge reduction in both cases (see Table 2). In Figure 3, line 1 shows the test image GIRL and the results of preprocessing it by MHNS, FACET, SADVS, and XSHARP, line 2 presents the results of the gradient operator application, line 3 shows the results of thinning by NMS, and line 4 presents the results of the connectivity test.

TABLE 2 - GIRL (number of edges)

	GRAD	THRES	NMS	CON
ORIGINAL	16132	4798	2863	3965
MHNS	11677	3809	2232	2762
FACET	12961	3851	1911	3050
SADVS	14383	4014	2042	3102
XSHARP	12961	3851	1911	3050

4.2. Region Detection

A split-and-merge approach to image segmentation is presented in (10). This algorithm is based on recursive subdivision into quadrants. A block image is split into quadrants if it is too inhomogeneous, i.g. if the mean gray levels of its quadrants differ too much. Conversely, four blocks merge into a single block if the result is sufficiently homogeneous, i.e. if the mean gray level values are close enough.

Table 3 presents the detected regions number by applying the split-and-merge approach to segment the test image GIRL before and after preprocessing it by spatial filters. Fig. 4 shows the segmentation of the images filtered by SADVS (left) and MHNS (right).

TABLE 3 - GIRL (number of regions - NOR)

	ORIG	MHNS	FACET	SADVS	XSHARP
NOR	81	62	65	67	62

5. CONCLUSIONS

Sum of absolute difference values smoothing algorithm is relatively fast and simple to implement. It smooths noise well while providing edge preservation and edge sharpening. This method is also almost free of created artifacts. When applyed to segmentation, the results show that preprocessing of the images by the analysed spatial filters reduces the number of elements in the edge map, and the number of detected regions. It remains the question of wether this enhancement is consistent with visual obervation of the segmentation. Application of spatial filters as preprocessing shemes in an ultrasound tissue characterization task is reported in (11). There SADVS brought out an enhancement of up to 100% in the characterization tests.

ACKNOWLEDGEMENTS

The author would like to thank Prof. Dr. -Ing. W. Ameling, Rogowski-Institut, RWTH Aachen, West Germany, for the support of this work, and Dr. J. M. de Carvalho, UFPB, Brazil, for the valuable discussions.

REFERENCES

(1) Araujo, A. de A., Sum of Absolute Grey Level Differences: an Edge-Preserving Smoothing Approach, Electron. Lett., 1985, 21, pp. 1219-1220.
(2) Pratt, W., Digital Image Processing, (Wiley & Sons, N.Y., 1978).
(3) Davis, L.S. and Rosenfeld, A., Noise Cleaning by Iterated Local Averaging, IEEE Trans., 1978, SMC-8, pp. 705-710.
(4) Wang, D.C.C., Vangnucci, A.H., and Li, C.C., Gradient Inverse Weighted Smoothing Scheme and the Evaluation of its Performance, Computer Graphics & Image Processing, 1981, 15, pp. 167-181.
(5) Nagao, M. and Matsuyama, T., Edge Preserving Smoothing, ibid., 1979, 9, pp. 394-407.
(6) Haralick, R.M., and Watson, L., A Facet Model for Image Data, ibid., 1981, 15.
(7) Lester, J.M., Brenner, J.F., and Selles, W.D., Local Transforms for Biomedical Image Analysis, ibid., 1980, 13, pp. 17-30.
(8) Robinson, G.S., Edge Detection by Compass Gradient Masks, ibid., 1977, 6, pp. 492-501
(9) Hong, T.H., Dyer, C.R., and Rosenfeld, A., Texture Primitive Extraction Using an Edge-Based Approach, IEEE Trans., 1980, SMC-8, pp. 659-667.
(10) Horowitz, S.L., and Pavlidis, T., Picture Segmentation by a Tree Traversal Algorithm, Journal of the ACM, 1976, 23, pp. 368-388.
(11) Araujo, A. de A., Kubalski, W., Jensch, P., and Ameling, W., Einflüsse von "Moving-window"-Verfahren auf Texturdiskriminanzeigenschaften in Echokardiogrammen, Proceedings of the 7.DAGM-Symposium, 1985, Springer-Verlag, pp. 213-217.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW DECOVOLUTION METHOD FOR IMAGE RESTORATION IN SPATIAL DOMAIN

You-qui Shi, Jing Lai and Zhi-hong Xu

Graduate School, Academy of Post And Telecommunications
Beijing, China

Abstact

A new deconvolution method for blurred image restoration in spatial domain is presented. If the impulse response of degraded process is known,image restoration can be realized by this method. It avoids the shortcomings of inverse filter. Restoration of blurred image due to different blurred process has been carried out. It proves this is a general and effective method for image restoration.

1. INTRODUCTION

The common method of image restoration in spatial invariant system is inverse filter method.As well known,the inverse filter in absense of noise is

$$M(u,v)=1/H(u,v) \qquad (1)$$

where H(u,v) is tranfer function of degraded process. In presense of noise there are following relation

$$G(u,v)/H(u,v)=F(u,v)+N(u,v)/H(u,v) \qquad (2)$$

where G(u,v),F(u,v)and N(u,v) are the Fourier transform of blurred image g(xy) original image f(x,y) and random noise n(x,y) respectively.Here we can see the drawbacks of inverse filter.First,solution will be lost when H(u,v)=0,because of singularaties.And noise is considerably amplified during the inverse filterring process.Experimental results [1] shown that image restoration by inverse filter is failure,when SNR is low.

Recently,Kong [2] developed a deconvolution method in time domain to find the system impulse response from an observed input and output.Alternatively we can recover input from a known system response and observed output.We extend the theory to spatial domain and proved the theory in 2-D is held.And we describe the new spatial decovolution method for image restoration and give some experimental results.

2. THEORITICAL ANALYSIS

Deconvolution method in spatial domain is the method of finding input of the system from a known output and impulse response of system.Assume the impulse response of 2-D discrete system is $h_o(m,n)$,input is f(m,n) and output is g(m,n),where g(m,n) is NXN series.There is following relation among them

$$g(m,n)=f(m,n)*h_o(m,n)=\sum_k\sum_l f(k,l)h_o(m-k,n-l) \qquad (3)$$

where * denotes convolution operation. The key of decovolution method is to find deconvolution factor $h_r(m,n)$ from a known impulse response $h_o(m,n)$.Now let us introduce the basic idea of this method.

If take

$$h_o^*(m,n)=(-1)^{m+n}h_o(m,n) \qquad (4)$$

then we can construct a new function

$$\begin{aligned}h_1(m,n)&=h_o(m,n)*h_o^*(m,n)\\&=\sum_k\sum_r h_o^*(k,r)h_o(m-k,n-r)\\&=\sum_k\sum_r(-1)^{k+r}h_o(k,r)h_o(m-k,n-r)\end{aligned} \qquad (5)$$

According to exchange property of convolution, we have

$$\begin{aligned}h_1(m,n)&=\sum_k\sum_r h_o(k,r)h_o^*(m-k,n-r)\\&=\sum_k\sum_r h_o(k,r)(-1)^{m-k+n-r}\\&\quad h_o(m-k,n-r)\end{aligned} \qquad (6)$$

By using relation $(-1)^{-k-r}=(-1)^{k+r}$,equation (6) becomes

$$h_1(m,n)=(-1)^{m+n}\sum_k\sum_r(-1)^{k+r}h_o(k,r)\,h_o(m-k,n-r) \qquad (7)$$

Compare equation (5) with (7), it can be obtained

$$h_1(m,n)=(-1)^{m+n}h_1(m,n) \qquad (8)$$

First step: $h_o^*(m,n)=(-1)^{m+n}h_o(m,n)$, its value is

2.00	-9.00	1.00	-15.0	7.00
-3.00	2.00	-2.00	1.00	-5.00
4.00	-6.00	9.00	-4.00	3.00
-8.00	7.00	-3.00	6.00	-1.00
4.00	-2.00	7.00	-5.00	3.00

and $h_1(m,n)=h_o(m,n)*h_o^*(m,n)$, its value is

4.00	.000	-77.0	.000	-241.
.000	-46.0	.000	-118.	.000
7.00	.000	-72.0	.000	-196.
.000	-136.	.000	-260.	.000
-16.0	.000	14.0	.000	-71.0

Here half of total terms are constant-zero terms, there are only one constant-zero term between constant-non-zero terms.

Second step: $h_1^*(m,n)=(-1)^{(m+n)/2}h_1(m,n)$, its value is

4.00	.000	77.0	.000	-241.
.000	46.0	.000	-118.	.000
-7.00	.000	-72.0	.000	196.
.000	-136.	.000	-260.	.000
-16.0	.000	14.0	.000	-71.0

and $h_2(m,n)=h_1(m,n)*h_1^*(m,n)$, its value is

16.0	.000	.000	.000	-7860
.000	.000	.000	-8030	.000
.000	.000	-1610	.000	.000
.000	-444.	.000	.000	.000
-177.	.000	.000	.000	25400

Notice that there are 3 constant-zero terms between constant-non-zero terms.

Third step: $h_2^*(m,n)=(-1)^{(m+n)/2^2}h_2(m,n)$, its value is

16.0	.000	.000	.000	7860
.000	.000	.000	8030	.000
.000	.000	1610	.000	.000
.000	444.	.000	.000	.000
177.	.000	.000	.000	25400

and $h_3(m,n)=h_2(m,n)*h_2^*(m,n)$, its value is

256.	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	-117×10^5

Now the number of constant-zero terms between constant-non-zero terms are 7.

Fourth step: $h_3^*(m,n)=(-1)^{(m+n)/2^3}h_3(m,n)$, its value is

256.	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	117×10^5

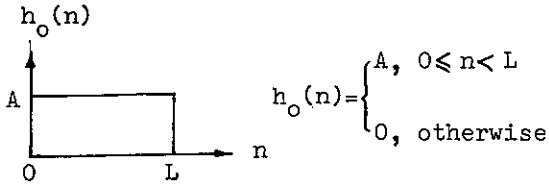
and $h_4(m,n)=h_3(m,n)*h_3^*(m,n)$, its value is

65500	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000
.000	.000	.000	.000	.000

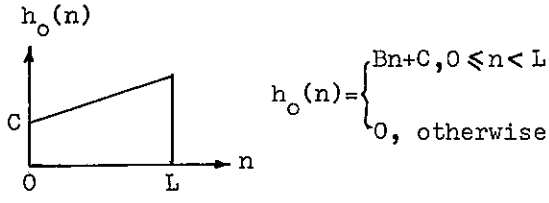
Because the number of $h_o(m,n)$ is $N\times N=5\times5$, here $2^2<N\leqslant2^3$, thus there is only one constant-non-zero term in the first 5×5 terms of $h_4(m,n)$, the other terms all are constant-zero terms. In result, deconvolution has been realized.

3. IMAGE RESTORATION BY DECONVOLUTION METHOD

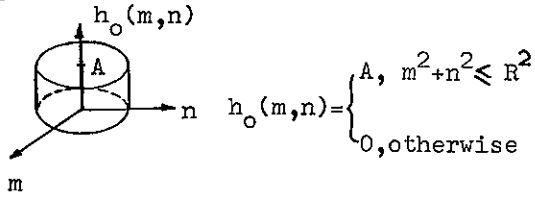
In experiment, we have successfully restored the blurred images due to different types of degradation, i.e. linear motion blur, decelerated motion blur and defocusing blur. Fig.1 shows their impulse responses respectively.



Response of linear motion blur.



Response of decelerated motion blur.



Response of defocusing blur.

Fig.1. The impulse responses of different degraded process.

The restoration procedure is following. At first, for a given degraded factor $h_o(m,n)$, using eq.(9),(10) and (18), the deconvolution factor $h_r(m,n)$ corresponding to $h_o(m,n)$ can be found. Then by eq.(19) the blurred image $g(m,n)$ convolves with $h_r(m,n)$, several restored images with different amplitude can be obtained. They do not overlap with each other.

Equation (8) shows the property of series h_1, i.e. $h_1(m,n)=0$, when m+n is odd number. Repeating above procedure, recurrence formula is following

$$h_i^*(m,n)=(-1)^{(m+n)/2^i}h_i(m,n) \quad (9)$$

$$h_{i+1}(m,n)=h_i(m,n)*h_i^*(m,n)$$
$$=h_o(m,n)*h_o^*(m,n)*h_1^*(m,n)*\cdots$$
$$\cdots*h_i^*(m,n) \quad (10)$$

where $i=0,1,2,\cdots,i+1$ denotes $(i+1)^{th}$ convolution. The new series $h_{i+1}(m,n)$ has an important property, i.e.

$$h_{i+1}(m,n)=(-1)^{(m+n)/2^i}h_{i+1}(m,n) \quad (11)$$

Eq.(11) is the core of spatial deconvolution method. Now we will prove that eq.(11) is held for any natural number by methemetical inductive method:
When i=1, eq.(8) has been proven that eq.(11) is held.
When i=j, assume eq.(11) held, i.e.

$$h_{j+1}(m,n)=(-1)^{(m+n)/2^j}h_{j+1}(m,n) \quad (12)$$

Thus it should be proven that eq.(11) is still held, when i=j+1. According to recurrence formula

$$h_{j+1}^*(m,n)=(-1)^{(m+n)/2^{j+1}}h_{j+1}(m,n) \quad (13)$$

$$h_{j+2}(m,n)=\sum_k\sum_r h_{j+1}^*(k,r)h_{j+1}(m-k,n-r)$$
$$=\sum_k\sum_r (-1)^{(k+r)/2^{j+1}}h_{j+1}(k,r)$$
$$h_{j+1}(m-k,n-r) \quad (14)$$

Similarly, due to exchange property of convolution we have

$$h_{j+2}(m,n)=\sum_k\sum_r h_{j+1}(k,r)h_{j+1}^*(m-k,n-r)$$

By using assumed eq.(12), above eq. becomes

$$h_{j+2}(m,n)=(-1)^{(m+n)/2^{j+1}}\sum_k\sum_r$$
$$(-1)^{(-k-r)/2^{j+1}}(-1)^{(k+r)/2^j}$$
$$h_{j+1}(k,r)h_{j+1}(m-k,n-r)$$
$$=(-1)^{(m+n)/2^{j+1}}\sum_k\sum_r$$
$$(-1)^{(k+r)/2^{j+1}}h_{j+1}(k,r)$$
$$h_{j+1}(m-k,n-r) \quad (15)$$

Comparing eq.(14) with eq.(15), we have

$$h_{j+2}(m,n)=(-1)^{(m+n)/2^{j+1}}h_{j+2}(m,n) \quad (16)$$

Thus we have proven that eq.(11) is held for all natural number i.

From eq.(11) we can see that:

When i=0, eq.(11) becomes eq.(8). It can be found $h_1(m,n)\neq 0$, when m+n is even and $h_1(m,n)=0$, when m+n is odd. Thus half of total terms of series $h_1(m,n)$ are constant-zero terms after first convolution operation and there is a constant-zero term between non-constant zero terms. When i=1, the term where (m+n)/2 is even number is non-constant zero. Besides the term where m+n is odd number is constant-zero term, it adds some constant-zero terms where (m+n)/2 is odd. Now there are 3 constant-zero terms between non-constant zero terms.

According to recurrence formula, along with increasing convolution times, the number of constant-zero terms between non-constant zero terms in the new series after several times convolution also increase. When $2^{I-1}<N\leq 2^I=L$, $I=1,2,\cdots$, then we have

$$h_{I+1}(m,n)=h_o(m,n)*h_o^*(m,n)*h_1^*(m,n)*\cdots$$
$$\cdots*h_I^*(m,n)$$
$$=\sum_a\sum_b A_{ab}\delta(m-aL,n-bL) \quad (17)$$

where A_{ab} is a weighed coefficient, thus the result of deconvolution of system impulse response becomes a delta sequence. And the distance between delta function is not less than N.

Following above deconvolution theorem, it is easy to find the deconvolution factor $h_r(m,n)$

$$h_r(m,n)=h_o^*(m,n)*h_1^*(m,n)*\cdots*h_I^*(m,n) \quad (18)$$

Therefore, the restored signal $\hat{f}(m,n)$ is

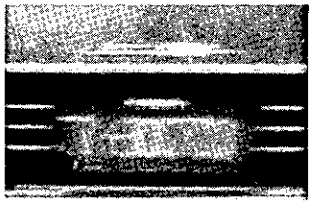
$$\hat{f}(m,n)=g(m,n)*h_r(m,n)$$
$$=f(m,n)*h_o(m,n)*h_r(m,n)$$
$$=f(m,n)*\sum_a\sum_b A_{ab}\delta(m-aL,n-bL) \quad (19)$$

It can be seen from eq.(19) that, for any known degraded impulse response $h_o(m,n)$, the corresponding deblurring function $h_r(m,n)$ can be found. Then by using eq.(19) several separated restored signals with different amplitude can be obtained. It should point out, the restored several inputs do not overlap with each other, because the series of input signal is not larger than N×N.

Example: given a 5×5 discrete series $h_o(m,n)$, deconvolve it in spatial domain. (only take the first 5×5 terms in each convolution result.)
Assume the value of $h_o(m,n)$ is

2.00	9.00	1.00	15.0	7.00
3.00	2.00	2.00	1.00	5.00
4.00	6.00	9.00	4.00	3.00
8.00	7.00	3.00	6.00	1.00
4.00	2.00	7.00	5.00	3.00

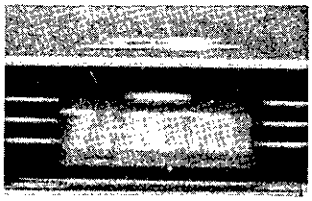
Obvously, restoration process of linear motion blur and decelerated motion blur can be reduced to 1-D deconvolution process. Fig.2 and Fig.3 are the recovered images of linear motion blurred image and decelerated motion blurred image respectively. And restoration of defocusing blur is a 2-D deconvolution process. Computer simulation shown that it also can be restored effectively.



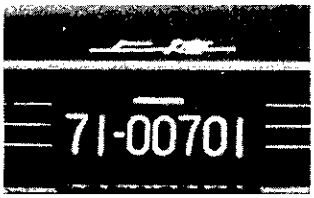
(a)Blurred image

(b)Restored image

Fig.2.Restoration of linear motion blurred image



(a)Blurred image



(b)Restored imge

Fig.3. Restoration of decelerated motion blurred image

4. CONCLUSION

The deconvolution method in spatial domain has following advantages over the inverse filtering method.First,it avoids zero-point problem of transfer function in inverse filter, hence degraded image can be recovered effectively when inverse filter is failure. Second, for any complicated known system impulse response,restoration can be realized. And because there is no noise amplification in deconvolution process, the noise performance is improved. In addition, decovolution algorithm is only related to addition and multiplication of real number without the neccesity of transforming to complex domain, it is apt for hardware implementation. Therefore, it is an general and effective method for image restoration.

REFERENCES

[1] Andrews, H.C. and Hunt, B.R., Digital Image Restoration (Prentice-Hall,1977)

[2] Kong, F.N., Impulse Radar Ph.D.Thesis,Cambridge University, August,1983

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

TWO SOLUTIONS FOR REAL TIME DECODING OF INFRARED IMAGES CODED BY HADAMARD TECHNIQUE

Jean Appel and F. Dunand

Office National d'Etudes et de Recherches Aérospatiales,
BP 72, 92322 Châtillon Cedex, France.

After a recall of Hadamard coding technique for infrared image analysis, two solutions are presented for real time image decoding and visualization. The first one is based on an optical processing technique and the other one is a fully digital system. Advantages and inconvenients of these two solutions are compared.

1 . INTRODUCTION

Three methods of infrared image analysis can be considered:
- the classical method: a flux detector is used, associated with a mechanical, point by point scanning of image
- use of an image detector (mosaïc)
- use of a single flux detector (as in the classical solution), associated with a multiplex method: at each instant, the detector receives information provided by N/2 image elements (N being the total number of elements in the image).

With respect to optimal use of energy, the two last solutions are better than the first one which makes use of each pixel energy only during a time T/N (T being the total time of analysis).

Unfortunately, the second solution uses mosaïcs which are still very expansive devices and which are not available in a very large infrared wave length domain.

Therefore, a multiplex method has been studied and developed at ONERA by A. Girard [1].

The basic idea is to code the N image elements by a set of N orthogonal binary PN functions (PN = pseudo noise functions, used at first by Hadamard in spectroscopy [2]).

As shown on fig.1, the functions are easily materialized on a support transparent to infrared by laying transparent and opaque zones (transmission factor 1 or 0).

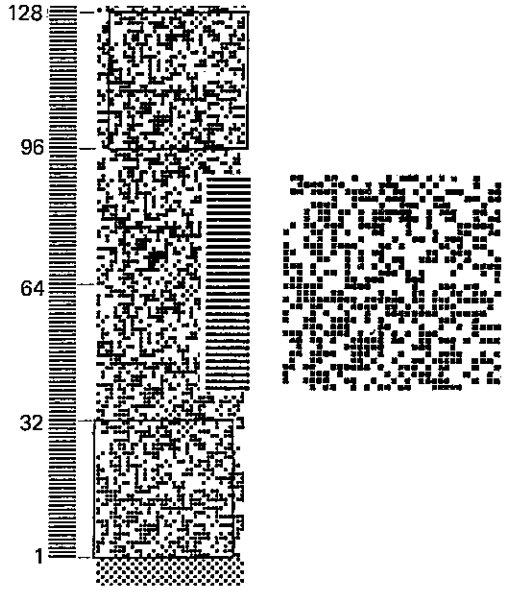


Figure 1 - Coding function.

This code, moving along with time in the plane of the image (fig. 2) realizes a

convolution product between the spatial distribution of the image elements and the code. Due to orthogonal property of PN functions, deconvolution is possible.

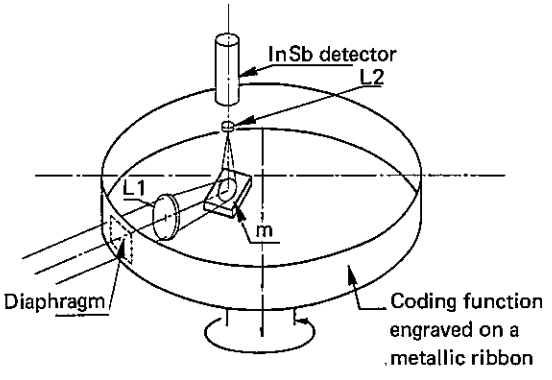


Figure 2 - Principle of the first Hadamard coding system

After decoding and with regard to the classical point by point scanning, a signal to noise ratio improvment of $\sqrt{\frac{N}{2}}$ is obtained.

In many cases, it is useful to have the result of this deconvolution in real time to be able to analyze quickly infrared images given by the camera.

With this object two different ways have been studied at ONERA:
- the first one, not yet tested, is a complete analog optical decoding system which uses a principle allowing to suppress the high continuous background generally inherent to optical processing.
- the second one is a all digital system which has been fully developed and tried.

These two systems will be now described more in details.

2 . PRINCIPLE OF OPTICAL PROCESSING FOR HADAMARD DECODING

It can be shown [1] that, using this PN code, a two dimensionnal image ($\sqrt{N} \times \sqrt{N}$ pixels) is processed as the linear image obtained by putting the $\sqrt{N}$ lines of the image jointly along the same line.

So let us consider $\mathcal{H}(N,1)$ a column matrix representing a N pixels image.

If $\mathcal{L}(N,N)$ is the theoretical coding matrix (with only + 1 and -1 elements), the real coding matrix $\mathcal{C}(N,N)$ is obtained from $\mathcal{L}(N,N)$ by replacing -1 elements by 0 (see [3] for theoretical developments).

Coded image $\mathcal{J}(N,1)$ is given by:

$$(1) \qquad \mathcal{J}(N,1) = \mathcal{C}(N,N) \cdot \mathcal{H}(N,1)$$

A digital decoding allows to use the theoretical decoding matrix $\mathcal{L}(N,N)$, with as result:

$$(2) \qquad \mathcal{L}(N,N) \cdot \mathcal{J}(N,1) = -\frac{N+1}{2}\,\mathcal{H}(N,1)$$

$\sqrt{\frac{N+1}{2}}$ represents the multiplex gain.

An analog optical decoding allows only to process positives signals with positives elements in the decoding matrix. Thus $\mathcal{C}(N,N)$ must be used to decode:

$$\mathcal{C}(N,N) \cdot \mathcal{J}(N,1) = \frac{1}{2}\,\mathcal{P}(N,N) \cdot \mathcal{J}(N,1) + \frac{N+1}{2}\,\mathcal{H}(N,1) \qquad (3)$$

($\mathcal{P}(N,N)$ has all his elements equal to 1).

As shown by (3) a strong continuous background is added to the image which makes difficult to obtain a good contrast. This continuous level $\frac{1}{2}\,\mathcal{P}(N,N) \cdot \mathcal{J}(N,1)$ is roughly equal to $\frac{N^2}{4}$ x the mean level of the image.

Let us suppose now that the coded image has a null mean value (obtained by high pass filtering of signal at the output of the detector).

$\mathcal{J}(N,1)$ can be written*:

$\mathcal{J} = \mathcal{J}+ - \mathcal{J}-$ with $\mathcal{J}+ > 0$ and $\mathcal{J}- > 0$

$\mathcal{J}+$ represents the positive part of $\mathcal{J}$ and $\mathcal{J}-$ the negative one ($\mathcal{J}+$ and $\mathcal{J}-$ are exclusive).

In the same way the theoretical coding

matrix $\mathcal{L}(N, N)$ can be transformed into two positive matrix C and $\bar{C}$ with:

$$2\,C = \mathcal{P} + \mathcal{L}$$
$$2\,\bar{C} = \mathcal{P} - \mathcal{L}$$

C and $\bar{C}$ are complementary matrixes with only 0 and 1 elements.

Let us to decode optically $\mathcal{D}+$ by 2C and $\mathcal{D}-$ by $2\bar{C}$. If we add the two results, we obtain:

$$(\mathcal{P} + \mathcal{L})\mathcal{D}+ + (\mathcal{P} - \mathcal{L})\mathcal{D}- = \mathcal{P}.\mathcal{D}+ + \mathcal{P}\mathcal{D}- + \mathcal{L}(\mathcal{D}+ - \mathcal{D}-)$$
$$= \mathcal{P}(\mathcal{D}+ + \mathcal{D}-) - \frac{N+1}{2}\,\mathcal{H} \quad (4)$$
$$= \mathcal{P}\,|\mathcal{D}| - \frac{N+1}{2}\,\mathcal{H}$$

$|\mathcal{D}|$ is the modulus of $\mathcal{D}$.

$\mathcal{D}$ having presently a null mean value the continuous background is now equal to $\sum_{i=1}^{N} |di|$ (di are the elements of the coded image). This level represents an optimal value just required to ensure the image elements to be positive.
Figure 3 shows the principle of a system which can realize these operations.

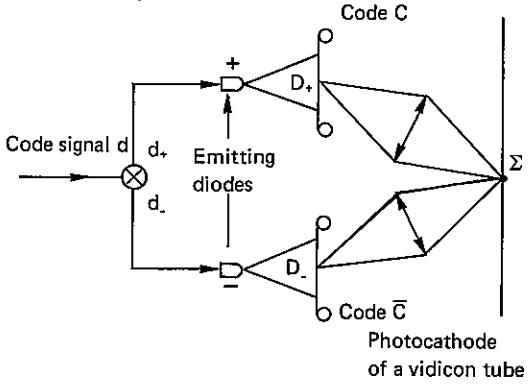


Figure 3 - Principle of the analog optical decoding system.

Coded signal d is separated into d+ and d- respectively sended to the input of the emitting diodes. $\mathcal{D}+$ and $\mathcal{D}-$ corresponding fluxes pass through the C and $\bar{C}$ codes.

The two code areas are integrated on the photocathod of a vidicon tube (with an integration time of the same order as the image analysis time). It is also possible to imagine that coding and decoding are realized by the same physical code and that C and $\bar{C}$ codes are obtained one by transmission and the other one by reflexion.

Unfortunately these solutions have not yet be tested.

3 . A DIGITAL REAL TIME PROCESS SYSTEM FOR HADAMARD DECODING

The digital solution has been fully developed and is now used in association with an experimental infrared Hadamard camera as a quick look system.

The figure 4 illustrates the principle of the digital system.

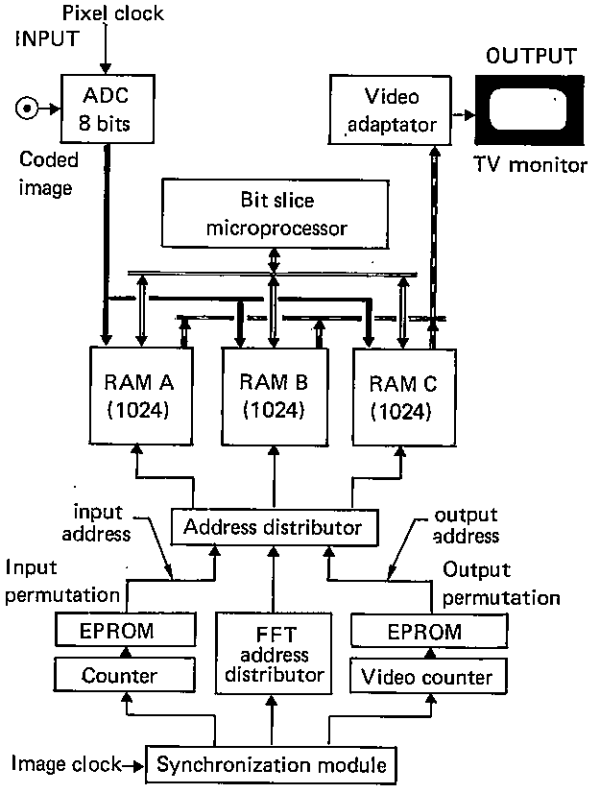


Figure 4 - Block diagram of the digital real time decoding system.

The algorithm used to decode is similar to fast Fourier Transform algorithm with allows a time gain of $N/\log_2 N$ (100 for a 32 x 32 pixels image). To be used in Hadamard decoding, this algorithm needs a special input data permutation and an other one for the output.

The system is built around 3 RAM memories which are circulary permuted, for each new image to be processed. When the first memory acquires a new digitized image, the second exchanges data with a bit slice microprocessor for F.H.T (Fast Hadamard Transform) and the third is read through a video adaptator for real time TV visualization.

Addresses for Fast Hadamard Transform algorithm are provided by an one chip FFT address generator. Input and output addresses are provided by two independant counters, through two EPROM memories that ensure respectively input and output permutations.

This system used at 25 or less images of 32 x 32 pixels per second could be used up to 50 images per second.

4 - CONCLUSIONS

The digital solution is certainly less limited in precision than the analog one, but also more bulky, at the actual state of art for circuit integration. An analog optical solution would be more artful and would probably lead to a more compact device.

* *This technic is described in the french patent n 2.289.973 - Optische Industrie - "De oude delft" - Netherland*

REFERENCES

[1] Girard, A., Analyse d'image et codage multiplex, *Propagation limitations in remote sensing*, AGARD Conf. Proc., 1971, n°90, Memoire n° 33

[2] Nelson E.D., Fredman M.L., Hadamard spectroscopy, J. Opt. Soc. Amer., 1970, 60, 1664.

[3] Appel J. and Girard A., Analyse d'image par codage optique multiplex, Nouvelle Revue d'Optique, tome 7, n° 4, pp. 221-134.

BIOGRAPHY

Jean Appel received the degree of Dipl. Engin. (1966) in electromechanics from the Ecole Spéciale des Travaux Publics in Paris, France. He joined ONERA (Office National d'Etudes et de Recherches Aérospatiales) in 1968 to develop data acquisition systems for laboratories, testing benches and wind tunnels. Others activities are real-time data processing and digital data filtering. He is now co-head of the "Electronic and Measurment" division in the Physics department.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMAGE DATA COMPRESSION TECHNIQUES USING KALMAN AND ALPHA-BETA FILTERS

Giuliano BENELLI, Romano FANTACCI

Dipartimento di Ingegneria Elettronica, Università di Firenze,
Via S. Marta 3 - 50139 Florence, Italy

Image data compression techniques usually aim at an optimal trade-off between efficiency and implementation simplicity, according to the user's needs. In this paper, the application of the Kalman and alpha-beta filters as predictors for image data compression algorithms is considered. In particular, the optimum values for the alpha-beta filter parameters which guarantee the highest performance, are derived.

1. INTRODUCTION

In this paper, image data compression techniques by using the Kalman and α-β filtering are presented. These methods can be considered as a modified DPCM coding method; indeed, in this case, the prediction of the actual pixel is obtained through a recursive strategy, following the well-known Kalman and α-β filtering theories [1]. The initialization procedure influences rather critically the performance, i.e. the compression ratio values of the presented image data compression techniques. The initialization of the Kalman and α-β filters is performed for each line of the image by using the first two pixels. Thus, the steps of the image data compression algorithms described in this paper are the following: i) initialization of the algorithm; ii) filtering and prediction; iii) quantization of the error sample, i.e. the difference between the actual pixel and its prediction obtained through the predictive algorithm at the previous step.

In order to compare the image quality, fixed and adaptive quantizers have been applied to the error samples. The application of an adaptive quantizer, with the same number of quantization levels, offers substantial advantages in comparison with the fixed quantizer. Adaptivity has been achieved by varying the quantizer representation levels by means of the bidimensional Jayant algorithm [2].

2. KALMAN FILTERING FOR IMAGE DATA COMPRESSION

In this section, the state model (i.e. image representation) and the Kalman filtering theory are shortly presented. The image state can be expressed by the following equations:

$$x(k+1) = \Phi x(k) + G u(k) \quad (1)$$

$$y(k) = Hx(k) + v(k) \quad (2)$$

where x(i) is a vector with two components defined as:

$$(3)\ \begin{bmatrix} \text{grey level of the actual pixel (i-th)} \\ \text{grey level variation referred to the grey level of the previous pixel (i-1 - th)} \end{bmatrix}$$

The term Φ represents a 2x2 matrix given by:

$$(4)\ \Phi = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

The term u(i) represents a disturbance on the evolution of the state according to the hypothesis of a constant grey level difference between consecutive pixels. This disturbance is considered as white and Gaussian with zero mean and variance σ_u^2 [1]. The term v(i) represents a disturbance on the acquisition of the actual pixel; v(i) is considered as a white Gaussian noise with zero mean, and variance σ_v^2, according to the Kalman filtering theory [1]. Therefore, in the following, the contribution of v(i) can be considered as negligible and an ideal acquisition of the actual pixel is assumed. The last two terms G and H in eqs. (1) and (2) are defined, respectively, as:

$$(5)\ G = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$(6)\ H = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

In the image model described above we have assumed the hypothesis of a time invariant (t.i) system. Starting from the above model the Kalman filter permits the achievement of a filtered estimate of the grey level of the actual pixel and a prediction of the grey level of the following pixel. The Kalman filtering equations are:

$$(7)\ \hat{x}(i/i) = \hat{x}(i/i-1) + K(i)[y(i) - H\hat{x}(i/i-1)]$$

$$(8)\ \hat{x}(i+1/i) = \Phi \hat{x}(i/i)$$

where $\hat{x}(i/i)$ is the filtered estimate of the i-th pixel state, $\hat{x}(i+1/i)$ the predicted estimate of the i-th pixel state derived at the i-th step, and K(i) the gain matrix of the filter derived according to the Kalman filtering algorithm [1] referred at the i-th step.

The block diagram of the implementation structure of the Kalman filter is shown in Fig. 1. It can be noted that the use of this algorithm

requires the evaluation of the gain matrix K at each step. Then, a suitable initialization must be effected because the performance of this algorithm depends rather critically on this operation. In this application, the Kalman filtering algorithm is initialized at the beginning of each line of the image by using the following state equation:

$$(9)\quad \hat{x}(2/2)=\begin{bmatrix}\text{grey level of the actual pixel (second)}\\ \text{grey level variation between the second and the first pixel}\end{bmatrix}$$

the other initialization equations are derived starting from eqs. (1), (2), (7), (8) and (9) according to the Kalman filtering theory [1]. The block diagrams of the transmitter and receiver for the implementation of the image data compression algorithm based on the Kalman filtering are shown in Figs. 2 and 3, respectively. The efficiency of the image data compression algorithm based on the Kalman filtering can be evaluated by considering some typical images such as, for example, remote sensing, artistic and biomedical images. Therefore, in sect. 4 only the results obtained by considering an artistic image are presented.

3. IMAGE DATA COMPRESSION BY USING THE ALPHA-BETA ALGORITHM

The image data compression algorithm based on the Kalman filtering requires a high implementation complexity and for some applications, when the system parameters σ_U^2 and σ_V^2 are not properly chosen, it does not represent a suitable solution [1].

In this section, the application of the α-β algorithm to image data compression is considered. It is well known that this algorithm has been introduced in order to reduce the overall implementation complexity with respect to the Kalman filtering and the dependence on the system parameters. The image-state model is that represented by eqs. (1) and (2); thus, the α-β filter equations are given by eqs. (7) and (8) where the gain matrix K(i) is given by:

$$(10)\quad K(i)=\begin{bmatrix}\alpha\\ \beta\end{bmatrix}$$

The parameters α and β can be chosen in order to guarantee a fixed performance. For the image data compression application, we have considered α as an independent variable, while β is given by:

$$(11)\quad \beta = \alpha^2/(2-2)$$

where α can be determined in order to obtain a good performance as will be shown in sect. 4. It can be pointed out from eq. (10) that the α-β algorithm does not require the evaluation of the gain matrix K at each step; indeed, it is constant for each step. The initialization procedure is required also for the α-β algorithm; therefore, in this case, it is simpler. The α-β algorithm requires the initialization for the beginning of each line of the image; in this case, it can be obtained only through eq. (9). The block diagrams of the transmitter and receiver for the image data compression system described in this section are shown in Figs. 4 and 5, respectively. The results obtained by using this image data compression algorithm will be presented in sect. 4; in particular, it will be shown that an optimum value of the parameter α, when β is given by eq. (11), can be found for each processed image.

4. RESULTS

In this section, the results obtained by using the image data compression algorithms described in sects. 2 and 3 are presented. In Fig. 6, a sequence of images processed through the image data compression algorithm based on the Kalman filtering is shown. The original image represents Raffaello's self-portrait and was quantized by using a bidimensional adaptive quantization algorithm (3 levels) [2]. For this image the signal-to-noise ratio is 21.55 dB; the signal-to-noise ratio expressed as amplitude normalized ratio (ANE) is defined as:

$$(12)\quad ANE = 20 \log \left(\frac{255}{MSE}\right)\ dB$$

where MSE represents the mean square error for the processed image, equal to 28.70 dB with a compression ratio (CR) equal to 2.98. In Fig. 7, a sequence of images processed through the image data compression algorithm based on the α-β filter is shown. The original image is the same as that in Fig. 6. In Fig. 8, the parameter ANE versus the parameter α of the α-β filter is shown for different values of the number of levels L used for the bidimensional quantization. We recall that the parameter β is given by eq. (11). It can be noted that for all L an optimum value of α exists; this value permits the achievement of the highest ANE and, in general, of an optimum performance. Finally, it can be pointed out that a higher compression ratio can be obtained by using the image data compression algorithm presented herein, jointly with a suitable source coding (i.e. adaptive Huffman coding).

5. CONCLUSIONS

In this paper, the application of Kalman and α-β filters as predictors for image data compression techniques has been considered. In particular, the results which have been obtained show that a suitable choice of the α-β filter parameters could give an optimum performance. As a possible development of this study, we point out the opportunity of applying the image data compression algorithms presented herein, jointly with a suitable source code such as, for example, the adaptive Huffman code. This could improve the algorithms efficiency achieving higher compression ratios.

REFERENCES

[1] Singer, R.A., and Frost, P.A., On the Rela-

tive Performance of the Kalman and Wiener Filters, IEEE Trans. on Aut. Contr., vol. AC-16, Aug. 1969.

[2] Zatterberg, L.H., and Ericson, S., DPCM Picture Coding with Two-Dimensional Control of Adaptive Quantization, IEEE Trans. on Comm., April 1984.

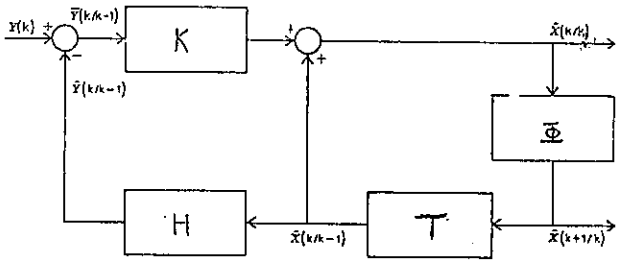


Fig. 1

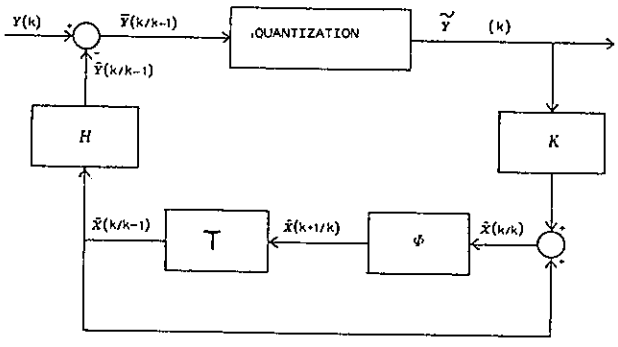


Fig. 2

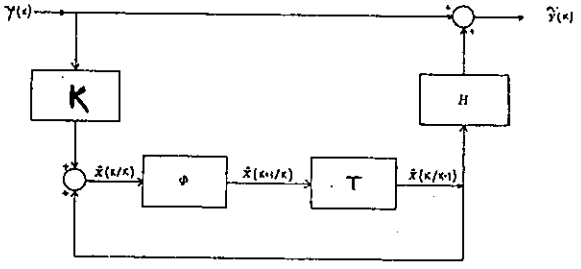


Fig. 3

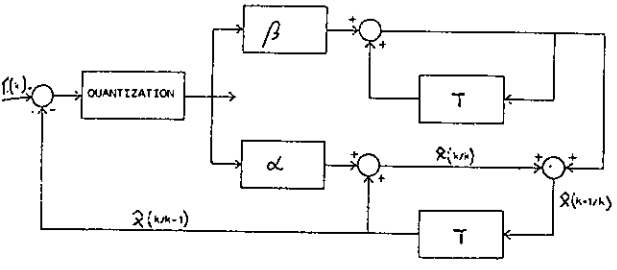


Fig. 4

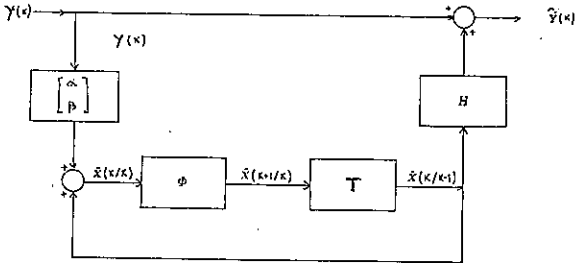


Fig. 5

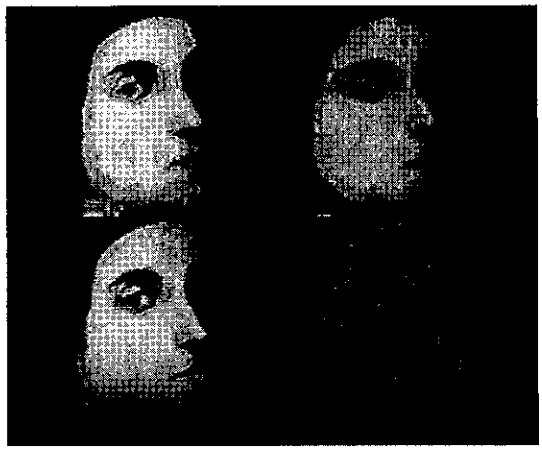


Fig. 6 - a) original image; b) predicted image;
c) restored image; d) transmitted image.

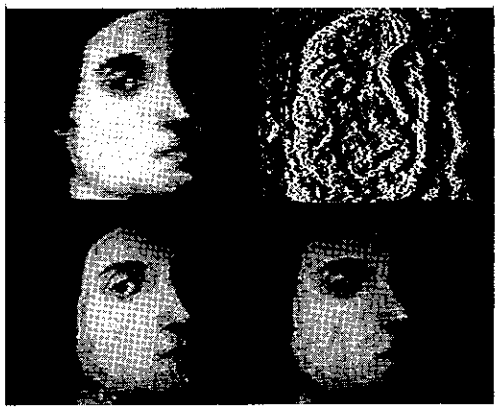


Fig. 7 - a) original image; b) transmitted image
c) restored image; d) predicted image.

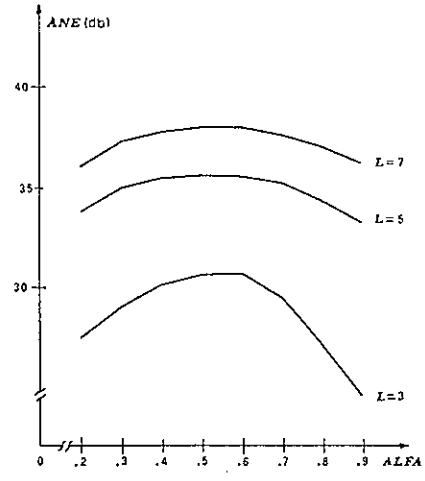


Fig. 8

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ADAPTIVE TRANSFORM CODING OF IMAGES USING VECTOR QUANTIZATION

Manfred Götze and Yonggang Du

Institut für Elektrische Nachrichtentechnik
Technical University of Aachen
Melatener Str. 23
D-5100 Aachen, Fed. Rep. of Germany

An efficient adaptive transform coding system for monochrome images using vector quantization technique is described. Practical system application is attained by decomposing the 8x8 spectral domain into several separated subblocks. For given transmission rate and coder complexity an iterative algorithm is developed to determine the decomposition of the spectral domain as well as the size of the different codebooks for each subblock. Both the decomposition of the spectral domain into subblocks and the decomposed VQ for them are adapted to the activity and prevailing direction of the subpicture structure. Theoretical and experimental results show that the gain of VQ in the spectral domain strongly depends on the angle of subpicture structure. For oblique structures a gain of up to 3 dB is achieved by using binary tree search VQ in comparison with the conventional scalar quantization.

1. INTRODUCTION

Vector quantization (VQ) and transform coding (TC) are blockwise source coding methods widely used for image data reduction. For good reconstruction quality, processing in relatively large blocksizes (e.g. 8x8 or 16x16 pixels in TC) is required in order to exploit as much interpixel redundancy as possible at reasonable computional effort. To avoid the disadvantage of VQ, namely the substantially increased number of multiplications required for large block dimensions, a variety of suboptimal VQ techniques have been developed [1].

TC, although not originated from VQ and thoroughly investigated much earlier, can simply be derived as such kind of suboptimal VQ technique [2],[3]. In TC the number of multiplications is strictly reduced at the cost of decreased reconstruction quality, which is due to the regular codebook structure even in adaptive TC systems.

The gap between TC and VQ can be closed by VQ in small subblocks in the spectral domain of an orthonormal transform (TC-VQ). As the gain achieved by VQ is at the cost of additional computations, VQ in the spectral domain should only be applied effectively to exploit remaining statistical dependencies.

As has been reported earlier [4], due to restricted decorrelation properties the DCT yields only reduced performance for orientated structures other than horizontal or vertical (chapter 2). Chapter 3 describes a TC system with adaptive VQ in the DCT domain to exploit the remaining correlations [3], as well as an approach to optimize this proposed ATC-VQ system. The better performance in comparison with conventional adaptive TC is presented in chapter 4.

2. DIRECTIONAL DEPENDENCE OF THE DCT PERFORMANCE

We assume that the second order statistics of an 8x8 block can be modeled by a covariance matrix $COV(\underline{X},\underline{X})$ in the original domain, which contains the elements

$$\mathrm{cov}(\Delta i,\Delta j)= \sigma_x^2 \cdot \exp\{-(\alpha_s^2 \cdot |\Delta i \cdot \cos\varphi + \Delta j \cdot \sin\varphi|^2 + \alpha_p^2 \cdot |-\Delta i \cdot \sin\varphi + \Delta j \cdot \cos\varphi|^2)^{1/2}\}, \quad (1)$$

$\Delta i = k-i$; $\Delta j = l-j$; $i,j,k,l = 1,\ldots,N$.

The angle φ is assigned to the prevailing direction of the structure; the correlation factors in principal and secondary direction are $\rho_p = \exp(-\alpha_p)$ and $\rho_s = \exp(-\alpha_s)$, respectively.

Let $\underline{T}$ be the transform matrix, the covariance matrix $COV(\underline{Y},\underline{Y})$ of the spectral coefficients is obtained from

$$COV(\underline{Y},\underline{Y}) = \underline{T} \cdot COV(\underline{X},\underline{X}) \cdot \underline{T}. \quad (2)$$

Since $COV(\underline{X},\underline{X})$ is a function of φ, the spectral variances σ_{yij}^2 are functions of φ as well.

The performance of the transform is evaluated by the criterion of signal energy concentration. The concentration gain G_o [5] is defined as a

function of the binary logarithm values of the variances in the original and spectral domain,

$$G_o = \frac{1}{2N} \sum_{i=1}^{N} \sum_{j=1}^{N} lb(\sigma^2_{xij}/\sigma^2_{yij}) \text{ bit.} \quad (3)$$

Due to the angle dependence of σ^2_{yij} we get from (3) the following curves of G_o.

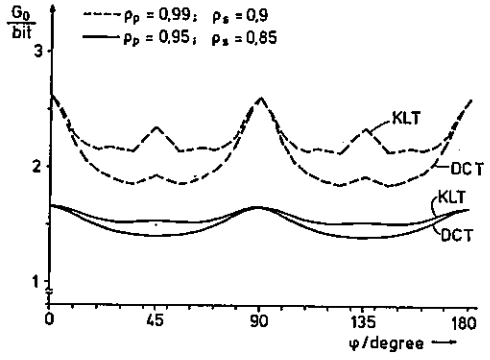


Fig. 1: concentration gain for DCT and KLT [4]

An estimation value of the achievable gain in signal to noise ratio is given by

$$SNR \approx 6 \cdot [G_o(KLT) - G_o(DCT)] \text{ dB.} \quad (4)$$

Fig. 1 shows, in respect of exploitable linear statistical dependencies in the DCT-domain, for which angles φ of orientation a VQ achieves reasonable improvements, namely for diagonal and halfdiagonal, but not for horizontal and vertical structures.

3. ADAPTIVE DECOMPOSED VECTOR QUANTIZATION OF THE DCT-COEFFICIENTS

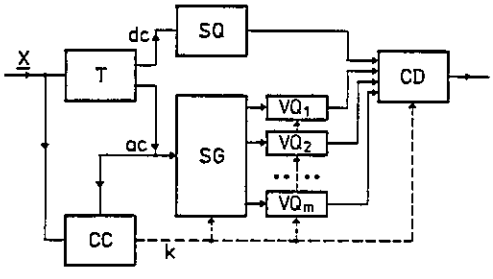


Fig. 2: block diagram of the ATC-VQ system

T	transform unit	SG	subblocks generator
SQ	scalar quantizer	CC	classification unit
VQi	VQ for i-th subblock	CD	coding unit

The block diagram of the proposed coding system is shown in Fig. 2. The spectral ac-coefficients are assembled together in the subblock generator (SG) to m disjunct subblocks SBKi, i=1,...,m, with small dimensions to enable VQ. The dc-coefficient is processed independently in a uniform scalar quantizer (SQ). The decomposition into the SBKs is adapted to the statistical dependencies of the coefficients (angle-orientation of local image structure, cp. Fig. 1). For the directional classification of each transform block 4 variables are computed in CC, which denote the mean absolute differences between adjacent pels in horizontal and vertical

$$\underline{\Delta}_M = [\bar{\Delta}_h, \bar{\Delta}_v]^t \quad (5)$$

as well as principal diagonal and secondary diagonal directions

$$\underline{\Delta}_D = [\bar{\Delta}_{pd}, \bar{\Delta}_{sd}]^t . \quad (6)$$

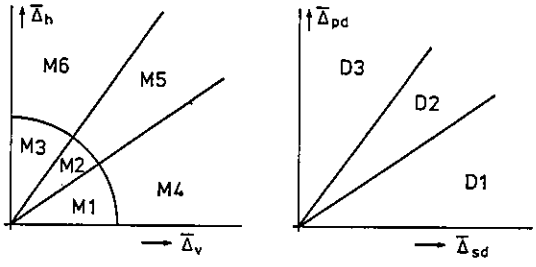


Fig. 3: division of the classification areas

classification area	activity and direction of structure (in degree)
K1 = M6 X D2	high activity ; 0
K2 = M6 X D3	high activity ; 20
K3 = M5 X D3	high activity ; 45
K4 = M4 X D3	high activity ; 70
K5 = M4 X D2	high activity ; 90
K6 = M4 X D1	high activity ; 110
K7 = M5 X D1	high activity ; 135
K8 = M6 X D1	high activity ; 160
K9 = M5 X D2	high activity ; mixed
K10 = M3 X (μ=1)	medium activity ; 0
K11 = M2 X (μ=1)	medium activity ; mixed
K12 = M1 X (μ=1)	medium activity ; 90
K13 = (μ=0)	low activity

Tab.1: classification areas and corresponding structures

To check the perceptibility of luminance change within each 8x8 subpicture an additional variable μ is determined from the following definition

$$\mu = \begin{cases} 0 & \text{if } |y_{ij}| < w_{ij} \;,\; \forall\, i,j \text{ except } i=j=1 \\ 1 & \text{else,} \end{cases} \quad (7)$$

where w_{ij} is the perception threshold of the (i,j)-th DCT coefficient [6]. If μ=0, only the dc-coefficient is transmitted.

Dividing the area vectors $\underline{\Delta}_M$ and $\underline{\Delta}_D$ into the sections M1 to M6 and D1 to D3 respectively, 13 disjunct classes are defined if Mi, Dj and μ are combined as in Tab. 1.

The optimization of the coding system in Fig. 2 is a multidimensional problem. Since the gain of VQ increases with higher statistical dependencies between the vector components, here the following approach was used: decompose the transform block into m disjunct SBKs such, that the coefficients are

- highly correlated within a SBK
- only weakly correlated with coefficients in other SBKs.

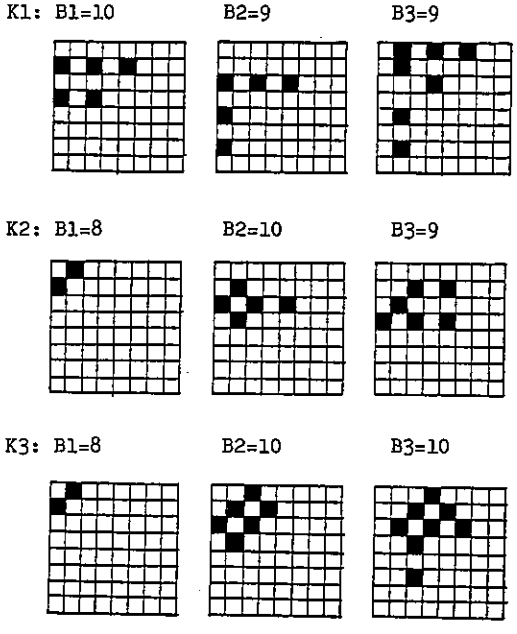


Fig. 4: results of decomposition and bit assignment [7]
(Bi: number of bits for SBKi)

Thereby the optimization constraints are:

- minimal reconstruction error MSE
- given total transmission rate and computational effort.

Suboptimal results were iteratively found by alternative application of two independent procedures [7]

PROC1: find good decomposition SBKi for given bit assignment, i=1,...,m
PROC2: find optimal bit assignment Bi for given decomposition, i=1,...,m

PROC1 is a heuristic method based on a simple trial and error search. PROC2 is an extension of a bit assignment procedure [9] from one- to multidimensional quantization.

As an example of optimization result, the first three SBKi and bit numbers Bi are illustrated in Fig. 4 for the classes K1 to K3 (cp. Tab. 1). The results for classes K4 to K8 can be obtained by transposing the corresponding matrices of SBK-patterns.

Since PROC1 requires much CPU-time, the alternative repeat of PROC1 and PROC2 is not a practicable solution of optimization for different data rates. Hence, in our investigations, PROC1 was only repeated for the maximal total bit number of each class k. For smaller total bit numbers the results of the decomposition were maintained, only the procedure for bit assignment was carried out again.

4. EFFICIENCY AND REALIZATION EXPENSE

The codebooks for the computer simulations were generated from a set of 15 different natural pictures. The simulation results were obtained taking pictures from outside this training sequence; all VQ was carried out applying binary tree search.

4.1 Comparison between VQ and ATC-VQ

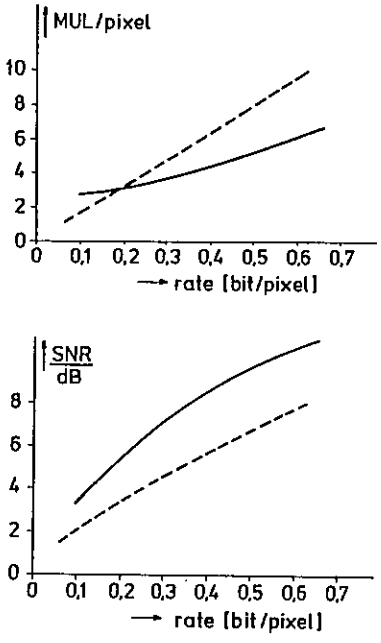


Fig. 5: number of multiplications MUL and SNR for
- - - VQ in original domain (4X4)
——— ATC-VQ (8X8)
(dc-term not included in both cases)

The curves in Fig. 5 show that the proposed ATC-VQ system enables processing in larger blocksize than conventional VQ, thus yielding better performance at given computational requirement. Considering the additional transmission of the dc-term omitted here, the difference would even be more significant.

The required memory size is approximately the same for both methods. In Fig. 5 two multiplications per pixel are assumed to perform the DCT [8].

4.2 Comparison between conventional ATC and ATC-VQ

To investigate the efficiency of the coding system in Fig. 2 the specific gain

$$\Delta SNR = SNR(ATC\text{-}VQ) - SNR(ATC) \qquad (8)$$

has been calculated for each class, where ATC is the corresponding adaptive TC system with scalar quantization. The scalar quantizers have been optimized according to the Lloyd-Max-conditions for a fitting generalized exponential distribution [2]. Fig. 6 shows the specific gain for the first eight classes, obtained from the test image "Resolution chart" and the natural image "Boats". Obviously, the simulation results correspond to the theoretical results shown in Fig. 1; remarkable gain is only achieved for diagonal and halfdiagonal directions, this has also been verified by subjective comparison 3 .

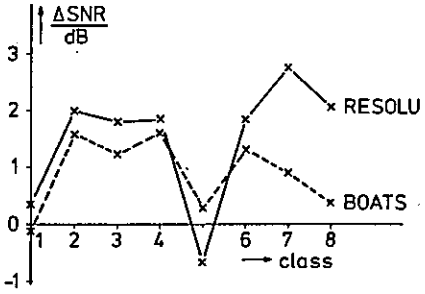


Fig. 6: gain in SNR for ATC-VQ relative to ATC

The high specific gains for the image "Resolution chart" in the range of 2.5 dB are due to the fact that in this image the subpictures contain only distinct edges and lines, whereas in the natural image "Boats" these structures appear less pure. The additional number of multiplications required by the VQ is about 2.4 on the average for the picture "Boats".

5. CONCLUSIONS

The coding system investigated here achieves improved performance of the DCT for diagonal and halfdiagonal image structures by separated VQ in the spectral domain. The activity and prevailing direction of image structure are measured for each transform block, so that a decomposition of the image source into several subsources with different statistical features is obtained. Depending on these features an adaptation of the different spectral VQs is carried out: The decomposition of the spectral domain into the subblocks as well as the size of the different codebooks depend on the spectral variances and correlations. This coder concept combines several suboptimal VQ techniques: multistage VQ, VQ with separated mean, VQ with segmented codebooks, tree search, and adaptation of vector dimension. Coder performance and computational effort are in the range between those of multistage VQ and transform coding.

The concept of this coding system can easily be extended for color image coding. It also offers the possibility to use subjective optimization criteria based on spectral error weighting functions. The use of fixed codeword lengths so far implies high robustness against transmission errors and easy error protection facilities [2], or alternatively the possibility of additional redundancy reduction.

REFERENCES:

[1] Gray, R.M.; Vector Quantization, IEEE ASSP Magazine, vol. 1, no. 2 (1984) pp. 4-29
[2] Götze, M.; Kombinierte Quellen- und Kanalcodierung in adaptiven Transformationscodierungssystemen (VDI-Verlag, Düsseldorf, Reihe 10, vol. 52, 1986)
[3] Götze, M.; Adaptive vector quantization of images in the discrete cosine transform domain, Picture Coding Symposium 86, Tokyo, Japan (1986) pp. 142-143
[4] Götze, M., Peifer, H.J.; Image Coding by Direction Adapted Transforms, Second Europ. Conf. on Signal Proc. EUSIPCO-83, Erlangen, W. Germany (1983) pp. 179-182
[5] Mauersberger, W.; Comparing orthonormal matrices for two-dimensional image transform coding by means of the gain distortion function, Signal Processing 2 (1980) pp. 67-70
[6] Lohscheller, H.; Einzelbildübertragung mit wachsender Auflösung, Ph. D. thesis, Inst. f. El. Nachrichtentechnik, Techn. Univ. of Aachen (1982)
[7] Du, Y.; Adaptive Transformationscodierung von Bildsignalen mit vektorieller Quantisierung im Spektralbereich, diploma thesis, Inst. f. Elektr. Nachrichtentechnik, Techn. Univ. of Aachen (1985)
[8] Kamangar, F.A., Rao, K.R.; Fast algorithm for the 2-D discrete cosine transform, IEEE Trans. COM-25 (1977) pp. 1004-1009
[9] Guglielmo, M.; Bit-assignment procedure for blocks of uncorrelated random variables, CSELT Rapporti tecnici 4 (1975) pp. 63-67

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW PROCEDURE FOR IMAGE VECTORISATION

Guillermo CISNEROS (*,+) and Narciso GARCÍA (#,+)

(*) Telefónica, Res. & Develop. Dpt., C/Lérida 43, 28020-Madrid, Spain
(#) IBM Madrid Scientific Center, P. Castellana 4, 28046-Madrid, Spain
(+) E.T.S.Ing.Telecomunicación, Univ. Politécnica, 28040-Madrid, Spain

An algorithm for image vectorisation is presented. Raster images are converted into binary ones, and transformed into a set of vectors, after either a thinning operation or an edge extraction. A general scheme for the whole process of the vectorisation is proposed and a new method for the main step, where the coding with vectors takes place, is developed. The most important features of the algorithm are the exclusive use of logical operations and the reduction with a previous processing of all the particular situations to the general case, increasing efficiency and accuracy, and avoiding errors. Several examples of its application are shown.

1. INTRODUCTION

A vectorisation can be defined as the operation which transforms an image into a set of vectors [4]. This technique can be related with those of edge and skeleton encoding. Thus, the vectorisation is an operation which must be applied to images having structures with predominant directions.

Two vectorisation schemes can be defined depending on the desired result. The original image is segmented obtaining a binary one. The first scheme performs a thinning obtaining the structures skeletons. The second extracts the binary edges, thinning special crossing environments afterwards. At last, a smoothing over the vectors is made as a post processing of the vectorisation. Usually, the segmentation is simplified if an image enhancement is applied before. The edge detection is more efficient applied to a binary image than to the original one.

Thus, the complete vectorisation process can be decomposed into the following steps, where the asterisk means optional:

1.- Image Enhancement (*)
2.- Edge Extraction in Multilevel Images (*)
3.- Segmentation
4.- Edge Extraction in Binary Images (*)
5.- Thinning
6.- Vectorisation
7.- Smoothing (included in the previous step)

Several algorithms have been proposed before for carrying out the mentioned steps [2,3,4,5,6,7,9], and a similar scheme for the whole process can be found in [8,10]. Here, a new algorithm to solve the sixth step is presented, consisting of the definition of a hierarchy among the pixels of the thinned image [10]. The algorithm is based on two definitions: the range and the environment, and the implementation uses 8-connectivity. Particular situations are solved with a special algorithm applied previously to the vectorisation process, transforming them into the general case. This feature increases efficiency and accuracy.

2. DESCRIPTION OF THE METHOD

2.1 Previous Definitions

The proposed vectorisation algorithm is based on the definition of two parameters for every pixel: the range and the environment. The implementation uses 8-connectivity.

The range of a pixel is defined as a value equivalent to the number of non-zero neighbor pixels, unused by the vectorisation process. Therefore, the range diminishes as the processing evolves. According to the connectivity, the value of the range is a number between 0 and 8. Several images containing range information are defined. The initial ranges image contains the ranges computed prior to the beginning of the vectorisation process. The current ranges image starts with the value of the initial one, and contains the current ranges values at every time of the process. It stops the processing when all its elements are null. The modified ranges image also starts with the value of the initial one, and holds the modifications required to process pathological environments.

The environment of a pixel is defined as an octal value mask indicating the position of the neighbor pixels, previously used by the vectorising process [2,3]. The environment image is created having every element a decimal integer value equivalent to the octal value of the mask of the corresponding pixel at the original ima-

ge. It is updated as the vectorisation processing advances.

2.2 The core of the algorithm

Every pixel belonging only to one line has an initial range equal to 2, a knot has an initial range bigger than 2, and a line-end has an initial range of 1. Pixels having zero initial range are not processed. The vectorisation connects the non-null pixels with lines.

The procedure starts using the pixels with maximum range (value equal to 8) as the initial ones for each line. The line is formed applying a fan searching from the last included pixel until an output is found. A new pixel is added to the line, and the current range of the two involved pixels is decreased by one. The following situations can take place:

1.- A real end of line is found if the current and initial ranges are equal to one.
2.- A pixel belonging only to the built line is found if its three ranges (initial, current, and modified) equal to 2. The searching procedure goes on only in this situation.
3.- An end of line is also marked if the current range is bigger than 2. So, the smoothing post-processing must lean on the knots.
4.- An end of line is also marked if the current range is equal to 1, but the initial one is bigger. This corresponds to a exhausted knot.

After completing all the lines starting with 8-range pixels, the maximum possible current range is 7.

The update of the range and the environment takes place simultaneously to the connection of a pixel pair. As told before, the range is decremented by one in both pixels. In the case of the environment, two auxiliary masks are defined, taking as central pixel for each one, the two pixels to be connected. These masks are null except in the place corresponding to the neighbor joined pixel. The update is made with an inclusive "or" operation between the old mask and the corresponding auxiliary one for each one of the two pixels.

The procedure is repeated starting with pixels having a current range equal to 7. After finishing, it is repeated for 6, 5, 4, and 3. Now, a line is started from a pixel with current range equal to 2, only if the initial range is higher than 2. Then, pixels having a current range equal to 1 are considered as starting points of lines. Now, there are only pixels having current range equal to 2 or to 0, corresponding to closed lines, that can be analised starting from any point.

Before joining two pixels with a segment, once it has been determined the need to join, a smoothing operation must be made with the previous segments of the line currently creating.

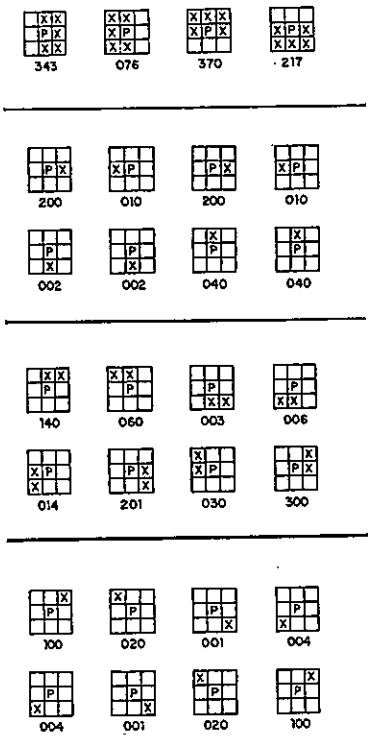


Fig.1 Sets of masks for particular situations

This operation is made using two thresholds, applying a new method designed for this purpose based on vectorial operations [10]. A use of vectors for the smoothing was previously outlined in [6], differing the applied algorithm. The smoothing is very important, because it is the only way to avoid the generation of a lot of very short segments, fully aligned, as components of a straight line.

2.3 Particular Situations

Some problems can arise in particular situations. A processing is performed previous to the vectorisation, once the initial ranges have been computed. The environment of masks is put to zero initially. The mask used to begin the vectorisation will not be null, as theoretically, but it will be the new mask resulting from this previous processing. The ranges are also changed with respect to the real situation. These changes could force an error in the smoothing, if the modified range is not used. The processing which solves all the particular situations with no errors is developed into the following steps:

1.- The 4 sets of test masks shown in Figure 1 are defined.

2.- A logical "and" is made between the environment mask in the thinned image and the test masks of the first set.

3.- If the result is equal to the test mask, the initial and current ranges are artificially increased by the number of masks which yield the mentioned result. If the range is not increased, some posterior errors can take place, due to the tests which will be made afterwards decrease the range. The pixels in that situation verify more than one of the posterior test, and would finish without range.

4.- For every pixel, a logical "and" is made between the environment mask in the thinned image and the test masks of the third set.

5.- Every time the result is equal to the test mask, a new logical "and" is made between the same environment mask of the pixel in the thinned image and the corresponding test mask of the second set to that of the third set which yields the mentioned result, and the three ranges are reduced one unity.

6.- If the result of the last logical "and" is not equal to the test mask, a logical "or" operation is made between the vectorisation environment mask and the test mask of the fourth set corresponding to the test mask of the third set.

7.- If the result of the last logical "and" is equal to the test mask, a logical "or" operation is made too, but between the vectorisation environment mask and the test mask of the second set corresponding to the test mask of the third set.

The option of processing the particular situations on the basis of false ranges and environments is the best one. It substitutes their post processing by a previous one, with no errors, and with a not common quickness in the most of the vectorisation algorithms.

2.4 Storage Formats

The goal of a vectorisation is to encode the skeletons or the edges of the structures of the image with vectors stored in two databases: the Universe and the Pointers [1,4]. The first one contains all the vertices of the vectors, and, the second one sets of pointers over the elements of the Universe, one for every line.

The Universe is a compact database, because the only important thing is the order of its elements for the pointer of the second database is able to appoint them. Its format is not specially important.

In the database of Pointers, there is a list of pointers for every line of the image, with a suitable format. This one consists of the use of two records for every structure. The first one contains information about the number of pointers belonging to the list of the structure, and some additional informations such as if the line is a branch, a trunck, an isolated line,..., its lenght, its graphic characteristics for posterior display,..., all of them coded. The second record contains the list of pointers of the structure.

2.5 Final Coding of the Structures

The final database of the vectors contains in the protocol record of every line some informations about it. Moreover, the number of its straight segments and the kind of graphic display, other informations are incorporated here, such as if it is a branch, an isolated line,... A line is a branch if the initial range of any of its extreme pixels is equal to 1, while the other extreme has an initial range greater than 2. A line is a trunck if the initial range of any of its extreme points is greater than 2. A line is isolated and opened if the initial range of any of its extreme points is equal to 1. It will be isolated but closed if the two extreme pixels have a initial range equal to 2.

3. RESULTS

The edges of the original image of a dog shown in Figure 2 have been vectorised displaying the result in Figure 3. The skeletons of another example of an original image in Figure 4 have been also vectorised and the result is shown in Figure 5.

4. CONCLUSIONS

A vectorisation scheme has been proposed, and an algorithm has been developed to encode the structures with vectors. Its main features are as follows:

1.- The whole algorithm has been carried out using only logical operations.

2.- The particular situations are processed before beginning the vectorisation, and reduced to the general case.

These characteristics make the method very efficient and the results are obtained with great accuracy and without errors.

REFERENCES

[1] J.Jiménez and J.L.Navalón, "The Structure of Queries on Geometric Data", in **Data Base Techniques for Pictorial Applica-**

tions", A.Blaser ed., Springer Verlag, No.6, pp.219-232, Berlin, 1979.
[2] T.Pavlidis, "A Thinning Algorithm for Discrete Binary Images", Computer Vision, Graphics and Image Processing, Vol.13, No.2, pp.142-157, 1980.
[3] T.Pavlidis, "Algorithms for Graphics and Image Processing", Springer Verlag, Berlin, 1982.
[4] J.Jiménez and J.L.Navalón, "Some Experiments in Image Vectorisation", IBM Journal of Research and Development, Vol.26, No.6, pp.724-734, 1982.
[5] M.Nadler, "Document Segmentation and Coding Techniques", Computer Vision, Graphics and Image Processing, Vol.28, No.2, pp.240-262, 1984.
[6] K.Wall and Per-Erik Danielson, "A Fast Sequential Method for Polygonal Approximation of Digitized Curves", Computer Vision, Graphics and Image Processing, Vol.28, No.2, pp.220-227, 1984.
[7] R.M.Haralick and L.G.Shapiro, "Image Segmentation Techniques", Computer Vision, Graphics and Image Processing, Vol.29, No.1, pp.100-132, 1985.
[8] M.S.Landy and Y.Cohen, "Vectorgraph Coding: Efficient Coding of Line Drawings", Computer Vision, Graphics and Image Processing, Vol.30, No.3, pp.331-334, 1985.
[9] F.Cohen, "Adaptative Hierarchical Algorithm for Accurate Image Segmentation", Proceedings of the ICASSP85, Vol.2, pp.897-900, Tampa (Florida-USA), 1985.
[10] G.Cisneros, "Tratamiento y Representación de Estructuras Gráficas", Doctoral Thesis, Universidad Politécnica de Madrid, 1986.

Fig.2 Original image

Fig.3 Vectorisation of edges

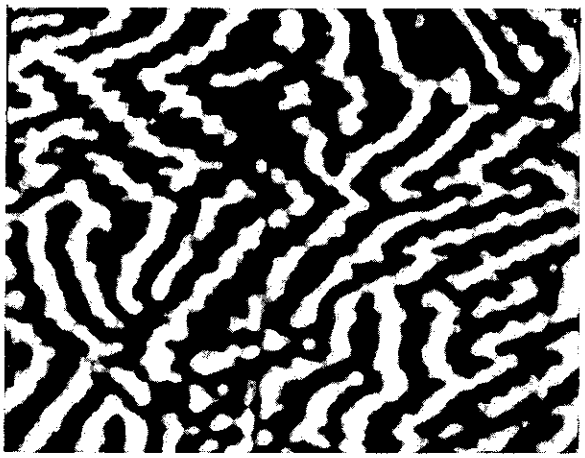


Fig.4 Original image

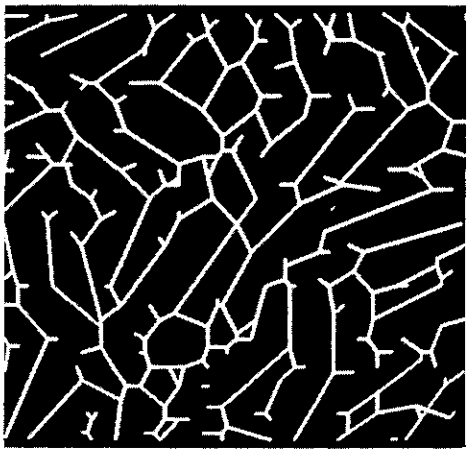


Fig.5 Vectorisation of skeletons

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOME RESULTS ON VECTOR QUANTIZATION

Luis ALVAREZ and Narciso GARCIA

E.T.S.Ing. Telecomunicación, Univ. Politécnica, 28040 Madrid, Spain
and
IBM Madrid Scientific Center, Po. Castellana 4, 28046 Madrid, Spain

A vector quantization system for digital images has been designed. The quantization depends on the algorithm that generates the codebook. This codebook is developed by choosing a random initial codebook and afterwards running the algorithm. Other important parameters used in quantization have also been analyzed. We have studied the vector training sequence behavior and proved that the improvement of the quantizer quality is quite limited although using greater sequences to generate the codebook. We have also verified the results of dimension influence on the final image quality. The greater the dimension is, the better the result obtained. Another interesting point is the relationship between the theoretical bit rate and the real one taking in account the effect caused by the transmission of codebooks. Some graphics on this topic have been obtained.

1. INTRODUCTION

The growing demand on image storage and transmission is drawing in continuous research of new techniques enabling to reduce costs, being compression one of the most promising ones. The compression of information produces either a smaller data volume (storage) or lower bit rate minimizing channel capacity (transmission). As it is necessary to maintain the adequate fidelity to the initial data, several fidelity criteria are defined (1). Although subjective criteria (based on human vision) offer better results in distortion measurement, their cost and the time needed to obtain them, lead to the use of objective criteria. There exists several different methods, but the most accepted is the mean square error (m.s.e.). The performance of a compression algorithm is measured by its data compression capability, the resulting distortion and its facility to be implemented.

Encoding can be modeled as a sequence of three stages (2):

- Transformation. An appropriate representation of the image is made.
- Quantization. The accuracy of the representation is reduced.
- Codification. The statistical redundancy of the image is eliminated.

Only stage two is intrinsically irreversible and so, responsible for the quality loss of the overall process. The decoder is composed only by the inverse operators of the first and third stages.

Let us consider transform encoding techniques. They operate dividing the image in blocks of elements and considering each one as an independent entity. A transform is applied over the block obtaining a representation on a different domain, whose elements are called coefficients. The purpose is to concentrate the energy (information) in the minimum amount of coefficients as possible. The better this objective is achieved, the higher the compression can be obtained.

Retaining only the significant coefficients, an approximation of the image is obtained. Data volume can be reduced, while maintaining a close resemblance to the image (small m.s.e.). The most popular transform is the cosine transform, being its main feature to avoid symmetries. As the encoding technique processes the blocks in an independent way, it is very usual to apply postprocessing techniques to improve quality of the results.

2. VECTOR QUANTIZATION

Vector quantization (VQ) (3,4) is a transform method based on the conversion of each block of K elements (pixels) into a vector. It is then quantized comparing it to a set of code vectors (codebook). The nearest codevector to the original one is selected as the quantized one. As the codebook is kept, only the codevector index inside the codebook is required to univocally identify the quantized vector. The decoder holds the same codebook, (previously sent by the coder) and on receiving the index, it assigns the appropriate vector, filling the corresponding area on the decodes image. The distortion of the encoding process is the sum of the differences between the original vectors and their quantized replica. Depending on the distance measure adopted, different distortions can be obtained, existing a trade-off between minimum distance and complexity.

Formally, a vector quantizer consists of a set of N K-dimensional codevectors plus a function F(x) which assigns a code vector y=F(x) to each input vector, x=x(0),x(1).....x(k-1) using a distortion measure d(x,y). The codevector selected must be the one that minimizes the value: d(x,y). This vector will produce the lower distortion, if the distortion measure is properly selected. The simplest and widest used distortion measurement is the mean square error with:

$$d(x,y)= \sum_{0}^{k-1}(x(i)-y(i))^2$$

There are several measuring criteria. Most of them significantly increase the complexity needed in calculus and implementation. Using the definition mentioned above, the number of bits required for index transmission (N is the number of codevectors) is log2(N) and as each vector contains K pixels the resultant theoretical bit rate is R=log2(N)/K. Although vector quantization is a quite recent technique it has developed a considerable number of different methods (5).

Here, a spatial vector quantizer is analyzed. The first step in VQ design is codebook design. How must the set of vectors be distributed, how many vectors does the codebook contain, and how to improve initial codebook with iterative methods are questions that the designer musts solve. They have great importance because they influence in a decisive way to the final quality. Therefore, codebook design is not only the first problem, but the basic one too, being the Linde, Buzo and Gray (LBG) algorithm the most used one (6). A modification of it has been employed here.

The first decision to be taken is the election of the vector dimension K and the total number of codevectors N. A larger block length, a higher dimension, optimizes VQ, but at the same time increases the operations required. The bit rate R=log2(N)/K is limited by the acceptable distortion degree. The square distortion has been chosen as a measure mainly because its simplicity and relative good performance.

The codebook generation algorithm begins its process with some input information and one initial set. Input information may be given through the knowledge about vectors distribution function. This approach is only feasible if a known distribution is being encoded. The usual situation in image processing is not to know that function, therefore being necessary to apply another design strategy. Instead of using the above mentioned function, it can be simulated taking an important amount of information on the input image. An approach is to extract a vector sequence, but the selection must be carefully done. If the first n vectors in the image are taken, it is clear that the representativity has not been achieved.

3. MAIN CONSIDERATIONS

The image presented in figure 1 (512x512 pixels and 8 bpp) has been selected to analyze encoder performance. A random training sequence of image vectors has been taken in this work, as a partial knowledge about the distribution all over the image. Once this sequence is randomly prepared, the codebook design is started. Beginning with N of that n training vectors and using the LBG algorithm the codebook is designed. The first question that must be solved is how great is the influence of the number of training vectors during the design. Some trials have been done in order to answer that question. For example with blocks of 4 x 3 pixels (k=12) and a codebook of N=128 codevectors (it means R=0.58 bpp) the results were:

TRAINING VECTORS	MSE (total image)
500	4.27 %
1000	4.15 %
2000	3.93 %
3000	3.66 %
5000	3.50 %

As can be seen the improvement produced by using a greater number of training vectors is significant.

Once the codebook is created the performance must be analyzed. The way to do it is through certain interesting quantizer parameters. Some of them have been already chosen, as vector dimension is. This is k (k=12) and as it was supposed the behavior of square or near square blocks is better than rectangular ones. Why 4 x 3 then? It is because its performance is very similar to the 4 x 4 blocks but the number of calculations required is slightly lower. The reason that leads to square blocks is quite

Figure 1. Original image: (512x512 pixels, 8 bpp).

clear: as transform methods explode signal correlations is better to take blocks where correlation between pixels is expected to be greater. Another interesting parameter is image dimension. As mentioned before we have used both 512 x 512 and 256 x 256. The first one means a higher accuracy in signal representation which leads to better results after image processing. It is very usual to talk about VQ results in terms of bit rate which is one of the best parameters to measure vector quantizer performance. The other essential parameter is signal quality. However, there is something that must be taken into account. When it is said that resultant bit rate after quantization is R must be also specified if it includes or not the codebook. As can be seen in figure 2 it is quite different. In both graphics the X-axis represents log2(N) where N is the number of codevectors used in the quantization. The straight lines represents the theoretical bit rate R:

$$R=\log2(N)/k$$

where k is the block size (vector dimension). The curves corresponds to the expression:

$$R=\log2(N)/k + b * k * N /DIM$$

that gives the real bit rate including the second term which adds the effect caused by the transmission of the codebook. In these formula b is the number of bits per pixel in the original image and DIM the total number of pixels in the image (512 x 512 or 256 x 256). If we carefully observe the curves we can see that for large codebooks the real and the theoretical bit rate tend to diverge. This effect is stronger as image dimension decreases. For images with a greater number of pixels (higher accuracy) the real curve keeps nearer to the theoretical one than in the others but they also need a greater amount of calculations. These graphics show the improvements in terms of bit rate added by using greater block sizes (case of 4 and 16 pixels). Anyway it can be seen that if the codebook must be transmitted (curves) the real and the theoretical (lines) bit rate can not be considered as the same thing. This is very important because although the use of greater codebooks will improve quality it can produce bit rates (number of bpp) higher than the original image has. Of course this can be avoid by using the same codebook for quantizing different images. In case the images are similar (for example, not with great changes in the kind of vectors that form the image) the same codebook may be used without been necessary to design another for the new one.

It is easy to see that in images with sharp edges, there will be a lot of image vectors with a sharp change in the value of their elements. This is produced by the well defined image edges which should need a codebook adequate to its characteristics. Such set of vectors is very different form the codebook designed for the Pamela image where there was very few sharp

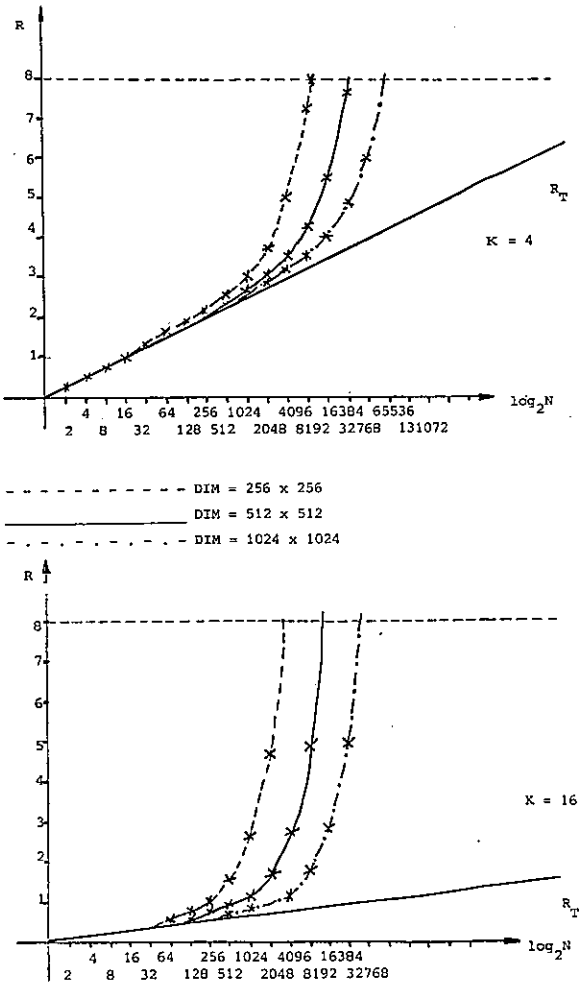


Figure 2. Bit rate curves: R: bit rate, N: codebook dimension.

edges. In sharp-edged images, we can see the needing of creating a set of blocks involving edges only and other set with shape blocks.This will perform special codebooks where every input vector must be classified as an edge or shape one.

That is why we are trying in the use of different codebooks for edges and shapes. The performance obtained is rather good than with simple codebooks. By the way, not enough results are available to give figures on this topic. We can ensure that the quality is enhanced using the same bit rate.

4. RESULTS AND DISCUSSION

The image shown in figure 3 has been quantized using a codebook of only 64 codevectors and blocks of 16 elements (4 x 4) which gives a bit rate of R=0.375 bpp, and including the codebook

Figure 3. Coding at 0.375 bpp: N: 64, Block: 4x4.

Figure 4. Coding at 0.59 bpp: N: 700, Block: 4x4.

R=0.4 bpp. As can be seen the quality is not good but it is enough when the only requirement in transmission is to recognize the image with not very much quality. The next example (fig.4) gives much more quality using also square blocks of 16 elements (4 x.4) with 700 codevectors and a training sequence of 2000 (not very large). It means R=0.59 bpp without the codebook and R=0.93 bpp including it. A detailed look at the image shows how the worst quality is in the edges, specially where lines are light one on a dark background. Finally, it is possible to compare in figure 5 a relevant zone in the image, the eye. In the left up corner the original eye quite well defined with pixels easily visible. Right up corner contains an example of quantization using 2 x 2 blocks. It gives higher quality but the bit rate grows up to 1.75 bpp. Edges are better represented than with greater blocks. Right down corner has the 0.375 bpp image and gives poor quality, edge are very bad drawn. The last corner is the 0.59 bpp version,

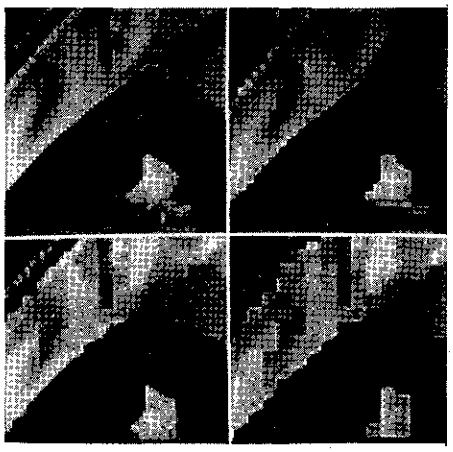


Figure 5. Detail of differents encodings: Upper left: original image, Upper right: 1.75 bpp, Lower left: 0.59 bpp, and Lower right: 0.375 bpp.

giving quite good quality and a very interesting bit rate in order to use compression in useful applications.

Some data on how vector quantization should impact in image processing are: the original image needs 256 kbytes of memory to be stored. The image quantized (R=0.59 bpp) needs less than 20 kbytes if the codebook is not considered and 35 kbytes including it. This means a reduction of about a 90 % in the first case and a 85 % including the codebook. If transmission is considered, to transmit the original image at 64 kb/s would take almost 33 seconds while transmitting the quantized image 2.5 seconds (if the codebook is included the less than 4.5 sec).

REFERENCES

1. D.J.Granrath, "The Role of Human Vision Models in Image Processing", **Proc. I.E.E.E.**, Vol. 69, No. 5, pp. 552-561, May 1981.
2. A.N.Netravali and J.O.Limb, "Picture Coding: A Review", **Proc. I.E.E.E.**, Vol. 68, No. 3, pp. 366-406, March 1980.
3. R.M.Gray, "Vector Quantization", **I.E.E.E. Acoust., Speech, and Signal Process. Mag.**, Vol. 1, No. 2, pp. 4-29, April 1984.
4. A.Gersho and B.Ramamurthi, "Image Coding Using Vector Quantization", **Proc. ICASSP-82**, Paris, April 1982, pp. 428-432.
5. N.M.Nasrabadi, "Use of Vector Quantizers in Image Coding", **Proc. ICASSP-85**, Tampa, March 1985, pp. 125-127.
6. Y.Linde, A. Buzo, and R.M.Gray, "An Algorithm for Vector Quantizer Design", **I.E.E.E. Trans. Commun.**, Vol. 28, No. 1, pp. 84-95, January 1980.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A TRIANGULATION ALGORITHM FOR SURFACE DISPLAY IN BIOMEDICAL ENGINEERING

A. EKOULE, F. PEYRIN, C ODET

Laboratoire de traitement du signal et ultrasons. UA 1216 CNRS
INSA Bat 502 69621 Villeurbanne Cedex FRANCE

In many fields of medecine, generation and display of surfaces of three dimensional objects from contour lines can be helpfull.After a review of some triangulation methods we propose a new heuristic algorithm trying to include the problem of branching contours. The linking procedure betxeen two single contours is first presented. Then the problem of branching a contour to multiple contours is solved by using a new interpolated contour between two slices.At last an illustraion of the algorithm is presented.

1. INTRODUCTION

The generation and display of surfaces from contour lines can be usefull in medical applications and particularly in computer asssisted tomography (CAT). Indeed CAT allows to reconstruct slices of the human body. Generally the single observation of one slice is not sufficient for diagnosis and the tomographic exam is repeated at different levels. The stack of these different cross-sections can be used to create a three dimensional array representing the scanned volume. Note that the so obtained volume can suffer of some distorsions. A first reason is that the resolution between the slices is generally lower than the resolution within the slice. This effect can be partly corrected by interpolating new slices. A second problem is that the patient must stay motionless during all the duration of the tomographic exams. If not, the cross-sections will not correctly be aligned.Most of these drawbacks can be eliminated using truly three dimensional tomography (see for instance the Dynamic Spatial Reconstructor of Mayo Clinic [1]).
Whatever is the procedure used, it may be interesting to display the surface of a feature of interest whithin the reconstructed volume. Such a display can be helpfull for surgery,radiation therapy planning...
The surface can be created using a triangulation algorithm which connects each pair of adjacent contours by a set of triangular patches By this way a polyhedral approximation of the surface is obtained. This representation can be conveniently handled and displayed.
In this paper we describe a triangulation algorithm. Special attention is given to the problem of multiple contours. In the first section we review some triangulation methods and give their advantages and limitations. In the second part we present a new algorithm. An illustration of the method applied to the generation of a simulated surface is presented on the last section.

2. SURFACE TRIANGULATION ALGORITHM

The general problem of surface triangulation from a set of points distributed in space has been studied by O'Rourke [2],Hermeline [3],Boissanat [4] ..However for the display of tomographic data,the problem can be restricted to the case where the points lie in a set of parallel slices. This more particular problem has first been studied by Keppel [5] for radiation therapy planning.
In Keppel's work the contours are first divided in a set of convex sub-contours The triangulation procedure between two convex sets is defined in order to maximize the volume of the constructed polyhedron It leads to an optimal approximation of the surface relatively to this triangulation criterion. The problem is then formalized in terms of graph theory and shown to be equivalent to find a minimum-cost path in a graph The complexity of the algoritm is MN O(MN) where M and N are respectively the numbers of points on each contour
The same approach has been generalized by Fuchs [6] who solves the graph theory problem exploiting some particular properties of the graph under investigation. Using this method the complexity of the algorithm becomes $\log_2 M$ O(MN).
In these two works the contours are supposed to be simple closed contours.
An heuristic tiangulation method reducing the complexity of the algorithm has been developed by Christiansen [7]. The triangular patches are constructed using a shortest-diagonal criterion. More precisely let $P_1,P_2,\ldots P_M$ and $Q_1,Q_2,\ldots Q_N$ be the set of ordered points in two successive slices.If P_iQ_j is the previously chosen segment, the next one which is either P_iQ_{j+1} or Q_jP_{i+1} is the shortest. The problem of branching two contours to one is also examined in some simple situations.
Other heuristic methods have been proposed by Cook [8] [9] or Ganapathy [10]

It appears from these methods that most often the complexity of the algorithm is reduced when using an heuristic method, but that good displays are obtained only when the successive contours are not too different.
The proposed heuristic algorithm tries to solve the problem of branching multiple contours which has scarcely been discussed.

3.ALGORITHM

We start from a set of contours C_k at different levels z_k obtained for instance from a 2D contour detection algorithm. These contours are first resampled in order to reduce the number of processed points. This is realised using a polygonal approximation ofthe contours so that more points are kept in regions with a high curvature.
After this preliminary work, the problem is to link the M points of slice z_{k-1} (contour C_{k-1}) to the N points of slice z_k (contour C_k). We detail the two following situations:
- link between single closed contours in each slice,
- link from a single closed contour in a slice to multiple contours in the other slice.
The problem of linking multiple contours in each slices is discussed.

3.1. Link between two single contours.

The triangulation procedure is obtained by creating an ordered list of edges each extremity of which belongs to one of the contour. In order to define a set of adjacent triangular patches, this list must possess the two following properties:
- two successive edges have a unique common vertice either on C_{k-1} or on C_k and they define a triangular patch,
- three successive edges define two adjacent triangular patches.

The list of edges is obtained by a two steps methods. In a first step each point of the contour having the smaller number of points, say for instance C_{k-1} (i.e. M<N) is connected to the nearest point in the other contour point. If P_i is a point of contour C_{k-1} we note $Q_{j(i)}$ the point of contour C_k to which it is connected. At the second step the free points of contour C_k are connected according to a local minimum-length edge criterion. More precisely let $S_i = \{ Q_j / j \in [j(i),j(i+1)] \}$ be the set of points lying between $Q_{j(i)}$ and $Q_{j(i+1)}$ (Figure 1). Its points are sequentially exlored to be connected either to P_i or to P_{i+1} . Each point Q_j in S_i is connected to P_i as long as it is the closest i.e. $d(Q_j,P_i) \leq d(Q_j,P_{i+1})$. Let Q_q be the first point in S_i closer to P_{i+1} i.e. $d(Q_q,P_i) > d(Q_q,P_{i+1})$. All the remaining points $\{ Q_j / j \in [q,j(i+1)[\}$ are linked to P_{i+1} to avoid crossing edges. The processing of S_i is over when a last edge is generated. It is choosen as the smallest between $Q_{q-1}P_{i+1}$ and Q_qP_i.

This algorithm has some similarities with Christiansen's algorithm [7]. However in [7] the determination of a new edge uses a local criterion entirely depending of the previous edge. Then if one edge is not very reliable it can introduce an increasing distortion as the process goes on. As we use a global connection between the points in the first step this problem is attenuated. Furthermore, because of this problem Christiansen's algorithm gives better results when it is initialized with the shortest edge between the two contours [8]. In this case the complexity of the two algorithm is of the same order. Note that if the contours are too different in size it can be usefull to preprocess them in order to scale them into unit lengths as described in [7].

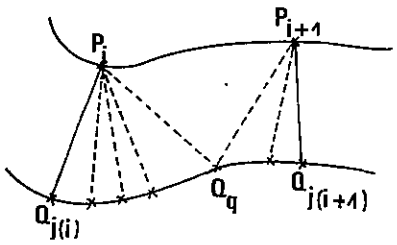


Figure 1

3.2. Link from a single contour to multiple contours.

Let C_{k-1} be a single contour at level z_{k-1} and C_{k1}, C_{k2}, ... C_{kr} be the set of r contours at level z_k . The idea is to generate automatically from these multiple contours a single closed contour C'_k. It will be put at the intermediary level $(z_{k-1}+z_k)/2$ and linked both to C_{k-1} and C_k.

To create C'_k the following procedure is developed :
1- Find the centroids B_i for i=1,r of the sub-contours C_{ki}
2- Generate a polygon P(r) passing through each vertice $B_1,B_2 \ldots B_r$.
3- Obtain the new contour by branching the contours $C_{k1},C_{k2} \ldots C_{kr}$ with P(r).

Steps 2 and 3 are described below.

Step 2: The problem is to link N points by straight-line segments avoiding crossing edges. Clearly there is not a unique solution. We propose two possible procedures to generate such a polygon.
The first procedure is recursive. Suppose that at iteration l, P(l) is a satisfying polygon (i.e. without crossing edges) of vertices $B_1,B_2 \ldots B_l$. Then P(l+1) is created by adding the new point B_{l+1} (Figure 2).It is linked

to the closest point B_c of P(1) if that does not introduce crossing edges.Else the closest point of P(1)- $\{B_c\}$ is tried, and the process is repeated. The solution obtained by this method is highly dependant on the order in which the vertices are numbered.
To avoid too large differences between the polygon angles, we propose a second procedure.It consists in finding which is the "most similar" to the convex envelop of the vertices $B_1 \ldots B_r$ (Figure 3).

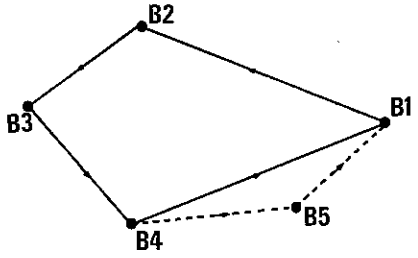


Figure 2
Generation of the polygon P(5) from the P(4) by adding the new vertice B_5.

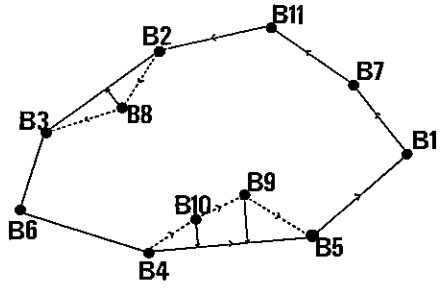


Figure 3
The continuous line represents the convex envelop, the dashed line show how the remaining points are included.

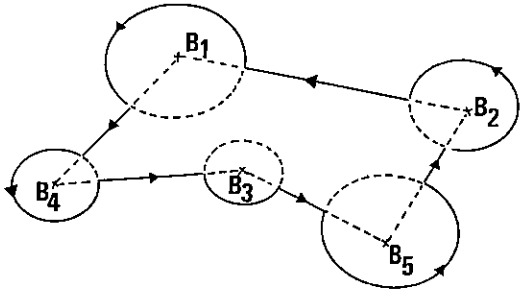


Figure 4
Generation of a single contour C'_k (continuous line) from the subset of contours $C_{ki}(i=1,5)$.

For that, the convex envelop of the r points is first computed using Graham method [11]. The unused vertices which are necessary inside of the convex envelop are linked to the vertices of the closest edge.

Step 3: The polygon P(r) is then used to create a single contour C'_k from the sub-contours $C_{k1} \ldots C_{kr}$. It is obtained using the intersection points of the polygon with the sub-contours as shown on Figure 4.

The intermediary contour is then linked both to C_{k-1} and C_k.

C'_k is set at level $z'_k=(z_{k-1}+z_k)/2$ and linked to C_{k-1} using the procedure described in section 3.1. Then level z'_k is linked to level z_k by connecting each sub-contour C_{ki} to the same sub-contour displaced to level z'_k. At last horizontal triangular patches are created by triangulation of the polygon P(r) in order to close the surface.

3.3. Link between multiple contours.

The problem of linking p contours at level z_k to q contours at level z_{k-1} can be treated as one of the problems previously seen (sections 3.1 and 3.2) if it is known which contours of the (k-1)th slice are to be linked to which contours of the kth slice.In some situations the problem can be solved automatically. A possibility is to group the contours according to the position of their centroids. This solution may fail if the contours are too distorted. Another possibility is to estimate the overlapping of the contours of the two sections and group them according to this criterion.
However most of the time, the situation is ambiguous and if no other informations are included, the automatisation can lead to wrong choices. In this case the program can be guided by user interaction to decide of the grouping of the contours.

4. RESULTS

The algorithm described above has been applied to simulated surfaces. The computation is performed on a PDP 11/34 . The triangulated surface is displayed with hidden-surface removal using a depth-buffer algorithm. It can be observed under different angles of view and will be represented on a plotter.
As an illustration we have simulated an object composed of 4 slices.The first slice contains only one point,the second slice contains a single contour,the third slice is composed of five circular sub-contours and the last one is a set of five points. They are displayed on Figure 5. To link the 2nd slice to the 3rd one , the algorithm generates automatically the additional contour shown on Figure 6. The resulting surface is represented on Figure 7 a) and b) under two different angles of view.

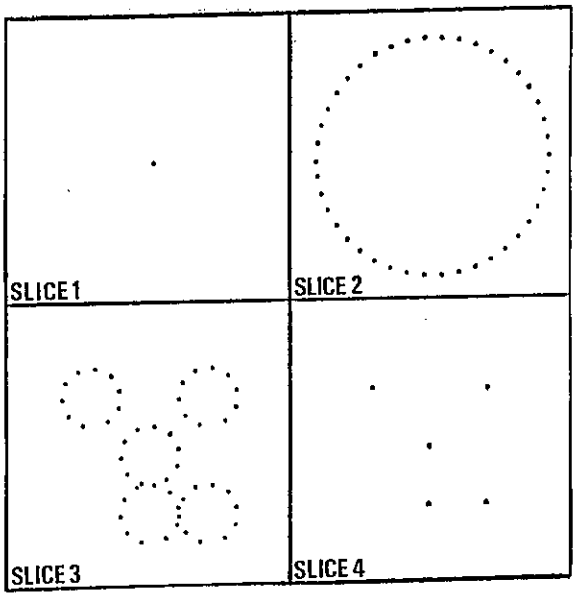


Figure 5
The four slices of the simulated object.

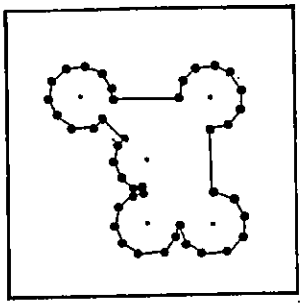
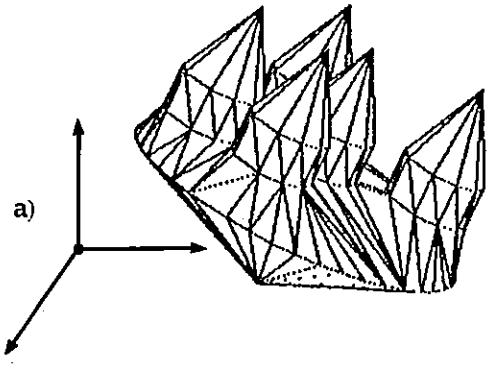


Figure 6
The additional contour generated between slices 2 and 3 .



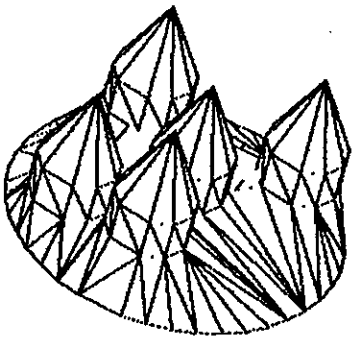


Figure 7
Display of the surface under two angles of view:
a)rotation of 30° around OY and OZ
b)rotation of 30° around OX, OY and OZ

5.CONCLUSION

In this paper we have described a tiangulation algorithm with special care to the problem of branching multiple contours. More than two contours can be automatically linked to a single one by using an interpolated line. This produces a good visual representation when using hidden-surface removal and shading. The processing overhead is low and for complex objects a relatively small number of additinal faces are generated. Nevertherless due to the lack of human supervision the algorithm can generate unexistant structures in the case of scattered multiple contours.In further works this algorithm will be evaluated on true biomedical data .

REFERENCES

[1] Robb, R.A., Hoffman, E.A., Sinak,L.J. , Harris,L.D. and Ritman,E.L., Proc. of the IEEE, vol 71,n°3,1983,308-319.
[2] O' Rourke, J.,Proc. of the 1981 Int. joint Conf. on Artificial Intelligence,664-666.
[3] Hermeline, F..,R.A.I.R.O. Numerical Analysis,vol 16,n°3,1982,212-242
[4] Boissanat, J.D., Proc.IEEE ICASSP 1984,745-748
[5] Keppel, E., IBM Jour.Res. Dev.,1975,2-11.
[6] Fuchs H. , Kedem, Z.M. and Useltom, S.P.,Communication of the ACM, vol 20,n°10,1977,693-702.
[7] Christiansen, H.N. and Sederberg, T.W., Computer Graphics,vol 12,1978,187.
[8] Cook, P.N. , Batnitzky, S. , Lee, K.R. , Cook, L.T., Fritz, S.L., Dwyer,S.J.III and Charlson, E.J., Proc. 14th Hawaii Int. Conf. on System Science, vol 2,1981,358-
[9] Cook, P.N., Automedica ,vol 4,1981,3-12.
[10] Ganapathy, S. and Dennehy, T.G., Computer Graphics, vol 16,n°3,1982,69-75.
[11] Graham , Information Processing Letters 1,1972,132-133.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

LINEAR PREDICTION IN DIRECTIONAL IMAGES

M. Benard and M. Kunt

Signal Processing Laboratory, Swiss Federal Institute of Technology
16, Chemin de Bellerive, CH 1007 Lausanne, Switzerland.

Abstract: The directional decomposition of images has been introduced as an efficient approach for image coding [1], [2]. Recently its improvements lead to very high compression ratios (up to 120:1) [3], [4]. In this paper a new step is taken with the use of linear prediction in directional images. Both theoretical and implementation point of views are considered. The goal is to further increase the compression ratio.

1. INTRODUCTION.

Image coding with compression ratios around 100:1 was recently achieved with the directional decomposition based technique [4]. In this second generation method, the image is divided into a low-frequency and a set of directional components [1]-[3]. The low-frequency component is coded using a classical strategy, whereas edges are detected and coded in each directional component seperately [4]. A new step can be made to decrease the redundancy of information by relating the directional images or edges to a prediction model. The goal of this approach is to code the prediction coefficients and errors with less bits than the original information, which implies a good choice of the prediction structure.

In this paper the general model for 2-D linear prediction applied to images is first presented in section 2. The computational problems related to the solution of large linear systems for prediction error minimization are also examined. A conclusion of this study is that a synthetic model of the information in the directional image must be built to perform the prediction to avoid tremendous computation. Since in directional images the main information is concentrated in edges, 2-D linear prediction using a synthetic description of edges is chosen. This description and the prediction model are presented in section 3. A vector is associated to each edge element, including its position and local profile parameters (magnitude and width). Then, the prediction model operates on these vectors and enables to estimate edge position and parameters within a prediction structure defined on a set of connected edges. Results on synthetic and real images are shown in section 4. The advantage of the technique described is an improvement of the compression ratio for images having connected edges. In the conclusion, future trends and applications of linear prediction in directional images are discussed.

2. GENERAL MODEL FOR 2-D LINEAR PREDICTION.

Let us consider a sequence of N images and let x(n), n=0..N-1, be a vector representing K elements of the n-th image, which may be pixel grey- level values or any other parameters extracted from this image. The 2-D linear prediction consists in defining the vector xp(n) as follows:

$$xp(n) = \sum_{i=1}^{M} a(i).x(n-i) \qquad (1)$$

where M is the order of the prediction and a(i) KxK matrices. The predicted value of x(n) is xp(n) and the associated prediction error is:

$$e(n) = x(n) - xp(n) \qquad (2)$$

The total prediction error E is obtained by summing the squared errors e(n) between two boundaries n0 and n1 (0[=n0[=n1[=N). This error must be minimized with respect to the coefficients a(i). This leads to the following linear system [6]:

$$\sum_{i=1}^{M} a(i).c(i,j) = -c(0,j) \quad \text{for } j=1,2,\ldots,M \qquad (3)$$

$$\text{where } c(i,j) = \sum_{n=n0}^{n1} x(n-i)^T.x(n-j)$$

Since each a(i) is a KxK square matrix, the above system to be solved for a prediction of order M has M.K.K equations for M.K.K unknowns. In the context of image processing, this leads to severe restrictions on K, that is on the number of elements included in x(n). Tremendous computational effort need to be developed for large values of K. Indeed, it is quite unefficient to use for x(n) a set of pixel

grey-level values (1024 prediction coefficients to compute for a 32x32 region). A more convenient and more interesting approach is to put in x(n) a set of parameters representing a local context (edge, region, texture, etc.), whose variations will be predicted using (1). Moreover, in the case of image coding the number of bits required to transmit the first values x(0)..x(M-1), the coefficients a(1)..a(M) and the prediction errors e(M)..e(N-1) must be smaller than the number of bits required to transmit x(0)..x(N-1). This last condition requires that x(n) be a slowly varying structure with respect to n.
Our goal is now to choose x(n) and perform the prediction with the directional images.

3. DESCRIPTION OF THE PREDICTION MODEL FOR DIRECTIONAL IMAGES.

In this section, the prediction structures, the content of the vector x(n) and the implementation of the model will be presented, after an introductory motivation. For clarity, a synthetic example based on the image shown in fig. 1a will be used.

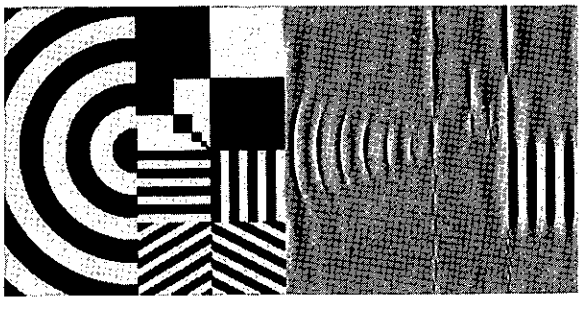
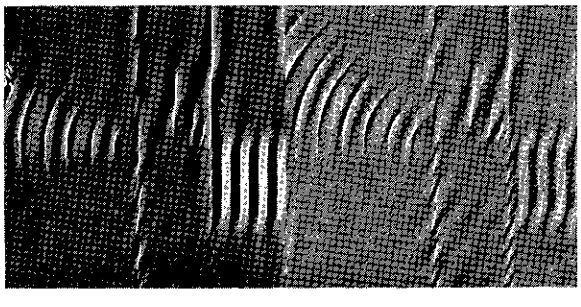


Fig. 1: a- Original image, b- first directional component,



c- second directional component, d- third directional component.

Let us first consider the first three directional components of this image as shown in fig. 1b, 1c and 1d for a decomposition into 17 components. They are made by one low-frequency and 16 directional images. In these images, the principal information is concentrated in edges. This fact is of primary importance in the directional component coding strategy [4], in which edges are detected, modeled and coded. For further improvement of the compression ratio, the correlation existing between edges in neighbouring components suggests us to perform a prediction model which can estimate an edge in one directional image given its spatial neighbours in the other directional images.

For example, this situation is shown in Fig. 2, which displays the first four directional images after edge detection. Edges corresponding to the first directional component (direction number 1) are shown in Fig. 2a. These edges are detected as being above a given threshold in the directional image. Edges in the neighbouring directions (direction number 16 and direction number 2) which are also above the same threshold are detected as strong edge in their direction but become weak edges in the central direction. In fig. 2b, the strong edges are detected in component number 2 while the weak ones are detected in components number 1 and 3. The same convention applies for fig. 2c and 2d. Starting from a strong edge of fig. 2a which has a neighbouring weak edge and following the corresponding curve in fig. 2b, 2c and 2d, it is interesting to see that along this curve the strong edge of fig. 2c can be predicted from the strong edges of fig. 2a and 2b using a second order prediction model.

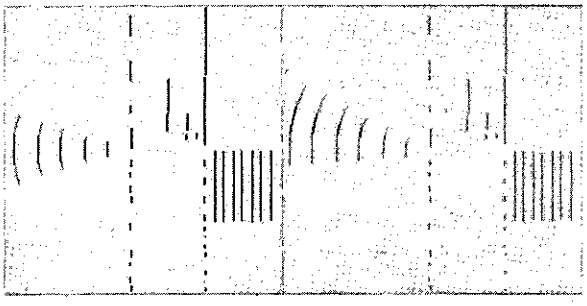
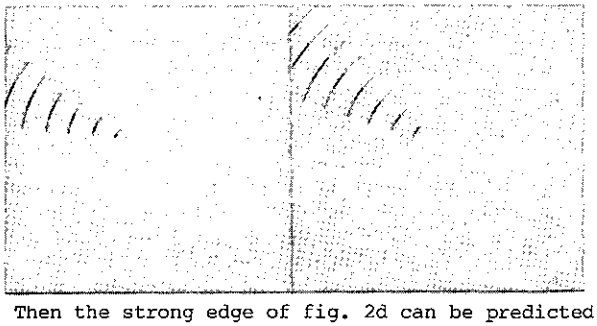


Fig. 2: Strong and weak edges in the first four directional images.



Then the strong edge of fig. 2d can be predicted from the strong edge of fig. 2b and the predicted edge in fig. 2c. Since this property is limited to particular regular curves (circles, spirals...), a necessary extension consists in defining a more general prediction structure.

To define the prediction structure, let us therefore consider the edge image of fig. 3a which is simply the union of the edge images computed for each directional component. In this image it is possible to follow curves formed by adjascent segments issued from different directional images. Such curves will be called prediction structures . One of them is shown in fig. 3b.

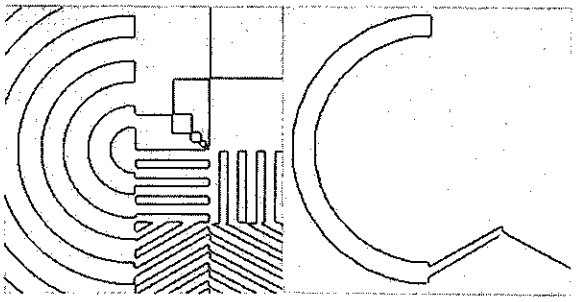


Fig. 3:

a- Edge image, b- a prediction structure.

A prediction structure is thus a set P of N neighbouring edges E(n), n=0.. N-1, each edge E(n) being detected from a particular directional image and having a direction denoted by D(n). The prediction model is then defined over these structures. Let x(n) be the vector characterizing the edge E(n). Equation (1) predicts xp(n) given x(n-1)..x(n-M) where M is the order of the prediction. The parameters of the prediction model are obtained after solving the system given by equation (3). To avoid large systems, x(n) must include only a small number of parameters as indicated previously. Let us now define x(n) more precisely within this context.

The synthetic description of the edge E(n) by x(n) is largely based on the description made for directional edge coding [4], [5]. The vector x(n) is composed of four parameters. The two first parameters are its length L(n) and its direction D(n) as shown in fig. 4a. With the coordinates of the starting point S(n), they allow to reconstruct exactly the position of the edge E(n). But since the coordinates of S(n) are equal to these of the ending point of E(n-1), it is not necessary to include them in x(n). The two other parameters are related with the description of the edge grey-level profile by a synthetic wavelet [5]. They are respectively the width W(n) and the magnitude I(n) of the wavelet characterizing the profile as shown in fig. 4b. These four parameters are real numbers, which shall be quantized for coding purposes. Let us now present how the model is implemented with this description of edges and with the prediction structures defined in the previous paragraph.

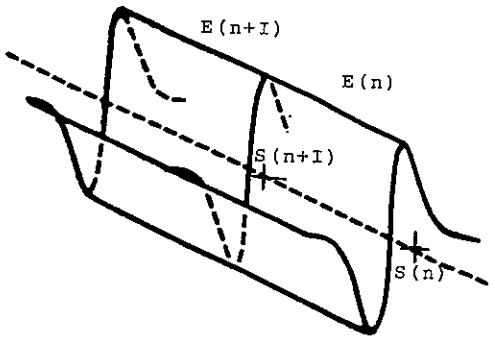


Fig. 4: Synthetic representation of edges: (a) Position- (b) Profile.

In the implementation of the Mth order prediction model on a structure of N edges, two data sets are involved. The first includes the coordinates of the startins point S(0) of the first edge E(0), the vectors x(0)..x(M-1) needed to begin the prediction and the prediction matrix a(1)..a(M). The second contains the sequence of the prediction errors e(M)..e(N-1) . Using equations (1) and (2), it is easy to reconstruct exactly the N edges of the prediction structure from these two data sets. However, it is not possible to consider one prediction model per structure for two reasons. First, some structures may have a number N of edges smaller than the order M of the prediction. Second, it is a waste of bits in a coding context to have the first data set for each structure, even if the prediction errors are then the smallest. Therefore, the implementation of the same model extends to several structures by adding in the second data set the coordinates of the starting point of the first edge for every structure change. This leads to the following implementation scheme:

First data set :
S(0), x(0)..x(M-1), a(1)..a(M) (4)
Second data set:
e(M)..e(M+N1)
end of first structure
S(M+N1+1), e(M+N1+1)..e(M+N1+N2)
end of second structure
etc. (5)

This scheme is available for any number of prediction structure and of edges per structure. Another point to consider is the order M of the prediction model. Large orders involve several K.K matrices a(i) to code. A compromise is to be found between maximum efficiency obtained for large values and coding requirements for small values of M. Finally the coefficients of the matrices a(i) have to be computed from (3) using an edge sequence representing a typical curve of the considered image, such as a part of circle in our example of fig. 3. Let us now present for this example and for a real image the implementation and its associated results.

4. RESULTS.

Let us first consider the synthetic example of fig. 1a. The total number of edges found in fig. 3a is 266. Among them, 21 are isolated and cannot be included in a prediction structure. The 245 remaining edges are contained in 19 prediction structures. For image coding applications, two results must be considered. The first is the compression ratio obtained after coding the two data sets defined in previous section. In our example, a prediction model of order M=2 is used and the following bit distributions is obtained after Huffman optimal coding strategy:

12 bits per position S.
10 bits for the two first vectors x(0) and x(1).
8 bits per coefficient of a(1) and a(2).
8 bits per error vector e(i).

This estimation leads to a compression ratio of 150:1 (0.006 bit per pel), since about 2900 bits are needed to code the edges and 511 the low-frequency component. Note that the introduction of the prediction structures enables us to spare many bits devoted to edge position and profile coding in the method presented in [4]. The second result to show is the reconstruction associated to the coded data. In fig. 5 the original (a) and reconstructed (b) high-frequency components are shown while fig.6 displays the original and decoded images.

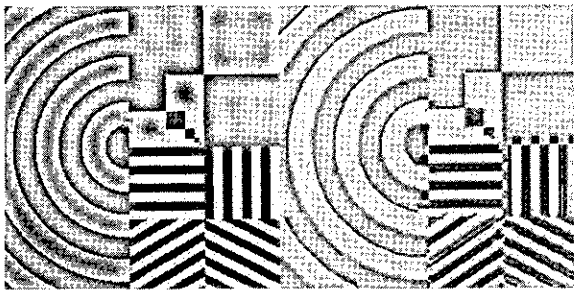


Fig. 5: a b

Fig. 6:

Original image- Decoded image (0.006 bits/pel)

The second example is performed on the real image shown in fig. 7a. A compression ratio of 110 to 1 is estimated. The decoded image is shown on fig. 7b.

Fig. 7: a b

5. CONCLUSION.

The prediction model presented in this paper is an efficient technique for increasing the compression ratio (up to 150 to 1) obtained with the directional decomposition based image coding method. Moreover, the prediction structuted results extend thus from image coding to scene analysis and object representation.

REFERENCES.

[1] M. Kunt, A. Ikonomopoulos and M. Kocher, "Second-generation image coding techniques" (invited paper), Proceedings of the IEEE Vol.73, No. 4, April 1985, pp. 549-575.

[2] A. Ikonomopuolos and M. Kunt, "High compression Image Coding via Directional Filtering", Signal Processing Volume 8, No. 2, April 1985,pp. 179- 205.

[3] M. Benard and M. Kunt, "Directional decomposition image transformation", IASTED International Symposium on Applied Signal Processing and Digital Filtering, 19-21 June 1985, Paris-France.

[4] M. Benard and M. Kunt, "Improvements of directional decomposition based image coding", SPIE International Image Processing Symposium, Dec. 2-6 1985, Cannes-France (to be published).

[5] R. Cusani, "New results in directional decomposition based image coding", SPIE International Image Processing Symposium, Dec. 2-6, 1985, Cannes-France (to be published).

[6] J. D. Markel and A. H. Gray, "Linear prediction of speech", Communications and Cybernetics 12, 1976, Springer Verlag-Berlin.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

K3.9

MULTICRITERION IMAGE RECONSTRUCTION AND IMPLEMENTATION

Wang Yuan Mei
Department of Scientific Instrument Engineering,
Zhejiang University, P.R.China

Lü Wei Xue
Professor, Vice President of Zhejiang University,
Hangzhou, P.R.China

ABSTRACT In this paper, a multiobjective optimization method of the maximum entropy image reconstruction from projections was described. We applied a new iterative algorithm to solve this problem. Computer simulation results are given.

1. INTRODUCTION

The problem of reconstructing image from projections has arisen in a large number of scientific and engineering fields.

G.T. Herman et al [1 - 3] have majored in a comprehensive study of quadratic optimization methods for image reconstruction and obtained many results.

The maximum entropy image reconstruction was formulated as the solution of constrained optimization problem [4 - 5]. Auther [6 - 8] has suggested that maximum entropy image reconstruction from projections was represented as the solution of nonlinear goal programming problem.

In this paper, we discussed a new multiobjective model and iterated algorithm for image reconstruction in detail. Furthermore, the related property of goal optimization was described.

2. NEW MODEL AND ALGORITHM

In the previous works [4 - 6], the maximum entropy image reconstruction from projections was formulated as the solution of constrained optimization problem:

$$\underset{f(x,y)}{\text{maximize}}\ H(f) = \iint_{\mathcal{D}} f(x,y)\ \ln\left(f(x,y)\right)^{-1} dxdy \qquad (1)$$

over the set of continuous functions $f(x,y)$, subjet to the linear constrains:

$$g(r,\theta) = \iint_{\mathcal{D}} f(x,y)\delta(x\cos\theta+y\sin\theta-r)dxdy = \int_L f(x,y)ds \qquad (2)$$

and $f(x,y) \geqslant 0$, for $f(x,y) \in D$. Where the

Line $L = \{(x,y): x\cos\theta+y\sin\theta=r\}$

$(r,\theta) \in D \triangleq \{(r,\theta): -\infty<r<\infty, 0\leq\theta\leq\pi\}$

$\delta(\cdot)$ is the Dirac delta function.

2.1. MULTIOBJECTIVE OPTIMIZATION MODEL FOR IMAGE RECONSTRUCTION

In the multiobjective optimization model of the maximum entropy reconstruction, the result problem formulation is then given by discrete version:

Find $f = (f_1, f_2, \ldots, f_{m^2})^T$ so as to minimize

$$a = [(p_1+n_1), (p_{21}+n_{21}), \ldots, (p_{2m^2}+n_{2m^2}), (p_3+n_3) \qquad (3)$$

such that

$$f^T \ln f + n_1 - p_1 = -\ln(m^2)$$

$$g_i - \sum_{j=1}^{m^2} q_{ij} f_j + n_{2i} - p_{2i} = 0, \quad i = 1, \ldots, m^2 \quad (4)$$

$$\|g - Qf\|^2 + n_3 - p_3 = 0 \quad (5)$$

and f, $n = (n_1, n_{21}, \ldots, n_{2m^2}, n_3)$,

$$p = (p_1, p_{21}, \ldots, p_{2m^2}, p_3)^T \geq 0 \quad (6)$$

Where g, f, Q is the known projections vector, image vector, and projection matrix, respectively.

2.2. THE SOLUTION SET

The general form of image reconstruction represented as the multiobjective programming, is:

Find $f = (f_1, f_2, \ldots, f_N)$ so as to minimize

$$a = \{a_1(n,p), \ldots, a_L(n,p)\} \quad (7)$$

such that:

$$h_i(f) + n_i - p_i = b_i, \quad i = 1, \ldots, k \quad (8)$$

and:

$$f, n, p \geq 0 \quad (9)$$

Where equation (7) is our achievement function and equations (8) are the problem objectives. A decision variable, denoted as f_j (with $j=1,2, \ldots, N$) will be assumed nonnegative unless otherwise noted. A deviation variable reflects the underachievement (negative deviation and denoted as n_i, $i=1,2,\ldots,k$) or overachievement (positive deviation and denoted as p_i, $i=1,2, \ldots, k$) of an objective i. All deviation variables are assumed to be nonnegation unless otherwise specified.

Prior to presenting our discussion of the new iterative algorithm for solving the nonlinear multiobjective programming, we shall first describe some of the definitions and concepts.

Definition 1: Feasible Solution. Any set of nonnegative f_j, n_i, p_i values constitute a feasible solution.

Definition 2: Implementable Solution. An implementable solution is a feasible solution in which all absolute objectives are satisfied.

Definition 3: Achievement Function (or vector). The goal programming achievement function (a) is an ordered vector of a dimension equal to the number (k) of preemptive priorities whithin the problem and expresses the level of achievement of each objective set whithin a given priority.

Definition 4: Optimal Solution. The solution (f) to a given goal programming model is considered optimal if, for this solution (termed f*), the corresponding value of a (termed a*) is the same or preferred to the value of a for any other feasible solution.

2.3. ALGORITHM

We have employed a set of New it-erative algorithm for solving image reconstruction problem with highly satisfactory results. Our iterative algorithm is one of a class of accelerated search methods. Such algorithm increase their search step size if previous searches have been successful and maintain or decrease the step size otherwise.

We may specify the steps of our iterative algorithm for image reconstruction as follows:

New Iterative Algorithm of Image Reconstruction

Notation: f image vector;
g projection vector;
Q projection matrix;
f_r reconstructed image;
$N=m^2$ number of projections.

(1) Begin set $k=0$. $l=0$, $\gamma=\lambda\theta$, $\lambda \in [0,1]$, θ: the primary set of increments;

IN_{max} :

the miximum iterative number; θ_{min} : the minimum of θ ; δ : the minimum resolution of achievement function a.

$f^{(1)} = f^{(0)}$ = the arithmetic mean of all the rows of the matrix Q.

(2) Obtain the achievement function value $a(f^{(1)})$ for $f^{(1)}$. Set $\xi_{1,0} = f^{(1)}$, j=k+1, k=k+1, l=l+1.

(3) Examine the changes concerning $\xi_{j,0}$ so as to determine $\xi_{j,i}$ as follows

(a) set i=1

(b) If $a(\xi_{j,0}+\gamma_i) < a(\xi_{j,0})$ then

$\xi_{j,i} = \xi_{j,0} + \gamma_i$ and go to 3 (d)

else go to 3 (c),

(c) If $a(\xi_{j,0}-\gamma_i) < a(\xi_{j,0})$ then

$\xi_{j,i} = \xi_{j,0} - \gamma_i$ and go to 3 (d)

else = and go to 3 (d)

(d) If $l=m^2$, go to 4 else i=i+1 and go to 3 (b).

(4) If $a(\xi_{j,m^2}) + \delta < a(f^{(j)})$ then

$f^{(p+1)} = \xi_j$, m^2 to 5

else $f^{(p+1)} = f^{(j)}$ and go to 6

(5) Set j=p+1, if $l > IN_{max}$ then go to 8

else $\xi_{j,0} = (2^{\lambda})f^{(j)} - f^{(j-1)}$ $\lambda \in [0,1]$ and go to 2.

(6) Set $\gamma = 2^{-\lambda}\gamma$, $\lambda \in [0,1]$.

(7) If $\gamma < \theta_{min}$ then go to 8 else j=p+1 and $\xi_{j,0} = f^{(p+1)}$ go to 2.

(8) End of algorithm, to select the last base point as reconstructed results.

3. Computer Simulation Results

The effectiveness of new model and algorithm was shown by numeral examples of reconstruction for Shepp-Logan head data. A comparison to single objective optimization methods were given. The advantages of our modeling and algorithm for image reconstruction are manifesting themselves in combination of other optimization methods.

The measures of image quality defined below in the reconstruction literature (3):

$$\mathcal{E} = \left\{ \sum_{i=1}^{N}\sum_{j=1}^{N} \left(f_0(i,j) - f_r(i,j)\right)^2 \Big/ \sum_{i=1}^{N}\sum_{j=1}^{N} \left(f_0(i,j) - \bar{f}_0\right)^2 \right\}^{1/2}$$

Where $f_0(i,j)$ and $f_r(i,j)$ denote the gray levels in the ith pixel of the jth row of the digital test phaton and the reconstruction, respectively, and $\bar{f}_0$ denotes the average gray level in the digital test picture. The simulation results are reported in Table 1. For each algorith, result of the 6th iteration is reported.

TABLE 1

ALGORITHMS	$\mathcal{E}$
Kashyap and Mittal [3]	0.2445 (7)
Least Squares [3]	0.2462 (7)
Summation [3]	0.2738 (6)
Improved [3]	0.2268 (8)
Our GP Model and Algorithm	0.2012 (6)

4. CONCLUSIONS

The multiobjective programming method for image reconstruction has many advantages over single quadratic optimization in terms of storage, algorithm simplicity and convergence rate. We add an entropy constraint to the classical image reconstruction in quadratic optimization model in order to preserve a desired level of accessibility in the solutions to the image reconstruction. We add an entropy of the image to the objective function with the purpose of introducing an element of smoothing into the solution of nonlinear programms, the element of the optimal solutions become strictly positive. The entropy constraint may in such cases be looked upon as a substitute for lost complexity.

Our modeling and algorithm may be applied to the sonar and SAR multidimensional sig-

nal reconstruction, and multidmensional power spectrum estimations.

REFERENCES

1 G.T. Herman and A. Lent, Quadratic optimization for image reconstruction, I, Computer Graphics Image Processing 5 (1976), 319-332.

2 E. Artzy and G.T. Herman, Investigation of quadratic optimization techniques for image reconstruction, in Proceedings of the 1977 IEEE Conference on Decision and Control, New Orleans, Louisiana, Dec. 1977, pp.350-360.

3 E. Artzy, T. Eliving, and G.T. Herman, Quadratic optimization for image reconstruction, II, Computer Graphics & Image Processing, Vol. 11, (1979), 242-261.

4 S.J. Wernecke, et al., Maximum entropy image reconstruction, IEEE Trans. Compt., Vol. C-26, (1977), 351-364.

5 G. Minerbo, MENT: A maximum entropy algorithm for reconstructing a source from projection data, Computer Graphics & Image processing, Vol. 10, (1979), 48-68.

6 Wang Yuan Mei, Goal Programming method of maximum entropy image reconstruction from projections, presented at International Conference on Numerical Optimization and Applications, Xian, shaanxi, China, June, 1986.

7 Wang Yuan Mei, Nonlinear goal programming model and algorithm of maximum entropy image reconstruction from projections, presented at the International Conference on Acoustics, Speech, and Signal Processing, Beijing, Institute of Acoustics, Academia sinica, 1986.

8 Wang Yuan Mei, New method of maximum entropy image reconstruction, presented at the International Conference on Acoustic, Speech, and Signal Processing, Tokyo, Japan, April, 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

VIDEOPHONE CODING USING BACKGROUND PREDICTION

Sergio Brofferio

Dipartimento di Elettronica Politecnico di Milano
Piazza L. da Vinci 32 MILANO (Italy)

Vittorio Corradi

Laboratori Telettra S.p.A.
Via Trento 30 VIMERCATE (Italy)

Scene background information can be exploited to improve videophone and videoconference coding. The present research investigates its potential capabilities and limits using elementary knowledge based concepts. Basic problems as scene classification and background updating are addressed so that a predictive coding algorithm is proposed.

1. INTRODUCTION

Videoconference and videophone systems allow the visual communication of well defined life scenes consisting of people acting in front of an approximately fixed background. The digital coding of these source signals has to exploit the algorithmic and technological performances in order to achieve a subjective satisfactory quality within the given channel capacity. VLSI availability allows large amount of storage and parallel processing which have to be exploited in order to achieve the desired goals.
Present video compression algorithms use structural and statistical properties of the source signal; specifically motion compensation increases the temporal correlation of the image zones belonging to moving objects while intraframe and interframe correlation are used for predictive or transform coding.
A substantial increase in transmission capacity can be foreseen if means are found to avoid retransmission of information already available at the receiver end. This requires not only additional memory space but especially suitable algorithms for the processing of the additional information.
In this paper we analyse how the background information of videophone and videoconference signals can be utilized.
In the next section background information will be defined and discussed, in section three its potential capabilities are presented. In section four the updating problem of background information is discussed together with a proposed solution in the simplest scene model. Section five presents a possible predictive coding algorithm using conditional replenishment with motion compensation and background prediction.

2. A SCENE MODEL

A videophone image is the projection of the optical energy reflected by human beings moving in front of a fixed background; a simple model is obtained if we associate to each pel P of the n-th image I(P,n) a semantic meaning: Object (O) if it is the projection of a moving human body or Background (B) when it corresponds to the room walls.
Not every pel of the image can be assigned to the semantic states O or B as for instance when a new object is entering the scene or it is suddenly changing shape or at last when it is appearing for the first time during a communication session. A third state is therefore necessary as fuzzy state and it is named Undefined (U).
We can therefore associate to each image I(P,n) its state IS(P,n) whose elements belong to the set (O,B,U).
The Undefined state is detailed into two substates: New Scene (NS) and New Background (NB) corresponding respectively to scene elements appearing on the scene and to unknown background uncovered by a moving object.
Object motion can be very useful for image state identification and will be

used whenever it can avoid undetermined situations.
The background information is represented by its luminance matrix b(P,n) and its state BS(P,n) whose elements assume value 0 if at pel P the background luminace is unknown and, value 1 in the opposite case. When the background is known and the objects translate the image state assumes only the values 0 or B, this specific and simpler case will first be used in the proposed analysis.

The two basic problems concerning the the proposed model are: image segmentation into its states and background updating.
Necessary, but not sufficient, conditions for image state assignment can be derived from the following luminance differences:
- spacial or intrafield difference:

$$SD = l(P,n) - l(P-v,n)$$

where v is a unity vector;

- frame or temporal difference:

$$FD = l(P,n) - l(P,n-1);$$

- displaced frame difference:

$$DFD = l(P,n) - l(P-d,n-1)$$

where d is the displacement vector of the object;

- background difference:

$$BD = l(P,n) - b(P,n)$$

Table I shows the correspondence between image state element and the smallest luminance difference when no clustering is applied to the segmentation.

State Element	Significative Difference
0	DFD
B	BD
NB	FD
NS	SD

Table I

In order to simplify image segmentation and according to most common structural situations we assume that each image partition does not surround other types of partitions except the case of scene background.

Background updating consists in generating the current background b(P,n) from (n-1)th frame information and present image; this requires its segmentation into semantic elements.

3. BACKGROUND POTENTIAL CAPABILITIES

The capabilities of background information for reducing the bit rate of interframe predictive coding have already been analysed by Mukawa and Kuroda (1) who obtained a relationship between the significative pel bit rate and the interframe object displacement using an exponential correlation model. An extension of these results to motion compensated predictive coding using background information is not possible unless a model for motion compensation is introduced; in fact assuming exact displacement measurement and background prediction the bit rate could be reduced to zero!
Here we use a much simpler model to evaluate the potential gain and have an insight into the critical coding aspects. Let us model the moving object with a circle of radius r measured in pels, moving at a speed of d pels/frame; moreover assume that nc and nb are respectively the efficiency of motion compensation and background prediction, where efficiency defines the number of pels which can be correctly compensated and predicted. We have:

$$Nc = (1 - nc) * \pi * r^2 + 2r * d \quad \text{and}$$

$$Nb = (1 - nc) * \pi * r^2 + (1 - nb) * 2r * d$$

where Nc and Nb are the image pels which require transmission with motion compensation and with additional background prediction.

The gain of motion compensation with background prediction with respect to predictive coding using only motion compensation is:

$$A = Nc/Nb =$$

$$= 1 + nb * (d/r) / (\pi * (1 - nc)/2 + (1 - nb) * (d/r))$$

For instance when nb=nc=.8 we have A=2 for d/r=.5.
This very simple model and expression confirm the intuitive idea that background information can efficiently be exploited when the displacement is large and the motion is correctly compensated.

The preceeding considerations have shown the capabilities and the limits of background information in reducing the effective bit rate, in the follo- we will examine how background infor- mation can be used for improving the coding algorithm, specifically displa- cement measurement and temporal inter- polation so that a large benefit of the background information can be fore- seen.

We now examine how block and recursive displacement measurement techniques are improved by background knowledge. We assume that current image state has been computed using the segmentation algorithm reviewed in the Appendix, in order to simplify this preliminary research IS(P,n) consists only of O and B elements.
Let us first consider block displace- ment; it is measured by the position of highest correlation within a dis- placement domain of the preceeding frame surrounding the block position in the current frame. Correlation mea- surement is improved if we discard all the pels which do not belong to the moving object; this is possible with the image state IS(P,n) in addition to the segmentation of the present ima- ge with respect to the previous one. Background information allows use of larger blocks as required by displace- ment vector transmission bit rate and by transform coding efficiency.
Let us now examine recursive displace- ment measurement; it requires that the previous pel in the recursion belongs to the moving object in order to have a significative measurement (2); in this case also image state allows a validation of the pels used in the re- cursion.
In the case of interpolative coding the subsampled images and backgrounds sequence are used to interpolate the skipped images, background information can be used whenever motion compensa- ted interpolation is not satisfactory.

4. BACKGROUND UPDATING TECHNIQUES

Background information is stored in a frame memory whose content has to be updated when new background informa- tion appears or its luminance changes. Updating is the most critical opera- tion for an efficient use of back- ground knowledge; a technique has proposed in (1): it is characterized by a revision time interval which slows down updating and does not avoid that the object luminance be used to update the background information.

The strategy investigated in this research is based on image segmenta- tions corresponding to its scene con- tent. We assume that l(P,n) is known together with l(P,n-1), b(P,n-1) and IS(P,n-1) while we are interested in obtaining b(P,n) so that the current image state can be obtained.
Background updating and current image state computation are conceptually a unique operation, in this research we have first updated the background and the segmented the image. Moreover the techniques strongly depend on the New Scene content of the current ima- ge.
The technique used here performs two segmentations into Changed (C) and Non-Changed (NC) areas (see Appendix) of l(P,n) with respect to l(P,n-1) and b(P,n-1) which are defined SLL(P,n) and SLB(P,n) . If the map- ping of IS(P,n-1) into SLB(P,n) using motion compensation is satisfactory, then the following updating table can be used:

SLL(P,n)	SLB(P,n)	IS(P,n-1)	b(P,n)
C	x	x	b(P,n-1)
NC	NC	B	b(P,n-1)
NC	NC	O	b(P,n-1)
NC	C	B	l(P,n-1)
NC	C	O	l(P,n-1)

Background updating, b(P,n)=l(P,n-1), occurs when: 1) background luminance is changing, 2) objects uncover new back- ground; Fig. 1 shows an example of ima- ge state (O,B,NB).

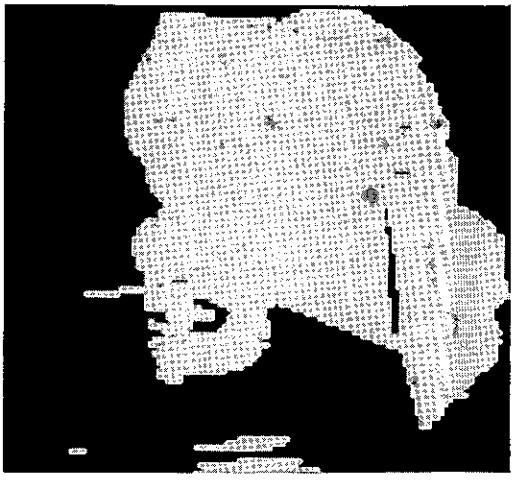


Fig.1

Image state
Black= Background; Gray= New Background
White= Object

5. CONDITIONAL REPLENISHMENT CODER USING BACKGROUND PREDICTION

Motion compensated conditional replenishment coder performances can be enhanced by an appropriate use of background information. Different solutions are possible, the one presented here is aimed at a compromise between minimization of the prediction errors and that required to transmit the different prediction modes.
Three predictors are available:
-intraframe for new scenes, the prediction error is trasmitted;
-interframe for object or new background, the prediction error is transmitted and it requires previously the displacement information computation, this can also be transmitted (block measurement) or predicted (recursive measurement);
-background luminance for the frame areas which can be classified as scene background, the prediction error is not transmitted.
The segmentation required for predictor selection is obtained minimizing the bit rate at constant quality and utilizes the algorithm proposed in (3). Background memory is updated at each transmitted frame using the technique proposed in the previous section.

6. CONCLUSIONS

The research has assessed that background information can be used for improving videocoding. Satisfactory motion measurement and scene content image segmentation are of paramount importance for background information updating. Scene analysis has therefore to become an integrated part of future videocoding algorithms so that further researches are needed in this direction.

APPENDIX

We shortly review a simplified version of a general segmentation algorithm based on dynamic programming optimization presented in (3).
Let l(P,n) be the image which has to be segmented into Changed (C) and Non-Changed (NC) areas with respect to a reference image r(P,n). The segmentation has to preserve the physical structure of scene and be insenstive to noise.

Segmentation is performed line by line locally optimizing the following cost function:

$$C(P,k) = \min_{h=C,NC} \left(C(P-1,h) + TC(h,k) + LDC(P,k) \right)$$

where k= C,NC; TC(h,k) is the state transition cost:

$$TC(h,k) = \begin{cases} 0, & \text{for } h=k \\ TP, & \text{for } h \neq k,\ TP = 2-8 \text{ depending on image noise} \end{cases}$$

and LDC(P,k) is the luminance difference cost:

$$LDC(P,k) = \begin{cases} l(P,n)-r(P,n) & \text{for } k=C \\ LP, & \text{for } k=NC,\ LP=2-8 \text{ depending on image luminance statistics} \end{cases}$$

Local optimization is used at the end of line scanning to obtain the global optimization.
TP and LP are determined experimentally for the best physical compromise.
The proposed optimization leaves some uncorrect segmentation areas which are suppressed by non linear spatial filtering and then by occlusion of the isolated areas.

References

(1) N. Mukawa, H. Kuroda: Uncovered Background Prediction in Intraframe Coding, IEEE Trans. on Com. Vol. COM-33,n.11 Nov.1985

(2) C. Cafforio, F. Rocca: The Differential Method for Image Motion Estimation, in Image Sequence Processing and Dynamic Scene Analysis T. S. Huang ed. Springer Verlag Berlin 1983

(3) S. Brofferio, F.Rocca: Interframe Redundancy Reduction of Video Signals Generated by Translating Objects, IEEE Trans. on Com. Vol. COM-25, n.4 Apr. 1977

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REDUNDANCY AND IRRELEVANCY REDUCTION OF TV-SIGNALS BY INTERFRAME DPCM-CODING

Ferdinand Arp

University of Wuppertal
Department of Electrical Engineering
Wuppertal, West Germany

The first part deals with the description of an interframe DPCM-coder which is implemented according to earlier considerations. The experimental redundancy reduction fits the theoretically derived values only if the predictor is switched by a movement detector. As so far the picture contents may be assumed to be generated by a composite source of differently stationary signals describing the unmoved background and the moving parts, respectively. The results show also that noise is a serious reason for failing in achieving desirable rates of redundancy reduction. In a second part an attempt is made for eliminating irrelevant structures of the picture contents by simulating visual properties of perception within the coding procedure. The relevancy of each differential signal value is determined on the basis of a nonlinear filtering procedure which is applied to the picture content. In this way the equispaced quantizing levels of the DPCM-coder are dynamically expanded. The result is a probability distribution with an increased non-uniformity and thus a reduced entropy of the differential signal.

1. INTRODUCTION

The contribution deals with interframe DPCM-coding of interlace scanned TV-pictures. The investigations were made by computer simulation. Real TV-camera signals were used which had been sampled and digitized and, after that, fed into a computer. The pictures consist mainly of moving portraits acting before an unmoved background. The investigations are separated into two parts. The first one deals with pure redundancy reduction by means of interframe DPCM. In this version a complete reconstruction of the picture sequence is achieved. The second part suggests a new strategy for the decision whether the differential value to be transmitted is relevant for its visual perception. On the basis of the decision the quantizer characteristic is linearly expanded due to the supposed irrelevancy of the differential value.

2. REDUNDANCY REDUCTION

In an earlier contribution [1] the author discussed the model of the interframe covariance function

$$R(x,y,t) = R(o,o,o)\ \exp\ (-\alpha|x|-\beta|y|-\gamma|t|) \quad (1)$$

for application to real time-variant pictures and the derivation of an interframe predictor network for predictive coding of interlace scanned picture sequences. The reason for this investigation was the fact that the corresponding predictive network has a very simple configuration when sequential scanning mode is used [2]. Eq. (1) is separable into the product of three functions which are dependent only on the horizontal coordinate x, the vertical coordinate y, and the time t, respectively. The predictive network was derived from Eq. (1) on the assumption that the power of the differential signal is minimized. The predictive network could be separated into four autonomous parts which are connected in series. Three of these partial networks have the configuration of the well-known previous sample predictor Fig. 1 using the delay of one pel, one line,

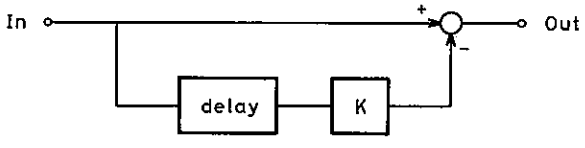


Fig. 1. Network configuration of the previous sample predictor type acting in series connection.

and one frame, respectively. Its weightings K of the delayed signals are chosen due to $\rho_x = \exp(-\alpha\Delta x)$, $\rho_y^2 = \exp(-2\beta\Delta y)$, and $\rho_t = \exp(-\gamma\Delta t)$, respectively, where Δx is the horizontal pixel distance, Δy the spatially vertical line distance, and Δt the time needed for scanning one frame. The fourth partial network contributes only unsignificantly to the expected reduction of the differential signal. It was therefore omitted in the experimental investigations. The expected redundancy reduction of the RMS differential signal yields, calculated on the basis of Eq. (1),

$$\Delta I = 0{,}5\ \mathrm{ld}\ [(1-\rho_x^2)(1-\rho_y^4)(1-\rho_t^2)]\ \mathrm{bit/pel}. \quad (2)$$

In all experimental investigations of redundancy reduction the quantizing steps of the DPCM-encoder were equispaced to those ones of the original signal. First the scanned picture sequence was considered to be a unique and stationary signal stream on which the described coding mechanism was applied. But the results distinctly failed to fit the expected redundancy reduction proposed by Eq. (2). There are two main reasons. It was obvious that the moved and the unmoved parts of the picture content do not have the same stochastic properties and that the composed signal is a non-stationary one. On the other hand the used derivation of the predictor network due to a certain covariance function like Eq. (1) assumes that the signal is stationary. The other reason is that a real camera signal is remarkably corrupted by coloured noise when pick-up tubes of the plumbicon or vidicon type are used. This noise is almost visually irrelevant and thus unimportant when classical transmission techniques are used for handling the signal. But this coloured noise becomes a severe problem when techniques like DPCM are used which take account of each signal value in its full relevance.

In a second approach the coding procedure was changed by introducing a movement detector which controls the predictor. The assembly is depicted in Fig. 2. The signal B to be coded is tested whether three neighbouring pixels from the past of the same frame are identical within a certain tolerance to their counterparts of the previous frame. If this decision is true, the picture content is assumed to be unmoved. Otherwise, if at least one of the three neighbouring pixels is not a replica of the previous frame within a certain tolerance, the picture content is assumed to be a moved one. The disadvantage of this decision rule is that the actual pixel may be a moved one although the tested pixels may belong to an unmoved part of the image sequence, and vice versa. The advantage of this procedure is the self-detecting operation which makes use of the reconstructed signal. Therefore this procedure can identically be applied at the receiver point without the need to transmit any auxiliary switching signal. As so far the movement detector will generate a certain rate of false decisions but it does not expand the transmitted data rate by avoiding the use of an auxiliary signal. The predictor is now switched according to the state of the moving detector. In the unmoved mode only the corresponding pixel of the previous frame is used for prediction. The procedure is performed by only one network Fig. 1 of the previous sample predictor type which has the delay of one frame and a weighting factor close to one. Spatial contributions are ignored. In the moved mode the covariance of the moved picture content is used to determine the three weighting factors ρ_x , ρ_y , and ρ_t , of the three partial predictors according to Fig. 1 and their predictive contributions from the horizontal, vertical, and temporal direction. The weighting factor ρ_t of the temporal contribution is significantly less than unity in the moved mode according to the reduced temporal covariance of a moved picture content.

The results of redundancy reduction including the described movement detector fit the proposed redundancy pretty good. Thus a partially moved picture content may be assumed to be generated by a composite source producing at least two differently stationary signals. The entropy of the differential signal according to the achieved redundancy reduction is in the order of 4 bit/pel for pictures having heavily moved parts. The result depends on the content of moving parts, of course. It depends also on the noise of the used camera signals in the way which has already been described. The results

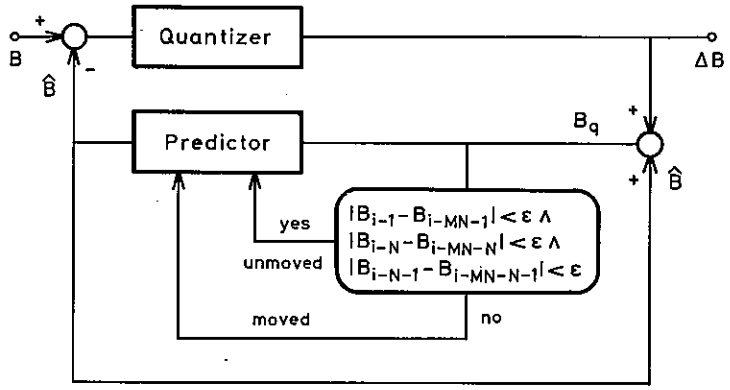


Fig. 2. Block diagram of a DPCM-coder including a movement detector. N is the number of picture elements per line. M is the number of lines per frame.

may become more satisfactory ones when picture sequences generated by a flying spot scanner or, not to say, by synthetically generated pictures would be investigated.

3. IRRELEVANCY REDUCTION

It is well known that the human visual system is significantly imperfect. Nevertheless the normal human observer has an impression of an observed scene which seems to be without imperfection. Among the numerous references to this topic only one [3] shall be cited which relates closely to picture coding. In the view of communication engineering the procedure may be over-simplified in the way that the observed scene is a blurred projection on the retina due to the imperfections of the human eye lens. When the excitation of neighbouring receptors is below a critical threshold, this difference is not visible due to the Weber-Fechner law. The subjective restauration or deblurring is achieved by the so-called lateral inhibition which is due to the well-known Mach effect acting spatially as well as temporally in the neural system.

The application of non-linear filtering methods in order to characterize the relevance of the differential signal is depicted in Fig. 3. The signal B is filtered by a spatial low pass filter LP1 which is a non-recursive and non-causal transversal filter of symmetric shape due to a Gaussian bell-shaped aperture. The output signal $\bar{B}$ defines the "local mean" of the picture content. From this a threshold Q_d is derived consisting of an additive part Q_a and a multiplicative part $\bar{B} \cdot Q_m$ which increases with the local mean $\bar{B}$. On the other hand this blurred picture $\bar{B}$ is subtracted from the original one and yields the significant detail picture B_d. This detail signal B_d is also filtered by the spatial low pass filter LP2 which again has a two-dimensionally and symmetrically bell-shaped aperture. It is also realized by a non-recursive and non-causal transversal filter. The signal B_b seems to be filtered from the signal B due to a spatial band-pass system. This blurred detail signal B_b is clipped by a limiter to the amplitude Q_d. Depending on the appropriate adjustment of the blurring apertures of LP1 and LP2 as well as the threshold Q_d one may expect that the limited signal B_1 indicates visual irrelevancy if its amplitude is below the threshold Q_d. Otherwise, if the limited signal B_1 is the result of clipping, the corresponding detail content is assumed to be relevant for visual perception.

The linearly spaced quantizer characteristic is now expanded due to the observed relevancy or irrelevancy of the pixel. When the amplitude of the limited signal B_1 equals the threshold Q_d, the quantizer characteristic is linearly expanded and spaced by the static increment Q_d. When otherwise the amplitude of the limited signal B_1 is below the threshold Q_d indicating irrelevancy, this signal B_1 is inversely filtered by filter $(LP2)^{-1}$ and yields B_r. The relation $|B_r/B_1|$ in the mean indicates the dynamically expanded irrelevancy scaling of the quantizer in integer multiples of Q_d as depicted in Fig. 3. By this method the number of assigned quantizing steps is reduced in two ways according to the degree of irrelevancy of the differential signal value.

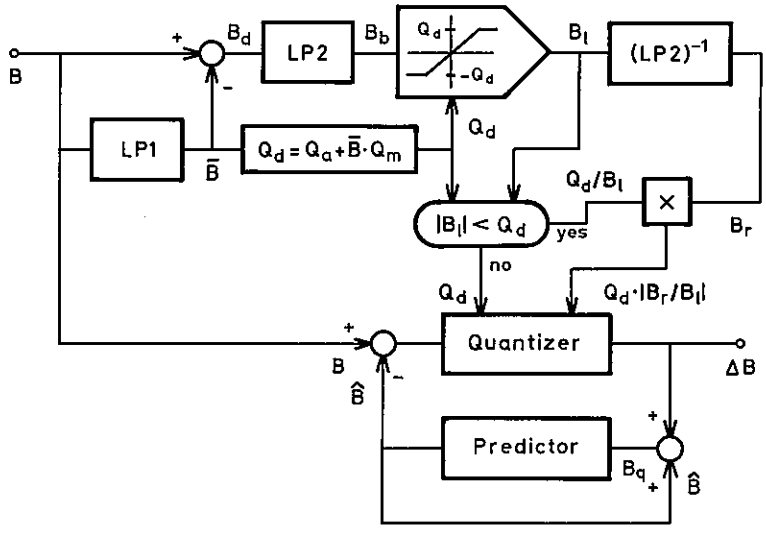


Fig. 3. Block diagram of a DPCM-coder with control of the quantizer by the detected irrelevancy of the differential signal.

The filter $(LP2)^{-1}$ is a non-recursive and non-causal transversal filter like the filters LP1 and LP2. It also simulates a symmetrically shaped aperture. Its spectrum is chosen to be the inverse one of a bell-shaped aperture like that of filter LP2. It is determined in the following way whereby the discussion is reduced to only one dimension for simplicity of explanation. First the coefficients R_k of the bell-shaped and discrete aperture are determined according to

$$R_k = C \cdot \exp\left(-\pi(ka)^2\right) . \qquad (3)$$

The constant a is responsible for the effective width of the aperture whereas the constant C is normalized according to the condition

$$\sum_{k=-\infty}^{\infty} R_k = 1$$

for achievement of an unchanged DC-transmission. The spectrum of the aperture Eq. (3) is

$$r(z) = \sum_{k=-\infty}^{\infty} R_k z^{-k} . \qquad (4)$$

The inverse aperture has the spectrum

$$h(z) = \sum_{k=-\infty}^{\infty} H_k z^{-k} , \qquad (5)$$

defined by $r(z)\, h(z) = 1$, and the coefficients H_k are required for construction of the corresponding inverse transversal filter. The procedure is handled in the following way. In a second step the spectrum $r(z)$ Eq. (4) is separated into the forward part

$$u(z) = \sum_{k=o}^{\infty} U_k z^{-k} \qquad (6)$$

and the backward part

$$u(z^{-1}) = \sum_{k=o}^{\infty} U_k z^{k}$$

generated by the spectral factorization [4]

$$\frac{1}{U_o}\, u(z)\, u(z^{-1}) = r(z) .$$

Spectral factorization of this kind is a well-known procedure for construction of a linear prediction filter from the known covariance function $\{R_k\}$ of a stationary signal to be predicted. The discrete bell-shaped aperture $\{R_k\}$ behaves like such a covariance function having a non-negative spectrum $r(z)$ on $|z| = 1$. Thus the "generating function" $u(z)$ Eq. (6) has only zeros inside the unit circle $|z| < 1$. In a third step the inverse spectrum

$$a(z) = U_o \cdot u^{-1}(z)$$

of the generating function $u(z)$ is determined. This inverse spectrum $a(z)$ has only poles inside the unit circle $|z| < 1$. It is therefore minimum phase and has a stable impulse response $\{A_k\}$. Finally, in the fourth step, the spectrum $h(z)$ Eq. (5) is obtained by the multiplication

$$h(z) = a(z) \cdot a(z^{-1}) ,$$

and the impulse response $\{H_k\}$ of the inverse, non-causal, and non-recursive transversal filter is known.

The investigation of irrelevancy reduction according to the performance described in Fig. 3 were carried out with an unswitched interframe predictor of the three-dimensional previous sample type. Again each directional component of the predictor was chosen according to Fig. 1 having a delay of one pixel, one temporal line distance, and one frame, respectively. The weighting coefficient of each part according to Fig. 1 was simply chosen $K = 1$ so that the complete predictor was not matched and therefore suboptimum. The achieved entropy of the irrelevantly reduced differential signal ΔB is in the order $H = 2{,}0$ to $2{,}5$ bit/pel without any visible degradation of the reconstructed picture sequence. When the degree of reduction comes to an entropy $H = 1{,}5$ bit/pel there is a slight increase of visible noise when an immediate comparison with the original sequence is permitted.

These provisional results illustrate that a reduction of irrelevancy may be performed in the outlined way. The reduced number of quantizing steps and the various patterns which are represented by them including the changed probability distribution may be interpreted in terms of vector quantization. Each member of this relevant set of patterns serves as a substitute for numerous different but visually undistinctable patterns which are originated from the original picture sequence.

REFERENCES

[1] Arp, F.; Interframe Predictor for DPCM Processing of Interlace Scanned Pictures. Eusipco-83, Signal Processing II: Theories and Applications (Ed. H. W. Schüssler), North-Holland, Amsterdam 1983, 247-249.
[2] Arp, F.; Exact Models for Predicting Picture Signals in Adaptive DPCM-Transmission. Applied Physics 6 [1975], 207-213.
[3] Kunt, M., Ikonomopoulos, A., Kocher, M.; Second-Generation Image-Coding Techniques. Proc. IEEE 73 [1985], 549-574.
[4] Arp, F.; Iterative Algorithm for Time Discrete Spectral Factorization. AEÜ 39 [1985], 251-258.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SIMULTANEOUS ESTIMATION OF ROTATION AND TRANSLATION IN IMAGE SEQUENCES*

Hans Burkhardt, Norbert Diehl

Arbeitsbereich Technische Informatik I
Technische Universität Hamburg-Harburg, Postfach 90 14 03
D-2100 Hamburg 90, West-Germany

This paper presents a new, iterative algorithm for the simultaneous estimation of rotation and translation parameters of moving planar objects in grey-scale image sequences. The algorithm combines several advantages such as large stability region, high image-bandwidth-adaptive convergence rate of at least second order near the optimum and a minimum of numeric expense within each iteration step. Furthermore an extension to estimate affine transform parameters and tests of the algorithm using real image data are presented.

1. INTRODUCTION

A well known problem in image processing and scene analysis is the estimation of motion parameters like rotation and translation in image sequences. Applications are found in several areas such as motion compensated image coding, remote sensing by satellites, robotics and biology [1,2,3]. A lot of different algorithms to estimate pure translatoric displacements have been published e.g. [3,4], but only a few papers [5,6,7,8] deal with rotation and translation or give an efficient parameter estimation algorithm based on ordinary grey-scale images without using special features as corresponding points.

In this paper a new fast converging algorithm to estimate rotation and translation simultaneously in grey-scale images is discussed. The estimation problem is formulated as a model adjustment identification problem which is solved by a special minimization algorithm. Also an extension to affine transforms is given.

2. THE ALGORITHM

Two grey-scale images $I_1(x,y)$ and $I_2(x,y)$ are assumed to represent a moving rigid planar object $S(x,y)$ in front of a uniform background where no occlusion effects occur:

$$I_1(\mathbf{x}) = S(\mathbf{x}) = S(x,y)$$
$$I_2(\mathbf{x}) = S(x\cos\phi - y\sin\phi - d_1, x\sin\phi + y\cos\phi - d_2).$$

$I_2(\mathbf{x})$ results from $S(\mathbf{x})$ and therefore from $I_1(\mathbf{x})$ by rotation ϕ and translation d_1, d_2.

To get an appropriate estimate $\hat{\mathbf{T}} = (\hat{\phi}, \hat{d}_1, \hat{d}_2)^T$ of the true motion parameters $\mathbf{T} = (\phi, d_1, d_2)^T$ a *model adaptive identification structure* is used. The motion is modelled by

$$I_m(\mathbf{x}, \hat{\mathbf{T}}) = S(x\cos\hat{\phi} - y\sin\hat{\phi} - \hat{d}_1, x\sin\hat{\phi} + y\cos\hat{\phi} - \hat{d}_2).$$

As model error criterion $J\{e(\hat{\mathbf{T}})\}$ we use the expectation E of the squared model error

$$J\{e(\hat{\mathbf{T}})\} = E\{e^2\} = E\{(I_m(\mathbf{x}, \hat{\mathbf{T}}) - I_2(\mathbf{x}))^2\}.$$

For stationary stochastic signals $J\{e(\hat{\mathbf{T}})\}$ equals twice the negative cross-correlation function of $I_m(\mathbf{x}, \hat{\mathbf{T}})$ and $I_2(\mathbf{x})$ plus an additive constant. $J\{e(\hat{\mathbf{T}})\}$ is assumed to be at least twice differentiable with respect to $\hat{\mathbf{T}}$.

A global search for an optimal parameter vector $\hat{\mathbf{T}} = \hat{\mathbf{T}}^*$ is unrealistic and results in the calculation of the three-dimensional cross-correlation function $R(\phi, d_1, d_2)$ in conjunction with a tremendeous numerical complexity.

We use instead an *iterative minimizing strategy* changing the model parameter vector $\hat{\mathbf{T}}$ well directed until $J\{e(\hat{\mathbf{T}})\}$ reaches its minimum. The proposed algorithm is an extension of a one-dimensional modified Newton-Raphson algorithm [9] which was developed for time-delay-estimation to estimate parameter vectors based on two-dimensional signals. The main structure of the algorithm is given by the iteration:

$$\hat{\mathbf{T}}^{K+1} = \hat{\mathbf{T}}^K - \mathbf{H}^{-1}(\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T}) \cdot \mathbf{g}(\hat{\mathbf{T}}^K).$$

The new estimate $\hat{\mathbf{T}}^{K+1}$ of iteration step $K+1$ is given by the estimate $\hat{\mathbf{T}}^K$ of iteration step K and an innovation given by the multiplication of the inverse of the Hessian $\mathbf{H}(\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T})$ with the gradient vector $\mathbf{g}(\hat{\mathbf{T}}^K)$. Note that the *Hessian* as the second derivatives of the error criterion is always taken *at the optimum* $\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T}$ and not at the actual iteration point $\hat{\mathbf{T}}^{K+1}$ as in the original Newton-Raphson-algorithm. Using the Hessian at the optimum has several advantages which will be discussed in the sequel.

First we will show how to calculate the Hessian at the optimum before starting the iteration without actually knowing the optimum, using the special motion structure of our model. Therefore we distinguish between pure translation and translation in combination with rotation.

* This work was supported by the Deutsche Forschungsgemeinschaft (DFG)

Expressing the Hessian at the optimum by the derivatives of the images we get

$$\frac{\partial^2 J\{e(\hat{\mathbf{T}}^*)\}}{\partial \hat{T}_i \partial \hat{T}_j} = 2\,E\left\{\frac{\partial I_m(\mathbf{x},\hat{\mathbf{T}}^*)}{\partial \hat{T}_i}\frac{\partial I_m(\mathbf{x},\hat{\mathbf{T}}^*)}{\partial \hat{T}_j}\right\},$$

noting that at the optimum $\hat{\mathbf{T}} = \hat{\mathbf{T}}^* = \mathbf{T}$ the image difference $e(\hat{\mathbf{T}})$ vanishes.

For *pure translation* without rotation the last equation can be simplified furthermore if we express the derivatives with respect to the translation parameters by derivatives with respect to the coordinates noting the fact that at the optimum $I_m(\mathbf{x},\hat{\mathbf{T}}^*) = I_2(\mathbf{x})$:

$$\frac{\partial^2 J\{e(\hat{\mathbf{T}}^*)\}}{\partial \hat{d}_i \partial \hat{d}_j} = 2\,E\left\{\frac{\partial I_2(\mathbf{x})}{\partial x_i}\frac{\partial I_2(\mathbf{x})}{\partial x_j}\right\},$$

using the abbreviation $\mathbf{x} = (x = x_1, y = x_2)^T$.

Thus the Hessian does not explicitely depend on the translation parameters d_1, d_2 and can be calculated before starting the iteration. A concrete interpretation of this fact is that at the optimum the cross-correlation function of $I_m(\mathbf{x},\hat{\mathbf{T}})$ and $I_2(\mathbf{x})$ becomes identical with the autocorrelation function of $I_2(\mathbf{x})$. Therefore we can a priori calculate the curvature of the autocorrelation function of $I_2(\mathbf{x})$ instead of the cross-correlation function at the unknown optimum.

However, if we have *rotation and translation simultaneously* the calculation of the Hessian is not as straightforward as in the case with pure translation. Because of the fact that rotation and translation are not independent and therefore do not commute the true rotation angle has to be known if we now want to express the derivatives with respect to the parameter vector by derivatives with respect to the coordinates. For instance now we get at the optimum with $\hat{\phi} = \hat{\phi}^* = \phi$

$$\frac{\partial I_m(\mathbf{x},\hat{\mathbf{T}}^*)}{\partial \hat{d}_1} = -\frac{\partial I_2(\mathbf{x})}{\partial x}\cos\phi + \frac{\partial I_2(\mathbf{x})}{\partial y}\sin\phi.$$

Thus if there is rotation and translation the partial derivatives at the optimum and therefore the Hessian explicitely depend on the unknown parameter ϕ. For the motion model under consideration the Hessian can therefore not be calculated before starting the iteration.

Nevertheless as will be shown in the following the Hessian of a *slightly modified structure* can be calculated beforehand and the advantages of the algorithm can be preserved. We introduce a running coordinate system $\{\mathbf{x}^K\} = \{x^K, y^K\}$ and $\tilde{\mathbf{T}}^{K+1} = (\tilde{\phi}^{K+1}, \tilde{d}_1^{K+1}, \tilde{d}_2^{K+1})^T$ as an additional motion vector describing the estimate at the $K+1$-th iteration on the basis of the coordinate system $\{\mathbf{x}^K\}$ of the K-th iteration.

Thus the relation between the model image $I_m(\mathbf{x},\hat{\mathbf{T}}^K)$ of the K-th iteration step and $I_m(\mathbf{x},\hat{\mathbf{T}}^{K+1})$ of the $K+1$-th iteration step is described as

$$\begin{aligned} I_m(\mathbf{x},\hat{\mathbf{T}}^{K+1}) &= I_m(\mathbf{x},\hat{\mathbf{T}}^K,\tilde{\mathbf{T}}^{K+1}) = S(\mathbf{x}^{K+1}) \\ &= S(x^K\cos\tilde{\phi}^{K+1} - y^K\sin\tilde{\phi}^{K+1} - \tilde{d}_1^{K+1},\ldots) \\ &= S(x\cos\hat{\phi}^{K+1} - y\sin\hat{\phi}^{K+1} - \hat{d}_1^{K+1},\ldots). \end{aligned}$$

Thus the innovation $\tilde{\mathbf{T}}^{K+1}$ is given in the transformed coordinates $\{\mathbf{x}^K\}$ and only indirectly in the coordinates $\{\mathbf{x}\}$; i.e. the new model image $I_m(\mathbf{x},\hat{\mathbf{T}}^{K+1})$ is related to the model image $I_m(\mathbf{x},\hat{\mathbf{T}}^K)$ by the motion vector $\tilde{\mathbf{T}}^{K+1}$. Therefore instead of differentiating the error criterion with respect to $\hat{\mathbf{T}}$ now the error criterion has to be differentiated with respect to $\tilde{\mathbf{T}}$. The slightly modified algorithm is given by

$$\hat{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^K + \tilde{\mathbf{T}}^{K+1} = \mathbf{A}^{K+1}\hat{\mathbf{T}}^K - \tilde{\mathbf{H}}^{-1}\cdot\tilde{\mathbf{g}}(\hat{\mathbf{T}}^K)$$

with the new Hessian $\tilde{\mathbf{H}}$, the new gradient vector $\tilde{\mathbf{g}}$ and the weighting matrix

$$\mathbf{A}^K = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\tilde{\phi}^K & -\sin\tilde{\phi}^K \\ 0 & \sin\tilde{\phi}^K & \cos\tilde{\phi}^K \end{pmatrix}$$

which describes the connection between $\tilde{\mathbf{T}}^{K+1}$, $\hat{\mathbf{T}}^{K+1}$ and $\hat{\mathbf{T}}^K$.

Using this indirect generation of the model image the Hessian $\mathbf{H}$ at the optimum can be calculated off-line without knowing the true motion vector. Assuming that the optimum is reached at iteration step N; i.e. $I_m(\mathbf{x},\hat{\mathbf{T}}^N) = I_2(\mathbf{x})$. The estimate $\tilde{\mathbf{T}}^{N+1}$ of iteration step $N+1$ has to be $\tilde{\mathbf{T}}^{N+1} = 0$. This leads to the following derivatives:

$$\left.\frac{\partial I_m(\mathbf{x},\hat{\mathbf{T}}^N,\tilde{\mathbf{T}}^{N+1})}{\partial \tilde{d}_1^{N+1}}\right|_{\tilde{\mathbf{T}}^{N+1}=0} = -\frac{\partial S(\mathbf{x}^N)}{\partial x^N}$$

$$\left.\frac{\partial I_m(\mathbf{x},\hat{\mathbf{T}}^N,\tilde{\mathbf{T}}^{N+1})}{\partial \tilde{\phi}^{N+1}}\right|_{\tilde{\mathbf{T}}^{N+1}=0} = -\frac{\partial S(\mathbf{x}^N)}{\partial x^N}y^N + \frac{\partial S(\mathbf{x}^N)}{\partial y^N}x^N$$

If we use these expressions to calculate the Hessian we get for instance

$$\frac{\partial^2 J\{e(\hat{\mathbf{T}}^*)\}}{\partial \tilde{d}_1^{N+1}\partial \tilde{\phi}^{N+1}} = E\left\{\frac{\partial S(\mathbf{x})}{\partial x}\left(\frac{\partial S(\mathbf{x})}{\partial x}y - \frac{\partial S(\mathbf{x})}{\partial y}x\right)\right\}.$$

Introducing the parameter vector $\tilde{\mathbf{T}}$ and thus relating the new model image to the old one, the Hessian at the optimum can be expressed only by the derivatives of $S(\mathbf{x}) = I_1(\mathbf{x})$ with respect to the original coordinates $\{\mathbf{x}\}$ and is independent of the true motion vector. A concrete interpretation of this fact is that calculating the derivatives at $\tilde{\mathbf{T}} = 0$ can be interpreted as infinitesimal rotation and translation which in this case do commute. The preceeding formula uses the fact that the expectation value is independent of an arbitrary coordinate system. Therefore all derivatives are given in terms of the original image $S(\mathbf{x})$. This is consistent if we have stochastic stationary signals or if we use spatial averaging instead of the expectation operation for isolated objects in front of a uniform background.

Finally with the abbreviation $\partial/\partial\phi = y\cdot\partial/\partial x - x\cdot\partial/\partial y$ and the operator $\partial = (\partial_1,\partial_2,\partial_3)^T = (\partial/\partial\phi, \partial/\partial x, \partial/\partial y)^T$ the elements of the Hessian $\tilde{\mathbf{H}}$ and the vector $\tilde{\mathbf{g}}$ can be written as:

$$\begin{aligned} \tilde{H}_{ij} &= 2\,E\{\partial_i I_1(\mathbf{x})\partial_j I_1(\mathbf{x})\} \\ \tilde{g}_i(\hat{\mathbf{T}}^K) &= -2\sum_{j=1}^{3} B_{ij}^K\,E\{(I_m(\mathbf{x},\hat{\mathbf{T}}^K) - I_2(\mathbf{x}))\partial_j I_2(\mathbf{x})\}. \end{aligned}$$

Because of the fact that under the preceeding assumptions the expectation value is independent of the used coordinate system it is possible to express the derivatives in the gradient vector as derivatives of image $I_2(\mathbf{x})$ and not as derivatives of the model image $I_m(\mathbf{x}, \hat{\mathbf{T}})$. Thus we have to differentiate $I_2(\mathbf{x})$ only once instead of differentiating $I_m(\mathbf{x}, \hat{\mathbf{T}})$ within each iteration step.

The matrix $\mathbf{B}$ describes the connection between the derivatives with respect to $\tilde{\mathbf{T}}$ and those with respect to the original coordinates $\{\mathbf{x}\}$:

$$\mathbf{B}^K = \begin{pmatrix} 1 & \hat{d}_1^K \sin\hat{\phi}^K - \hat{d}_2^K \cos\hat{\phi}^K & \hat{d}_1^K \cos\hat{\phi}^K + \hat{d}_2^K \sin\hat{\phi}^K \\ 0 & \cos\hat{\phi}^K & -\sin\hat{\phi}^K \\ 0 & \sin\hat{\phi}^K & \cos\hat{\phi}^K \end{pmatrix}.$$

3. PROPERTIES OF THE ALGORITHM

The algorithm has *several advantages* such as large stability region, high image-bandwidth-adaptive convergence rate and a minimum of numeric expense within each iteration step.

The algorithm has in general a much *larger stability range* compared to the normal Newton-Raphson-algorithm. One-dimensionally speaking it is like the gradient algorithm stable up to the next optimum of the error criterion $J\{e(\hat{\mathbf{T}})\}$. Like the normal Newton-Raphson-technique the algorithm has good *signal adaptive properties*. The Hessian which can be interpreted as the second derivatives of the autocorrelation function of the images adjusts to the image bandwidth. Therefore the *convergence rate* of the Euclidean error norm $e = \| \hat{\mathbf{T}}^K - \mathbf{T} \|$ is at least of *second order*, independently of the chosen signals.

If we have pure translation or pure rotation the convergence rate is even of third order. The same is true if all off-diagonal elements of the Hessian are zero

$$\tilde{H}_{ij} = 0 \qquad \text{for} \qquad i \neq j$$

and additionally

$$E\{\frac{\partial S}{\partial x}\frac{\partial S}{\partial x}\} = E\{\frac{\partial S}{\partial y}\frac{\partial S}{\partial y}\}$$

which means that the error criterion $J\{e(d_1, d_2)\}$ for pure translation is rotationally invariant with respect to d_1, d_2. Because of these properties even large parameter vectors can be identified in a few iterative steps.

Another advantage of the algorithm is the *low numeric complexity* within each iteration step because of the ability to calculate the Hessian once before starting the iteration. This is attractive for near real-time implementations [10].

4. EXTENSION TO AFFINE TRANSFORMS

The given algorithm can be extended to estimate affine parameters $\mathbf{T} = (c_{11}, c_{12}, c_{21}, c_{22}, d_1, d_2)^T$ of the coordinate transform

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}.$$

As before the two model parameter vectors $\hat{\mathbf{T}}$ and $\tilde{\mathbf{T}}$ are introduced with

$$\begin{aligned} \mathbf{x}^{K+1} &= \hat{\mathbf{C}}^{K+1} \cdot \mathbf{x} - \hat{\mathbf{d}}^{K+1} \\ &= \tilde{\mathbf{C}}^{K+1} \cdot (\hat{\mathbf{C}}^K \cdot \mathbf{x} - \hat{\mathbf{d}}^K) - \tilde{\mathbf{d}}^{K+1}. \end{aligned}$$

With the operator $\partial = (x \cdot \partial/\partial x, y \cdot \partial/\partial x, x \cdot \partial/\partial y, y \cdot \partial/\partial y, -\partial/\partial x, -\partial/\partial y)^T$ the Hessian $\tilde{\mathbf{H}}$ at the optimum can be written as

$$\tilde{H}_{ij} = 2\, E\{\partial_i I_1(\mathbf{x}) \partial_j I_1(\mathbf{x})\}$$

and the gradient vector as

$$\tilde{g}_i(\hat{\mathbf{T}}^K) = -2 \sum_{j=1}^{6} B_{ij}^K\, E\left\{(I_m(\mathbf{x}, \hat{\mathbf{T}}^K) - I_2(\mathbf{x}))\partial_j I_2(\mathbf{x})\right\}$$

with a matrix $\mathbf{B}$ which again describes the connection between the derivatives with respect to $\tilde{\mathbf{T}}$ and those with respect to the coordinates $\{\mathbf{x}\}$. The iteration structure is changed only slightly. The innovation is given by

$$\tilde{\mathbf{T}}^{K+1} = -\tilde{\mathbf{H}}^{-1} \cdot \tilde{\mathbf{g}}(\hat{\mathbf{T}}^K)$$

and the new motion vector $\hat{\mathbf{T}}^{K+1}$ is related to $\hat{\mathbf{T}}^K$ and $\tilde{\mathbf{T}}^{K+1}$ by the transforms

$$\hat{\mathbf{C}}^{K+1} = \tilde{\mathbf{C}}^{K+1} \cdot \hat{\mathbf{C}}^K \quad , \quad \hat{\mathbf{d}}^{K+1} = \tilde{\mathbf{C}}^{K+1} \cdot \hat{\mathbf{d}}^K + \tilde{\mathbf{d}}^{K+1}.$$

With this algorithm for instance it is possible to estimate the parameter vector of an object which is inclined, rotated and translated.

5. TESTS WITH REAL IMAGE DATA

The algorithm has been tested with real image data. Therefore several scenes with well defined rotation and translation have been digitized and analysed. Image I shows one typical scene used in the experiments digitized by 512 x 512 pixels with marked regions of interest.

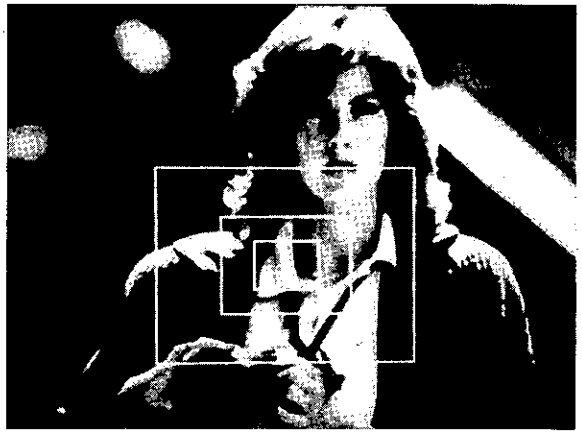


Image I

The first figure gives the joint identification of rotation and translation with the motion vector $\mathbf{T} = (20°, 4, 2)^T$ given in degrees and pixels. The rotation was always around the centre of the marked areas of image I and the smallest of these three areas (51 x 51 pixels) was used as region of interest to calculate the expectation values. The estimated values $\hat{\phi}$ in degrees and $\hat{d}_1, \hat{d}_2$ in pixels are plotted versus the iteration number K.

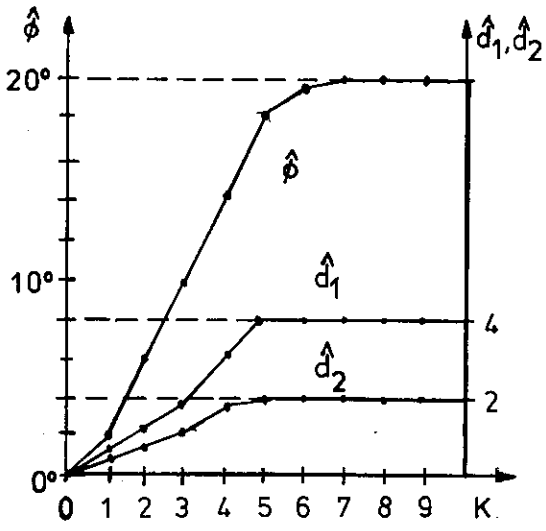


Figure 1

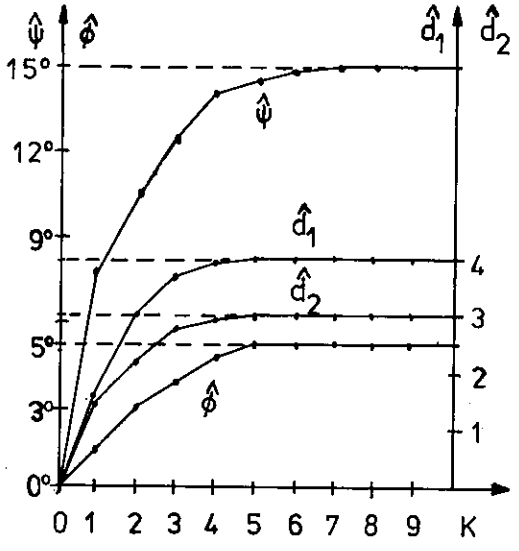


Figure 2

In the second example the parameters of an affine transform are estimated. Therefore the image was inclined by $\psi = 15°$, rotated by $\phi = 5°$, and translated by 4 pixels in x and 3 pixels in y direction. The inclination was around an axis through the centre of rotation. Again the estimated values $\hat{\psi}$, $\hat{\phi}$ and $\hat{d}_1, \hat{d}_2$ are given versus the iteration number K.

6. CONCLUSIONS

The paper describes a fast converging algorithm for the joint estimation of rotation and translation in image sequences. Furthermore an extension of the algorithm to estimate affine transform parameters and tests with real image data are presented.

REFERENCES

[1] Nagel, H.H.: Image sequence analysis: What can we learn from applications. Huang, T.S. (ed.): Image Analysis (Springer 1981)

[2] Nagel, H.H.: Analyse und Interpretation von Bildfolgen. Informatik-Spektrum 8 (1985) pp. 178-200 and pp. 312-327.

[3] Huang, T.S. (ed.): Proc. Nato Advanced Study Inst. on Image Sequence Processing and Dynamic Scene Analysis. Braunlage 1982 (Springer 1983)

[4] Netravalli, A.N.; Robbins, J.O.: Motion-compensated television coding: part I. Bell Syst. Techn. J. 58 (1979) pp. 631-670.

[5] Schalkoff, R.J.; McVey, E.S.: A model and tracking algorithm for a class of video targets. IEEE Trans. PAMI -4, (1982) pp. 2-10.

[6] Legters, G.R.; Young, T.Y.: Mathematical model for computer image tracking. IEEE Trans. PAMI-4, (1982) pp. 583-594.

[7] Axelsson, S.R.J.: On optimum algorithms for imaging tracking systems. Kunt, M.; de Coulon, F. (eds.): Signal Processing: Theories and Applications (North-Holland, 1980) pp. 723-728.

[8] Lenz, R.; Gerhard, A.: Adaptive geometrische Transformationen zur Mustererkennung mit Hilfe eines linearen, lokalen Distanzmaßes. Niemann, H.(ed.): Mustererkennung 1985, 7. DAGM-Symposium, Informatik-Fachberichte 107 (Springer, 1985) pp. 112-177.

[9] Burkhardt, H.; Moll, M.: A modified Newton-Raphson-search for the model-adaptive identification of delays. Isermann, R. (ed.): Identification and System Parameter Estimation (Pergamom, 1979) pp. 1279-1286.

[10] Diehl, N.; Burkhardt, H.: Planar motion estimation with a fast converging algorithm. Submitted for publication.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REAL TIME PICTURE PROCESSING SYSTEM WITH TWO-DIMENSIONAL FILTERING AND OFFSET MODULATION

E. Güttner

University of Dortmund, FRG

Concerning the horizontal and vertical resolution the picture quality in a conventional television system can be considerably improved with digital signal processing. Basing on the conception of error free picture scanning and flat field reproduction [1] a real time picture processing system was developed at the University of Dortmund including two-dimensional filtering at the transmitter and receiver end and offsetmodulation. The basic principles and the mode of operation of the system realized is described in this paper.

1 INTRODUCTION

A considerable enhancement of the vertical resolution in a conventional television system is possible with the conception of error free picture scanning and flat field reproduction [1]. Offset sampling combined with two-dimensional pre- and postfiltering yields an adequate horizontal resolution. Both methods are based on frame processing and picture pick-up and reproduction with a doubled number of lines compared to the transmission standard.

The two-dimensional bandlimitation in diagonal direction which is necessary for offsettransmission can be achieved, however, without doubling the line number at the picture pick-up. The acceptance of some vertical aliasing is of no severe disadvantage [2,3].

At the University of Dortmund a real time picture processing system was developed based on these conceptions. The system employs two-dimensional filtering at the transmitter and receiver end and includes offsetmodulation as well.

2 CONVENTIONAL TELEVISION PROCESSING

The picture scanning process in a television system can be described as two-dimensionally sampling of the picture signal $b(x,y,t)$ in vertical and temporal direction.

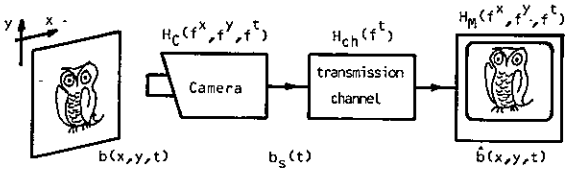


Fig. 1 Linear model of a television system

The picture signal, filtered by the transfer function $H_C(f^x,f^y,f^t)$ of the camera, describing the influence of the optical system, scanning beam, and storage effects, is subsequently sampled and transmitted as a one-dimensional, time dependent video signal (Fig. 1). The limited bandwidth of the transmission channel results in a horizontal bandwidth limitation of the picture signal $\hat{b}(x,y,t)$ reconstructed line by line by the monitor.

While the spatial reconstruction of the sampled picture is supported by the transfer function $H_M(f^x,f^y)$ of the monitor, the only reconstruction filter in temporal direction is the visual system itself.

Both, prefiltering by the camera transfer function as well as reconstruction filtering are defective regarding the sampling theorem. The picture quality is impaired by aliasing and periodic spectra, the sampling raster is visible (line structure, flickering, motion judder).

3 ALIAS-FREE PICTURE SCANNING WITH OFFSET SAMPLING AND FLAT FIELD REPRODUCTION

A conception for overcoming the deficencies of the conventional television system as far as sampling and reconstruction in vertical direction is concerned, was proposed in [1]. Doubling the line number at the picture pick-up and thus increasing the vertical sampling frequency to $f^y_o=2\cdot f^y_s$ offers the possibility of supporting the filtering by the camera transfer function with a digital vertical filter with subsequent down-conversion to the the standard line number. The video signal transmitted is free from aliasing.

Similarly, at the receiver end the picture is up-converted to a high line number signal by an interpolating vertical low-pass filter and reproduced by a high line number monitor. As the periodic spectra are now separated by the stop band regions of the vertical filter and finally attenuated by the transfer function of the monitor, merely the basic spectrum is visible (flat field reproduction).The vertical resolution reaches the theoretical limit.

An adequate increase of horizontal resolution can be achieved with the conception of offset-sampling combined with two-dimensional bandlimitation in diagonal direction [1]. The proposed high line number offset sampling raster (line spacing $y_0=y_s/2$)

$$d_0(x,y) = \text{Ш}_{y_s}(y)\text{Ш}_{x_s}(x) + \text{Ш}_{y_s}(y-y_0)\text{Ш}_{x_s}(x-x_0), \text{ with} \quad (1)$$

$$\text{Ш}_{y_s}(y) = \sum_n \delta(y-ny_s);\ \text{Ш}_{x_s}(x) = \sum_m \delta(x-mx_s), \quad (2)$$

can be described as the sum of two sampling rasters, mutually offset, defined by the product of two orthogonal trains of line masses with equal spacing in vertical (y_s) and horizontal (x_s) direction. The offset sampling results in a repetition of the horizontally bandlimited spectrum at multiples of $f^x_0=1/x_0$, $f^y_0=1/y_0$ and odd multiples of $f^x_s=1/x_s$, $f^y_s=1/y_s$ (Fig. 2a).

After two-dimensional filtering a down conversion in diagonal direction to an offset raster with standard line spacing y_s is possible.

$$d_{s0}(x,y) = \text{Ш}_{x_c}(x)\text{Ш}_{y_c}(y) + \text{Ш}_{x_c}(x-x_s)\text{Ш}_{y_c}(y-y_s),$$
$$\text{with} \quad x_c = x_s/2 \text{ and } y_c = y_s/2. \quad (3)$$

The alias free periodic spectra border upon each other without overlapping (Fig. 2b).

At the receiver (Fig. 2c) the periodic spectra are separated again by the stop band regions of the diagonal filter interpolating a high line number picture signal (Fig. 2d). With the attenuation of objectionable spectra by the monitor transfer function and the viewer's eye, the reconstruction of the picture is nearly perfect.

The vertical resolution reaches the theoretical limit and the horizontal resolution is enhanced

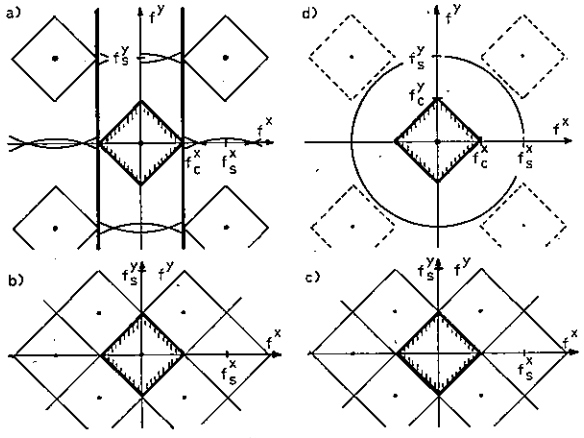


Fig. 2 Alias-free picture scanning and flat field reproduction with offset sampling

up to a factor of two compared to the conventional system, the picture quality is considerably improved [4]. The gaps separating the periodic spectra allow the picture sharpness to be enhanced by nonlinear processing as in the case of vertical filtering discussed in [5].

4 PICTURE PROCESSING SYSTEM

Fig. 3 shows the block diagram of the picture processing system with high line number picture pick-up and reproduction. At the transmitter

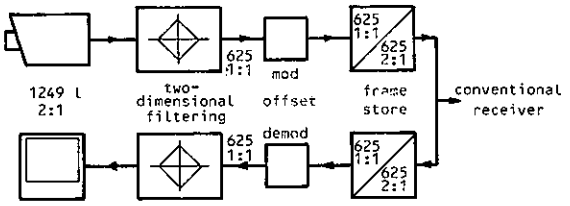


Fig. 3 Television system with two-dimensional pre- and post processing

the picture signal is diagonally filtered, offset modulated and subsequently transmitted with synthetic line interlace. The receiver recombines the corresponding fields. After offset demodulation and two-dimensional reconstruction filtering the picture is displayed by the monitor.

4.1 Transmitter Processing

The principle of offset sampling is based on frame processing and high line number picture pick-up. As available high line number cameras operate in line interlace mode, a frame, processed by the system, consists of two successive fields of the camera signal resulting in objectionable motion blurr even at low velocities of the moving object.

Accepting some vertical aliasing the two-dimensional bandlimitation, necessary for offset transmission, is therefore carried out with a standard line number picture, i.e. one field of a high line number line interlaced signal.

The spatial spectrum of the 625-line signal at the input of the processing system (Fig. 4a) shows the periodic repetition of the picture spectrum, filtered by the camera transfer function, at multiples of the vertical sampling frequency $1/y_s=f^y_s \hat{=} 625$ lines.

As proposed in [3] the subsequent two-dimenal bandlimitation in diagonal direction is performed by two one-dimensional transversal filters beeing cascaded. Each of them consists of two 21-tap transversal filters, which operate in parallel at a clock frequency of 16 MHz, sharing a common delay line.

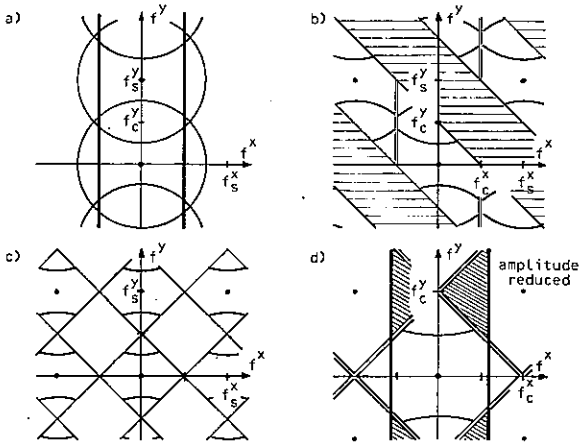


Fig. 4 Transmitter processing

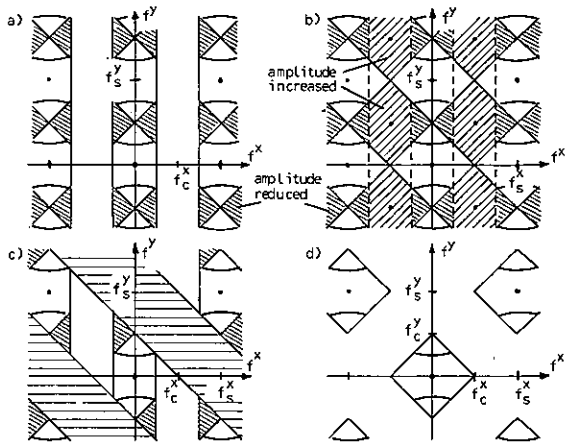


Fig. 5 Receiver processing

The first filter acts as a low-pass interpolator in the direction of the ascending diagonal (Fig. 4b), calculating a high line number picture. Bandlimitation in the direction of the descending diagonal is accomplished by the second filter, simultaneously down-converting the signal to 625 lines.

The resulting spectrum (Fig. 4c) shows, that the basic spectrum, bandlimited as desired, is periodically repeated at multiples of the horizontal and vertical sampling frequency, separated by signal free regions in diagonal direction. The corresponding spatial sampling raster $d_S(x,y)$ can be described as the sum of two offset rasters, mutually offset,

$$\begin{aligned} d_S(x,y) &= \text{Ш}_{x_S}(x)\,\text{Ш}_{y_S}(y) \\ &= (\text{Ш}_{x_C}(x)\,\text{Ш}_{y_C}(y) + \\ &\quad + \text{Ш}_{x_C}(x-x_S)\,\text{Ш}_{y_C}(y-y_S)) + \\ &\quad +(\text{Ш}_{x_C}(x)\,\text{Ш}_{y_C}(y-y_S) + \\ &\quad + \text{Ш}_{x_C}(x-x_S)\,\text{Ш}_{y_C}(y)), \qquad (4) \end{aligned}$$

with the two-dimensional Fourier transform

$$\begin{aligned} D_S(f^X,f^Y) &= \tfrac{1}{2}f^X_S f^Y_S\cdot(\text{Ш}_{f^X_S}(f^X)\,\text{Ш}_{f^Y_S}(f^Y) + \\ &\quad + \text{Ш}_{f^X_S}(f^X-f^X_C)\,\text{Ш}_{f^Y_S}(f^Y-f^Y_C)) + \\ &\quad +\tfrac{1}{2}f^X_S f^Y_S\cdot(\text{Ш}_{f^X_S}(f^X)\,\text{Ш}_{f^Y_S}(f^Y) + \\ &\quad +(-1)\cdot\text{Ш}_{f^X_S}(f^X-f^X_C)\,\text{Ш}_{f^Y_S}(f^Y-f^Y_C)). \qquad (5) \end{aligned}$$

Eq. 5, defining the repetition points of the spectrum (Fig. 4c), shows that the amplitude of the spectra at $\pm f^X_C$ can be controlled by differentially weighting the two offset rasters of Eq. 4. With a weighting factor of 4/3 for the first and 2/3 for the second raster these spectra appear with amplitude 1/3 thus avoiding picture quality impairment for the conventional receiver [6].

Fig. 4d shows the spectrum of the transmitted signal, partial spectra with reduced amplitude in hatching. The processed 625-line picture is transmitted with synthetic line interlace in two successiv fields of the standard 625-line television system.

4.2 Receiver Processing

By use of a frame store the corresponding fields of the transmission signal are recombined. The orthogonal sampling raster used is by synchronization identical to that of the transmitter processing (Eq. 4), necessary for a perfect offset demodulation [7].

Prior to offset demodulation the picture signal is filtered by a horizontal low-pass filter, with Nyquist slope at $f^X_C/2$, thus enabling the correct recombination of basic and offset spectrum (Fig. 5a). By weighting the two offset rasters (Eq. 4) with +4 and -2 respectivly, the offset demodulation is achieved. The basic spectrum is completed without a transition, but at $\pm f^X_C$ spectra with considerably increased amplitude appear (Fig. 5b).

As at the transmitter, the two-dimensional reconstruction filtering is accomplished by two one-dimensional transversal filters beeing cascaded. Each of them consists of two 15-tap filters, operating in parallel sharing a common delay line. To avoid an increase of wordlength offset demodulation is carried out within the second diagonal filter.

Fig. 5c shows the spectrum at the output of the first filter and Fig. 5d the high line number signal spectrum at the output of the filter.

After horizontal bandlimitation to $|f^X|<f^X_C$ and final filtering by the monitor transfer function, merely the basic spectrum is visible. Even at line interlace reproduction the viewer has the impression of a line free picture with considerably improved horizontal resolution.

5 MOTION ADAPTIVE RESOLUTION CONTROL

Due to frame processing, the temporal resolution of the system described is limited to the transmission of only 25 pictures per second. Faster movement, particularly of high contrast picture regions, appears jerky.

This can be avoided by fading over to field processing in picture regions showing faster movement [8]. However, in this mode of operation the transmissible resolution for vertical frequencies is halved and the horizontal resolution is limited by the bandwidth of the transmission channel.

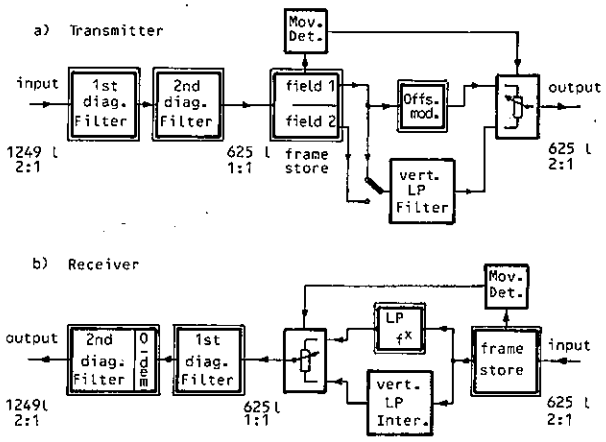


Fig. 6 Motion adaptive resolution control

To prevent field aliasing, an additional vertical filtering at the transmitter is necessary (Fig. 6a), limiting the bandwidth of the signal to one half of the sampling frequency of a transmitted field. The motion detector controls the fading over between the offset modulated signal with synthetic line interlace and the vertically bandlimited signal.

At the receiver each transmitted field is upconverted to a 625-line signal by the vertical low-pass interpolation filter. The motion detector controls the fading over between the 625-line signal generated by the frame store, this signal is horizontally bandlimited by the Nyquist filter, and the vertically interpolated signal. Motion adaptive control of the diagonal filter and the offset modulation, is not necessary

6 CONCLUSION

With high line number picture pick-up and reproduction and two-dimensional signal processing at the transmitter and receiver, the deficencies of the conventional television system can be overcome, yielding a considerable enhancement of subjective picture quality.

At the University of Dortmund a real time picture processing system was developed, employing two-dimensional diagonal filtering at the transmitter and the receiver and including offset modulation as well.

The two-dimensional bandlimitation is performed by two one-dimensional transversal filters being cascaded. The filter koefficients were optimized considering picture sharpness and resolution by computer simulation of the total system.

ACKNOWLEDGEMENTS

The author would like to thank the German Ministry of Research and Technology for supporting these studies. His thanks are also given to his colleagues for helpful discussion and supporting the realization of the system described.

REFERENCES

[1] Wendland, B.: High Definition Television Studies on Kompatible Basis with Present Standards. "Television Technology in the 80's", SMPTE, Scarsdale, New York, 1981, pp. 124-131

[2] Schröder, H.: On Vertical Filtering for Flicker-Free Television Reproduction. In "Signal Processing II: Theories and Applications". pp.167-170

[3] Schröder, H; Elsler, H.: Planare Vor- und Nachfilterung für Fernsehsignale. ntz-Archiv vol. 4 (1982), No. 10, pp. 303-312

[4] Stollenwerk, F.: Kompatibel verbesserte Fernsehsysteme und ihre subjektive Bewertung. Fernseh & Kino-Technik, vol. 39 (1985), No. 2, pp. 64-72

[5] Schröder, H; Elsler, H: Fritsch, M; Nonlinear Picture Enhancement Techniques for Vertically Interpolated TV-Signals. this volume

[6] Eiberger, B.: Kompatible Auflösungserhöhung bei einem Fernsehsystem nach Standard-PAL. Dissertation, University of Dortmund, 1983

[7] Güttner, E; Silverberg, M; Plantholt, M; Schröder, H; Konzeption zur kompatiblen Verbesserung der Bildqualität bei PAL-Übertragung. Fernseh & Kino-Technik, vol. 39 (1985), No. 3, pp. 115-122, No. 4, pp. 187-193

[8] Wendland, B.: High Quality Television by Signal Processing. 2nd International Conference on new systems and services in telecommunications, Liège, 1983

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MOTION ESTIMATION AND SUBBAND CODING USING QUADRATURE MIRROR FILTERS

Achim von BRANDT

Siemens AG, ZT ZTI INF 121
Otto-Hahn-Ring 6
D-8000 München 83

An application of quadrature mirror filters for motion compensated subband coding of image sequences is presented. The coder uses two-dimensional subband splitting and movement compensated predictive coding in the time axis. Motion estimation is based on a hierarchy of filtered and subsampled input pictures. For both subband splitting and motion estimation, quadrature mirror filters are used. This enables a reduction of the coder complexity.

1. INTRODUCTION

For transmission of color image sequences at 2 Mbit/s, e.g. for videoconference applications, a coding procedure based on two-dimensional subband splitting and interframe predictive coding has been presented /1/. The procedure utilizes a filter bank of half-band quadrature mirror filters (QMF, /2/) for sequential lowpass-highpass decomposition of the images at the coder and for reconstruction of the original image at the decoder. The algorithm for coding the subband signal samples is very similar to the procedure for coding the transform coefficients in a recently developed Discrete Cosine Transform hybrid coder /3/.

Meanwhile, the coder has been significantly improved by re-designing the quadrature mirror filters and by employing a new subband splitting scheme /4/. Moreover, we are investigating whether this coder can also be used for image transmission at lower bit rates. At the moment, we are considering a transmission rate of 320 kbit/s. To this end, motion compensated prediction shall be incorporated into the interframe DPCM loop. This requires estimation of motion vectors from the input image sequence or by comparison of the input sequence to the content of the DPCM image memory /5/.

An important task of motion estimation algorithms is the ability of efficient recognition of large motion vectors. For this purpose, it has been proposed to use a hierarchy of low-pass filtered and subsampled images for sequential (or parallel) estimation of motion vectors /6/. However, from the subband splitting process, a series of subsampled images is already available. Hence hierarchical motion estimation and subband splitting can be combined naturally by using the same QMF filter bank.

In section 2, we give an outline of the subband coder, specifying the quadrature mirror filters of the horizontal-vertical filter bank. In section 3, motion estimation using quadrature mirror filtered images is presented. In section 4 we give a block diagram of the complete motion compensated subband coder. Finally in section 5 an application example of the motion estimation scheme is shown.

2. SUBBAND CODING

We use three different FIR lowpass filters with 13, 7 and 2 coefficients together with their highpass counterparts as quadrature mirror (QM) filters for horizontal and vertical subband decomposition. The coefficients $h_{L7}(i)$ and $h_{L13}(i)$ for the first two filters, having an impulse response which is symmetrical around the zero coefficient, are displayed in table 1.

i	0	±1	±2	±3	±4	±5	±6
$h_{L7}(i)$	618	262	-53	-6	0	0	0
$h_{L13}(i)$	566	308	-45	-64	26	12	-8

Table 1: Coefficients of quadrature mirror lowpass filters with 7 and 13 coefficients, resp.

The corresponding QM highpass filters $\{h_{H7}(i)\}$ and $\{h_{H13}(i)\}$ are given by:

$$h_{Hj}(i) = (-1)^i \, h_{Lj}(i)$$

where j=7 or 13, resp.

By applying these filters to an input signal, the signal is split into a lowpass and a highpass component, which both can be subsampled by a factor of 2 without loss of information. For this purpose, alternate subsampling must be applied /7/, i.e. the even numbered samples are taken from the lowpass channel and the odd numbered samples from the highpass component. These subsampled signals are interlaced in order to yield a combined lowpass-highpass signal.

The original signal can be reconstructed from this combined signal by the same procedure: lowpass as well as highpass filtering, alternate subsampling and recombination of both channels /7,4/.

The third QM lowpass and highpass filter pair, with two coefficients each, merely consists of computing the sum or difference, resp., of two adjacent samples, as in a Walsh Hadamard transform of blocksize 2.

These filters are repeatedly applied horizontally and vertically in order to split the image signal into 31 subbands, as shown in fig. 1.

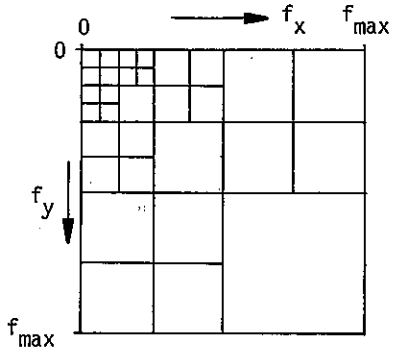


Fig. 1: Horizontal-vertical subband splitting into 31 subbands. f_x: horizontal frequency; f_y: vertical frequency.

The result of this subband splitting process is a transformed image containing the signal samples of all 31 subband signals. The sequence of transformed images is further processed by means of interframe prediction, adaptive subband selection, entropy coding and post-buffer control, as will be explained in section 4.

During the subband decomposition, lowpass filtered and subsampled versions of the image are available which can be used for motion estimation.

3. MOTION ESTIMATION

3.1 Subsampled Images from Subband Splitting

After application of the first QM filter pair (LP and HP filter with 13 coefficients each) in horizontal and vertical direction, the input image has been decomposed into 4 subbands, as shown in fig. 2. A sub-image which has been subsampled by a factor of two in each direction belongs to each of these four subbands A, B, C and D. The subimage which corresponds to subband A is a lowpass filtered and subsampled version of the original image. This sub-image "A" is retained for subsequent motion estimation.

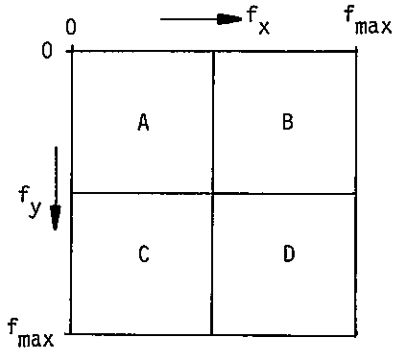


Fig. 2: Initial image decomposition into four subbands, A, B, C and D, in the frequency domain.

After application of the next subband splitting step which utilizes the 7-tap filters, we end up with a frequency chart containing 13 subbands as shown in fig. 3.

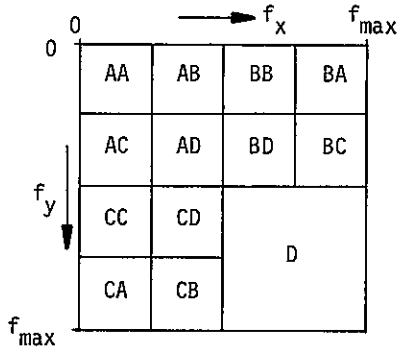


Fig. 3: Intermediate image decomposition into 13 subbands.

The subbands A, B, and C (see fig. 2) have now been splitted into 4 smaller frequency regions while subband D stayed unchanged. Subband AA represents a sub-image which has been subsampled by a factor of 4 in both directions, compared to the original image.

These two subimages corresponding to the subbands A and AA, and the original image, are the basis for a three-step motion estimation procedure for determining motion vectors for changing image blocks.

3.2. Hierarchical Three-Step Block Matching Procedure

There are basically two different principles for motion vector estimation: block matching and differential methods /8/. The block matching method consists of estimating the motion of small blocks of pixels by minimization of an interframe distortion function. Differential methods consist of determining motion vectors by means of iterations which are based on the spacial and temporal pixel differences.

In both methods, efficient estimation of large motion vectors is a main problem. The difficulties can be reduced by using lowpass filtered and subsampled images for the first steps of the estimation algorithm, or by estimating large vectors in subsampled images only, as has been suggested by Burt /6/.

Here we present a block matching procedure which is an extension of the three-step method of Koga et al. /9/. Let I(k) be the image at time k and I(k-1) the previous image. We try to estimate the displacement of objects from time k-1 to time k by means of hierarchical block matching. For this purpose, the sub-images "A" and "AA", as defined in the previous section, are calculated from I(k) as well as from I(k-1), which results in the four sub-images A(k), AA(k), A(k-1) and AA(k-1). The image AA(k) is subdivided into blocks of 4x4 pixel corresponding to blocks of 16x16 pixel in the original image I(k).

In subimage AA(k-1), a search area of 10x10 pixel, corresponding to 40x40 in the original image, is defined for each block in AA(k). By matching the 4x4 pixel blocks of subimage AA(k) to the respective search areas, one of 49 possible motion vectors is assigned to each block. The mean of absolute differences (MAD) has been adopted as the matching criterion.

In a second step, the sub-images A(k) and A(k-1) are used to refine the initial motion vector estimate. For this purpose, blocks of size 8x8 pel are defined in A(k), and search areas of size 10x10 pel in A(k-1), where the center of the search area is displaced by the initial motion vector, multiplied by 2. One out of 9 possible motion vectors is selected at this step.

In the third step, the original images, I(k) and I(k-1) are used for the last refinement of the motion vector. The blocksize in this last step is 16x16 pel and the search areas have 18x18 pel.

So the largest number of search points (49 possible vectors) must be processed in the first step where the block size is only 4x4 pel rather than 16x16 pel as in the original image. Hence a large overall search area can be managed with moderate computation load, leading to a significant reduction of interframe prediction error even in areas with a large displacement of objects.

4. MOTION COMPENSATED SUBBAND CODING

A block diagram of the motion compensated subband coder is shown in fig. 4. At the input side, QM filters are applied to the image sequence to transform the images into subband signals and to generate the sub-images A(k) and AA(k) as a by-product. The originals as well as the subimages are delayed using image memories M and MS where MS needs only 5/16 (=1/4 + 1/16) of the capacity of M. From these input signals, the motion estimator ME determines the motion vectors. These are used for motion compensated prediction within the DPCM loop. The difference of I(k) and the predicted image is uniformly quantized (Q) and coefficient selection (CS) is applied. In the DPCM loop, the coefficients (i.e. subband samples) are inverse transformed by another QMF application (IQMF) and stored in the image memory M, whose motion compensated output is again forward transformed. The coefficient codewords, motion vectors and some additional overhead information are input to the entropy coder (C). By means of an output buffer and feedback to Q and CS, a constant data rate is obtained.

5. RESULTS

The ability of reducing the displaced frame difference between two consecutive pictures by means of the hierarchical blockwise motion estimation procedure was evaluated. A temporally subsampled version of the so-called COST sequence "Splitscreen" was used as the test

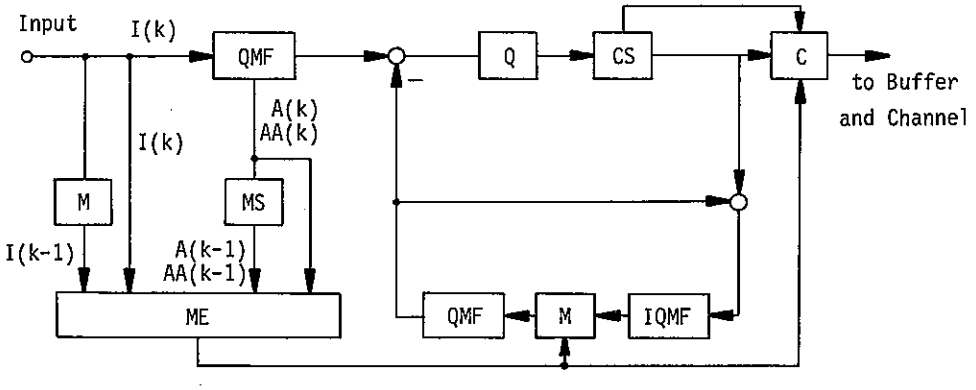


Fig. 4: Structure of motion compensated subband coder.

sequence. The sequence contained every 6th field of the original scene, resulting in a field rate of 8.33 Hz. In fig. 5 the subimage A and AA of image no. 2 are shown. In fig. 6a the difference between image no. 2 and image no. 3 of the temporally subsampled sequence without applying motion compensation is displayed while fig. 6b shows the displaced frame difference between image 2 and 3 after motion compensation. The mean squared frame difference has been reduced in this example from 537.9 to 99.7 by displacing each block of 16 by 16 pel of image no. 2 as indicated by the motion vector field.

6. CONCLUSION

The application of quadrature mirror filters (QMF) for subband coding and for calculation of subimages for motion estimation has been presented. The use of the same filters for motion estimation and for image transformation within the coder enables a facilitated integration of motion compensated prediction into the coder.

The hierarchical motion estimation principle is not restricted to block matching techniques. Object oriented motion estimation and differential methods are likewise suitable for hierarchical estimation based on subimages. By using these possibilities and by optimizing the coder parameters, a very good image quality at a transmission rate of 320 kbit/s seems to be achievable.

REFERENCES

/1/ A.v. Brandt, "Sub-band coding of videoconference signals using quadrature mirror filters", in: M.H. Hamza (ed.), Applied Signal Processing, Proc. of the IASTED Int. Symp., Paris, June 19-21, 1985, pp. 212-215.

/2/ M. Vetterli, "Multi-dimensional sub-band coding: some theory and algorithms", Signal Processing 6 (1984), pp. 97-112.

/3/ H. Hölzlwimmer, W. Tengler, A.v. Brandt, "A new hybrid coding technique for videoconference applications at 2 Mbit/s", 2nd Int. Techn. Symp. on Optical and Electro-optical Applied Science and Engineering, Cannes, France, 2-6 Dec., 1985.

/4/ A.v. Brandt, "Teilbandcodierung von Bewegtbildsequenzen mit 2 Mbit/s", submitted to "Frequenz".

/5/ G. Kummerfeldt, F. May, W. Wolf, "Coding television signals at 320 and 64 kbit/s", 2nd Int. Techn. Symp. on Optical and Electro-Optical Applied Science and Engineering, Cannes, France, Dec. 1985.

/6/ P.J. Burt, "Fast algorithms for estimating local image properties", Computer Vision, Graphics and Image Processing 21 (1983), pp. 368-382.

/7/ G. Wackersreuther, "On the design of filters for ideal QMF and polyphase filter banks", Archiv Elektr. Übertr. (AEÜ) 39 (1985) 2, pp. 123-130.

/8/ H.G. Musmann, P. Pirsch, H.-J. Grallert, "Advances in picture coding", Proc. IEEE 73 (1985) 4, pp. 523-548.

/9/ T. Koga, K. Iinuma, A. Hirano, Y. Iijima, T. Ishiguro, "Motion-compensated interframe coding for video conferencing", in NTC 81, Proc., pp. G5.3.1-G5.3.5 (1981).

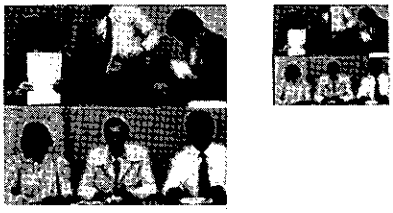


Fig. 5: Subimages of image no. 2.

Fig. 6a: Frame difference (enhanced by 4).

Fig. 6b: Displaced frame difference (enhanced by 4).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

STATISTICAL DPCM CODEC FOR TRANSMISSION OF TV SIGNALS AT 30 Mbit/s

C.C. EVCI and J.Y. BOISSON

T.R.T. 5, Avenue Réaumur-92350 - Le Plessis-Robinson - FRANCE

In this paper, we describe an adaptive DPCM (ADPCM) codec which is capable of component (luminance-chrominance) coding of broadcast-quality colour TV signals at 30 Mbit/s. First, several 2-D prediction schemes associated with an adaptative quantizer are compared. Secondly, the superiority of 3-D prediction scheme is demonstrated. Finally, the effects of variable word length (VLC) limitations are examined. The results show that the proposed codec with statistical coding has a considerable potential due to its good picture quality.

1. INTRODUCTION

The front runner in broadcast quality picture coding is differential PCM (DPCM) [1, 2]. When compared to transform coding DPCM has advantage for a given degree of hardware complexity [3].

In our studies, pictures digitized according to the 601. CCIR standard [4] have been used for simulation purposes : still pictures to optimize (2-D) predictor, quantizers and then two picture sequences ("car sequence" and "girl sequence") to validate the results obtained within the two spatial dimensions and to take benefit of the third dimension, namely the temporal dimension.

The three video components Y, U-V, digitized at 13,5 MHz and 6,75 MHz respectively, lead to a bit-rate of 216 Mbit/s. To fit in a 30 Mbit/s digital stream, these components will be compressed using several bit rate reduction steps. A first reduction from 216 Mbit/s to 165,6 Mbit/s (Y: 82,8 Mbit/s - U-V: 82,8 Mbit/s) is obtained just by removing line and field blanking intervals.

The luminance signal is not subsampled so that the whole luminance bandwith is preserved. DPCM processing allows to reduce the luminance bit rate from 82,8 Mbit/s to 41,4 Mbit/s. A further reduction from 41,4 Mbit/s to 23,8 Mbit/s is due to statistical coding. Hence, each luminance pel requires 2,3 bits. Such a scheme for the luminance leaves 6,2 Mbit/s for the chrominance signals. The chrominance signals are subsampled by a factor of 4. A first factor of 2 is due to field sequential colour transmission. The second is obtained by skipping one pel out of two according to a line and frame quincunx lattice. DPCM processing reduces the chrominance bit rate from 20,7 Mbit/s to 10,35 Mbit/s, and statistical coding from 10,35 Mbit/s to 6,2 Mbit/s. Each initial chrominance pel requires 0,6 bit. Now, let us see how to reach these goals.

2. COMPARISON OF PREDICTORS

2.1. General description

DPCM has been extensively described so we will only mention its salient features as shown in figure 1.

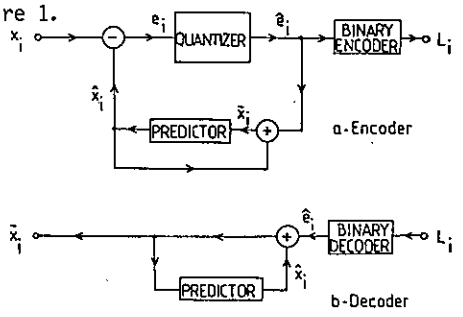


Figure 1

In this arrangement, it is clear that the prediction accuracy is affected by quantization noise. This limitation effect in the performance of the prediction is usually emphasized at low data rates where coarse quantization is used and the predictor-quantizer mismatching becomes significant. This seems to suggest that the optimization of the predictor should be performed using the statistics of the decoded pels. The transmission bit-rate concerned is however relatively high, allowing for the predictors to be designed using the statistics of the original image samples as described in the subsequent sections.

2.2. Predictor sets

The correlations between pels along the lines, between the lines, and between the fields offer great flexibility in designing predictors so that they can operate in one, two or even three dimensions. In this section it will be exclusively dealt with 2-D predictors, i.e., purely spatial predictors, using pels of the

same field. The predicted sample $\hat{x}_i$ in Figure 1 is given by

$$\hat{x}_i = \sum_{k}^{N} a_k \tilde{x}_{i-k} \quad (1)$$

where N is the number of prediction coefficients. The positions of the pels are shown in Figure 2.

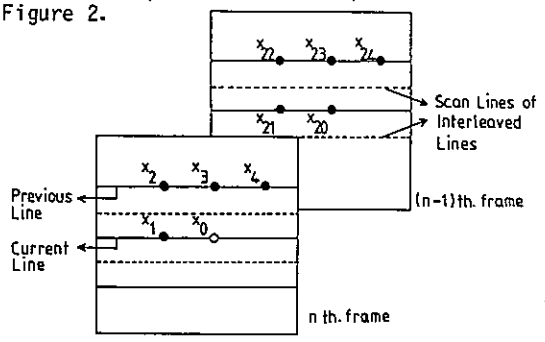


Figure 2

Four different predictions schemes are compared, viz :

(a) - Simple 1-point prediction (Horizontal) :

$$\hat{x}_0 = \tilde{x}_1$$

(b) - Simple 2-point prediction :

$$\hat{x}_0 = 0.5\tilde{x}_1 + 0.5\tilde{x}_3$$

(c) - Optimum 2-point prediction :

$$\hat{x}_0 = A.\tilde{x}_1 + B.\tilde{x}_3$$

(d) - Optimum 3-point prediction :

$$\hat{x}_0 = C.\tilde{x}_1 + B.\tilde{x}_2 + C.\tilde{x}_3$$

The weighting coefficients of the predictors (c)-(d) have been optimized as described in [2]. For 10 different pictures, we have calculated 10 set of optimized coefficients A,B,C, D,E. Their means have been retained as shown in Table 1.

Coef. / Schemes	a_1	a_2	a_3
a	1.0	-	-
b	0.500	0.500	-
c	0.688	0.312	-
d	0.8125	-0.5625	0.750

Table 1

2.3. Results

The DPCM of Figure 1 has been simulated with those four predictors. At this stage of our study, we employ one single quantizer, Q_1, which is given in Table 3. Visual tests, entropy measures [6] and SNR measure [2] are used to compare the four predictors. Entropy defined by

$$\text{Entropy} = \sum_{j}^{-1} p_j \log_2 p_j \quad (2)$$

indicates the minimum bound on the bit-rate, with p_j being the relative frequency of the jth quantization level. SNR is defined by

$$\text{SNR} = 10.\log_{10} \left(V_{max}^2 / \langle (x_i - \tilde{x}_i)^2 \rangle \right) \quad (3)$$

where V_{MAX} = 255. The Huffman codeword length used to calculate the bit-rates are given in Table 7, first column.

The results obtained for 5 different black and white test pictures are shown in Table 2.

Q	Images	Scheme (a)			Scheme (b)			Scheme (c)			Scheme (d)		
		Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR
Q1	YTDF	2.50	3.40	32.92	2.62	3.42	40.30	2.62	3.43	39.00	2.56	2.90	41.40
	YDDLP	3.03	3.70	41.33	2.84	3.35	41.70	2.86	3.40	41.70	2.85	3.28	41.83
	YPORT	2.61	2.60	42.30	2.38	2.62	42.40	2.33	2.54	42.50	2.26	2.36	42.73
	YDHEM	2.38	2.47	42.06	2.49	2.67	43.80	2.36	2.68	44.10	2.23	2.30	44.50
	YFUTU	2.06	2.24	43.71	2.21	2.35	43.92	2.10	2.21	44.20	1.95	2.00	44.70

Table 2

Two important conclusions can be drawn from this table. First, when the prediction schemes (a)-(d) are compared, it was noticed that the 3-point (2-D) prediction schemes drastically reduces the bit-rates. Recall that, a good prediction algorithm stands out by a low average entropy value combined with a small prediction error variance. Second, fixed 15 level quantizer is not sufficient as for the majority of the pictures, the bit-rate exceeds 2.3 bit/pel : especially for those containing high contrast and rapidly varying zones. As a consequence, the quantizer should be adapted to local changes of the pictures.

3. ADAPTIVE QUANTIZER

As mentioned above, a coarse quantizer should be used for rapidly varying zones and a fine quantizer for uniform zones, to avoid granular noise.

3.1. Quantizer description

An excellent survey of quantizer design is given by Pirsch [5]. However, we modify the quantization parameters, with respect to our sampling frequencies and to the fact that VLC does not require the restriction to a very small number of quantization levels as in the case of fixed word length.

In Table 3, non-uniform quantization characteristics, namely Q1, Q2, Q3 are shown.

Q1		Q2		Q3	
Input Range	Output	Input Range	Output	Input Range	Output
≥ 59	67	≥ 100	111	≥ 118	131
45 → 59	52	80 → 100	90	100 → 118	111
33 → 45	39	61 → 80	70	80 → 100	90
22 → 33	28	45 → 61	53	61 → 80	70
13 → 22	17	30 → 45	38	44 → 61	53
7 → 13	10	17 → 30	23	26 → 44	35
2 → 7	5	5 → 17	11	9 → 26	18
-2 → 2	0	-5 → 5	0	-9 → 9	0
-7 → -2	-5	-17 → -5	-11	-26 → -9	-18
-13 → -7	-10	-30 → -17	-23	-44 → -26	-35
-22 → -13	-17	-45 → -30	-38	-61 → -44	-53
-33 → -22	-28	-61 → -45	-53	-80 → -61	-70
-45 → -33	-39	-80 → -61	-70	-100 → -80	-90
-59 → -45	-52	-100 → -85	-90	-118 → -100	-111
≤ -59	-67	≤ 100	-111	≤ -118	-131

Table 3

The selection between the quantizers is done by comparing an activity measure to 2 thresholds, TH1 and TH2 respectively. From the reconstructed pels (see Figure 2), intra-frame activity (ACT1) is calculated over 4 pels :

$$ACT1 = Max\left\{\begin{matrix} |\tilde{x}_1 - \tilde{x}_2| , |\tilde{x}_1 - \tilde{x}_3| , |\tilde{x}_1 - \tilde{x}_4| , \\ |\tilde{x}_2 - \tilde{x}_3| , |\tilde{x}_2 - \tilde{x}_4| , |\tilde{x}_3 - \tilde{x}_4| \end{matrix}\right\} \quad (4)$$

We have taken 2 thresholds TH1, TH2 as being 33 and 74 respectively. The decision logic is as follows, viz :

IF ACT1 $\leqslant$ 33 Q = Q1
IF 33 < ACT1 < 74 Q = Q2
IF 74 $\leqslant$ ACT1 Q = Q3

Figure 3 gives the complete block diagram.

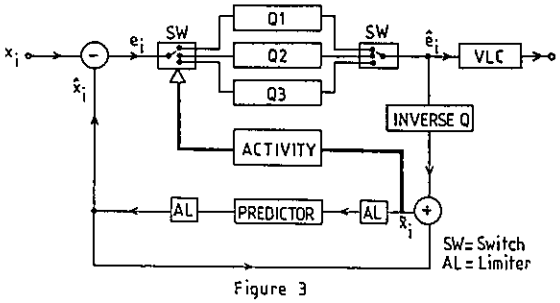


Figure 3

3.2. Results

Table 4 shows the results with an adaptive quantizer. When compared to Table 2, it can be seen that adaptative quantization reduces the entropy. Notice that there is a slight degradation in SNR measures.

Q	Images	Scheme (a)			Scheme (b)			Scheme (c)			Scheme (d)		
		Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR	Entropy	Bit/pel	SNR
Q1	YTDF	2.31	2.63	39.15	2.21	2.41	39.36	2.24	2.42	39.60	2.06	2.12	39.62
Q2	YDDF	2.30	2.40	38.03	2.08	2.15	38.00	2.10	2.18	38.10	2.14	2.20	40.00
Q3	YFDF	2.15	2.23	41.40	2.10	2.22	41.40	2.05	2.15	41.60	2.00	2.06	41.65
Q3	YDFF	2.17	2.21	42.40	2.23	2.36	42.10	2.12	2.19	42.30	2.04	2.10	42.36
	YFDF	1.84	1.91	42.00	1.98	2.03	41.90	1.82	1.88	41.99	1.73	1.80	42.12

Table 4

The picture quality however is not affected. Benefit is taken from masking effects. This obviously yields better results : since both granular noise that often exists in uniform zones and slope overload effect in sharp transitions are drastically minimized. Moreover, three quantizers result in improved edge-business. Figure 4 illustrates the results obtained from 2D-DPCM, the chrominance signals being processed as described in the introduction.
All these results concerning 2-D processing of still pictures must also be verified for picture sequences.

4. DPCM PROCESSING OF PICTURE SEQUENCES

For simulation purposes, we have used two picture sequences digitized by the CCETT in France. The first sequence called "Car Sequence" is a noisy sequence with a moving car. The second called "Girl sequence" is less noisy, with some local movements.

Figure 4

4.1. 2-D DPCM Processing of Picture Sequences

In these tests, the best 3-point predictor (scheme-d) associated with Q1 alone or with Q1,Q2,Q3 quantizers was employed. Table 6 gives the entropies resulting from 2-D DPCM processing. Again, we can see the gain obtained by using adaptive quantizer. The subjective results on the two sequences are not similar. The visual quality of the "Car sequence" is much better than that of the "Girl sequence". This can be attributed to the fact that "the "Car sequence" is much more noisy than the "Girl sequence". Temporel noise was slightly annoying in detailed fixed parts the "Girl sequence". Those remarks led us to investigate a 3-D predictor.

4.2. 3-D DPCM Processing of Picture Sequences

In this case, three predictors are used, viz :

- Intra-field predictor (scheme d)
- Inter-frame predictor : $\hat{x}_0 = \tilde{x}_{20}$
- Hybrid predictor defined by [7]

$$\hat{x}_0 = 0.625.\tilde{x}_{20} + 0.250.\tilde{x}_1 + 0.125.\tilde{x}_2 \quad (5)$$

The switching criteria involves the previous spatial activity ACT1 and a temporal inter-frame activity ACT2, i.e.,

$$ACT2 = Max\left\{\begin{matrix} |\tilde{x}_{21} - \tilde{x}_1| , |\tilde{x}_{22} - \tilde{x}_2| , \\ |\tilde{x}_{23} - \tilde{x}_3| , |\tilde{x}_{24} - \tilde{x}_4| \end{matrix}\right\} \quad (6)$$

These two activities are compared. The predictors are switched in accordance with Table 5. The results are given in Table 6.

For the "Car sequence" we obtain no improvement since the temporal predictor is rarely used because of noise and movement contained in the sequence. On the other hand, for the "Girl sequence" the gain is significant : smaller entropy and no more temporal noise.

Aside from those quantizer and predictor designs, in the next section we investigate the effects of the maximum word length limitation in Huffman codes.

Prediction used for the same pel in the previous frame	ACT1 > 2xACT2	ACT1 < 2xACT2
Temporal	Temporal	Hybrid
Hybrid	Temporal	Spatial
Spatial	Hybrid	Spatial

Table 5

Q	PICTURE SEQUENCES	2D PRED. ENTROPY	2D-3D PRED. ENTROPY
Q1	CAR	3.15	3.15
	GIRL	3.00	2.10
Q1 Q2 Q3	CAR	2.30	2.30
	GIRL	2.30	1.35

Table 6

5. WORD LENGTH LIMITATION EFFECTS

In order to obtain a low transmission rate, the quantized prediction error, $\hat{e}_i$ is coded by VLC. Short code words are assigned to the most probable levels and the longer codes, are assigned to less probable ones. Such a coding method is extremely effective.
The type of VLC employed is the Huffman code [6]. Table 7 shows the probabilities of occurence of each quantization level averaged over 3 quantizers and the code word length, resulting from Huffman coding.

The average bit/pel is defined by

$$b = \sum_{k}^{3} p(Q_k) \sum_{j}^{15} P_{jk}\, l_j \qquad (7)$$

where $p(Q_k)$ is the probability of occurence of each quantizer and P_{jk} is the probability of occurence of each jth. level for the kth. quantizer. In the previous sections, bit-rates resulting from Huffman coding were given.
However, when this coding is associated with an actual TV codec, maximum word length, $|L_i|$ is amounted to 14 as seen in Table 7. In practice, the word length is to be limited from the viewpoint of hardware configuration. For such a goal we adapted the modified Huffman algorithm [8]. In Table 7, code word lengths

N_Q	Probabilities	$\|L_i\|_{max}=14$	$\|L_i\|_{max}=12$	$\|L_i\|_{max}=10$	$\|L_i\|_{max}=8$
1	0.00006	14	12	10	8
2	0.00010	13	12	10	8
3	0.00035	11	11	10	8
4	0.00142	9	9	10	8
5	0.00691	7	7	7	7
6	0.07466	4	4	4	4
7	0.32040	2	2	2	2
8	0.42560	1	1	1	1
9	0.12093	3	3	3	3
10	0.02805	5	5	5	6
11	0.01714	6	6	6	7
12	0.00342	8	8	10	8
13	0.00072	10	11	10	8
14	0.00018	12	12	10	8
15	0.00006	14	12	10	8
	Entropy	2.051	2.051	2.051	2.051
	Bit/pel	2.076	2.076	2.082	2.115

Table 7

limited to 8,10,12 bits are given. Notice that the advantage of using such an algorithm is that it sacrifies a negligible amount of bit/pel for hardware simplicity.

6. CONCLUSIONS

We have simulated an ADPCM codec employing three different quantizers and various intra-frame/inter-frame predictors for the transmission of images at 30 Mbit/s. When ADPCM is associated with VLC we have shown that first, optimizing prediction coefficients with respect to minumum mean-square prediction error criteria significantly decreases the entropy. Secondly, an adaptive quantizer that copes with the local variations in the picture yields reduction in bit-rate. Further, the superiority of temporal (3-D) prediction scheme was demonstrated. Finaliy, the modified Huffman algorithm was examined as it simplifies the hardware realization.

The results show that the proposed codec with statistical coding at 30 Mbit/s with [4:1:0] sampling ratio can be a good candidate for the video transmission over the European hierarchical level of 34.368 Mbit/s.

REFERENCES

[1] Kretz, F., Ann. Télécom., 37, No.7-8, 1982, pp.1/26-25/26.
[2] O'Neal, J.B., BSTJ, May-June 1966, pp.689-7921.
[3] Grallert, M.J., and Tengler, W., Siemens Forsch Entwickl, No.3, Springer-Verlag, 1984, pp.95-99.
[4] CCIR, Recommen.601, 1982.
[5] Pirsch, P., IEEE Trans. Vol. COM-29, No.7, July 1981, pp.990-1000.
[6] Buley, M., and Stenger, L., IEEE Proc., Vol.73, No.4, April 1985, pp.765-772.
[7] Westerkamp, D., Proc.of Int-Conf. in Image, UK, July 1982, pp.184-187.
[8] Murakami, M., et al., IEEE Trans., VOL-COM-32, No.10, October 1984, pp.1157-1159.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

L.7

HYBRID MOTION ESTIMATION IN SUCCESSIVE TELEVISION PICTURES

Mékano MIJIYAWA

Laboratoires d'Electronique et de Physique appliquée*,
3, avenue Descartes,
94451 Limeil-Brévannes Cedex, France

We present a hybrid displacement estimation of moving objects from successive frames in a television scene. The algorithm uses both Kalman filter (KF) formulation and block matching (BM) for television images motion vector estimation, and chooses the method leading to a minimum motion compensated prediction error.

1. INTRODUCTION

The existence of redundancy in television signal has long been noticed. Interframe television coding systems use this characteristic to offer a very high coding efficiency for still pictures. However, the human visual perception of image degradation is not much reduced during simple movements of the object such those encountered when the camera moves. This interframe coding leads to a lower quality for moving pictures. It was shown e.g. [4] that one could achieve high efficiency and a large level of data compression with interframe coding if the motion of the various objects in the successive frames is known. The compression technique using this knowledge is called motion compensation. In order to achieve this goal one has to find accurate motion estimators. The displacement of picture element is considered as a two dimensional vector D(dx,dy), where dx designates the displacement in horizontal direction and dy, in vertical direction.

There are two basic methods to estimate D :

- "Pel recursive" algorithms which estimate motion vector pixel by pixel. One can minimize recursively a motion compensated prediction error by following objects's trajectory, e.g. [5], [7].
- Algorithms which operate on the blocks. The various objects of a picture are classified as blocks of elements, then the pictures are subdivided into blocks with fixed size. One must search for a motion vector of all the elements of the block, e.g. [1].

The first processes, although effective for complex motion, presents some difficulties relative to the choice of some parameters. This could lead to the inaccurate estimations and even divergence of the techniques. The second ones are more efficient with tilting or panning movement. But there are some problems on blocks with differently displaced picture elements since the same motion vector is applied to the all moving pixels of one block.

Up to now, authors, e.g. [2], used displacement estimators that sometimes lead to inaccurate values for D. Our work is based on the recent surveys reported in [4], [6], [7]. Indeed a process used by J. Stuller et al utilizes the first technique with a Kalman filter formulation. We improved this estimator by changing the model in the KF during the motion estimation at each pel. Nevertheless, inaccurate estimations may still occur, especially on the objects edges. The system proposed by Y. Ninomya and O. Yoshimichi and J. Jain, utilizes the second technique with picture block matching. Unfortunately this method may fail when it is applied to the areas of the picture where several matched blocks can be obtained. Moreover a complex implementation is needed [6], in order to estimate a non integer displacement. Consequently, to benefit from the respective advantages of both Kalman filtering and block matching, we propose to jointly use the two methods to get a good estimator. This process could be applied, for example, in bit rate or bandwidth reduction techniques needed for television picture transmission.

In section 2 we present our hybrid motion estimation. First, we recall the block matching principle. Then we describe motion estimation via Kalman filter algorithm and our modification. Section 3 contains experimental results and section 4 the conclusions of our study.

2. HYBRID MOTION ESTIMATION

Let I(X,t-T) and I(X,t) denote the respective luminances of two successive frames as a function of spatial location X and time t. The parameter T is the frame period. Object motion results in a frame to frame displacement of the luminance of picture element. In this section we describe the hybrid process for estimating this motion.

a. Block matching estimator : (BM)

We report here a simple version of the block matching algorithm. The principle is summarized by Figure 1.

*Laboratoires d'Electronique et de Physique appliquée - A member of the Philips Research Organization.

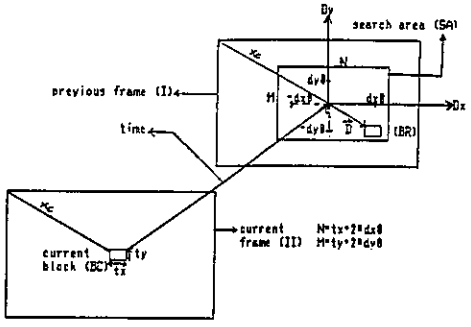


Figure 1 : simple block matching process

The spatial location of (BR) in the (O, Dx, Dy) system of co-ordinates gives directly motion vector with integer components. Notice that this process depends on the search area (SA) and the size of the blocks.

b. Motion estimation via Kalman filter : (KF)

As we mentioned in the previous section, this process is a "pel recursive" technique. Let i denotes the current picture element. In general, the motion estimation could be recursively achieved as follows :

$$DE(i) = DP(i) + Delta(i) \qquad (1)$$

DE(i) is the (2*1) displacement vector estimation, DP(i) its prediction, Delta(i) is the correction term.

In order to estimate the motion vector D via the Kalman filter (KF), one considers D [7] as the state vector. We model this vector as a discrete autoregressive process (AR). This model is quite suitable for non-uniform pel to pel motion. The principle of this algorithm is based on this model and the linearization of the frame difference (FD) which we consider as the observation. Indeed, in most cases, the relation between the FD and the displacement vector D is not linear. We consider [7] the image data I(.,.), as noisy samples and the motion model [5], [7] as below :

$$I(X,\ t-T) = O(X,\ t-T) + n(X,\ t-T) \qquad (2)$$

$$O(X,t) \quad = O(X-D),\ t-T) \qquad (3)$$

where X = *(x,y), D is the displacement vector of the object O(X,t) during a frame period T, X the spatial location of the pixel i considered and (*) denotes the transposition operator. The process n(.,.) is assumed to be a white random sequence with variance Vn. Then the frame difference is defined as :

$$FD(X) = I(X,t) - I(X,\ t-T) \qquad (4)$$

This equation of the FD can be written for small D, by Taylor's expansion about X as :

$$FD(X) = -*G(i).D + \text{higher order in } D$$

where G(i) is the gradient vector of O(X, t-T) with respect to X. This expansion decomposes the frame difference in two parts ; one is linear, the other non-linear (NL). One can show that this NL part depends on the square of spatial frequency of picture, the noise n(.,.) and the square of the motion prediction error. Thus a low pass prefiltering is used in order to achieve the FD's linearization and to reduce the magnitude of NL components. Then we obtain a linear expression of the Kalman filter measurement

$$FD(X) = -*G(i).D + N(i) \qquad (5)$$

N(i) is the measurement error which is taken, for use of the KF, as the measurement noise.

We can now write all the equations of the discrete Kalman filter. For any pixel we have :

- system :

$$D(i) = PHI.D(i-1) + W(i-1) \qquad (6)$$

where PHI is a known transition (2*2) matrix and W(i-1) is a white random sequence. This relation comes from the assumption on the state vector model (AR process).

- a scalar measurement :

$$FD(i) = H(i).D(i) + N(i) \qquad (7)$$

H(i) is an "observation" (1*2) matrix and N(i) the measurement noise as mentioned before. We assume that the covariances COVW(i) = E(W(i).*W(i)) and COVN(i)=E(N(i).*N(i)) are known and E(N(i))=E(W(i))=E(N(i).*W(i))=0. Equations (6) and (7) describe the model and have the same interpretation as in [7]. We refer to these equations of the model to establish the "pel recursive" motion estimator as in equation (1).

- prediction after (i-1) estimate :

. a (2*1) state vector :
$$DP(i) = PHI.DE(i-1) \qquad (8)$$

. a scalar measurement :
$$FDP(i) = H(i).DP(i) \qquad (9)$$

From equation (5) it is easily verify that H(i) = -*G(i).

. state error covariance : this (2*2) matrix is updated as :
$$VP(i) = PHI.VE(i-1).*PHI + COVW(i-1) \qquad (10)$$

. innovation : it is a scalar value defined as below :
$$INNO(i) = FD(i) - FDP(i) \qquad (11)$$

It is easy to show that the innovation under certain assumptions is equal to the displaced frame difference defined as :
$$dfd(X,D(i)) = I(X,t)-I(X-D(i),t-T) \qquad (12)$$

- filtering :

. state vector : as in equation (1) the (2*1) vector DE(i) determines the displacement estimation at pixel i and is function to the prediction DP(i) (equation 8).
$$DE(i)=PHI.DE(i-1)+K(i).INNO(i) \qquad (13)$$

K(i) is a (2*1) vector and determines the Kalman gain for which explanation is given below.

. Kalman gain :
$$K(i) = VP(i).*H(i).SIGMA \qquad (14)$$

with SIGMA = inverse(H(i).VP(i).*H(i)+COVN(i))
K(i) has two components and is conversely proportional to COVN(i) in order to avoid much of the noise which would influence the filtering.

. filtering error covariance : this (2*2) matrix is function of K(i), H(i) and VP(i).

$$VE(i) = (I-K(i).H(i)).VP(i). \qquad (15)$$

where I is the (2*2) identity matrix. The filtering error is equal to the difference of the vectors DE(i) and D(i). The initial conditions concern the first values of DE(.) and VE(.).

In order to improve this algorithm at each pel, we propose to reiterate the process as long as the innovation is greater than a certain threshold "TH", by changing the transition matrix after the first step of reiteration. Unfortunately, this could yield inaccurate estimations or divergence when speed changes occurs and the choice of the noises covariances is wrong.

c.Hybrid motion estimation : (HME)

This algorithm was used in order to avoid the defects of the KF and BM systems. Indeed it was found in certain cases that the KF fails where BM estimates successfully the motion vector and also does conversely. Figure 2 shows a block diagram of the HME.

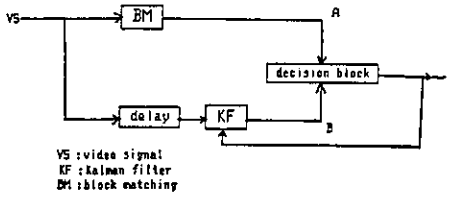


Figure 2 : schematic block diagram of the HME estimator

More precisely with the assumption that the KF and BM have complementary and respective advantages :

- step 1 : for all blocks of the current picture : use BM to determine the motion vectors.
- step 2 : for each pel along the scan line :
  . use the KF to estimate the displacement vector ;
  . comparing the motion compensation prediction error dfd(.,.) given with BM (dfdBM) and the KF (dfdKF), decide which is the best current pel's motion vector estimation. (Figures 3 and 4).

This vector is used in the KF recursive technique starting with the next pixel. The decision block is described by figures 3 and 4.

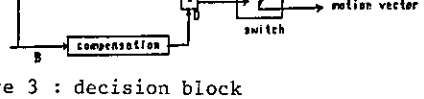


Figure 3 : decision block

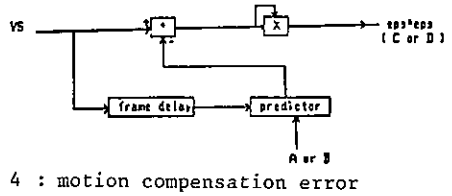


Figure 4 : motion compensation error

3. EXPERIMENTAL RESULTS AND CONCLUSIONS

a. Specifications of the parameters used

In order to evaluate the performance of each motion estimator, we compare actual and compensated pictures. The compensated picture is computed from the preceding frame with the following basic segmentation. Three classes of pictures' areas are defined : 1) moving areas with compensated elements, 2) moving areas with "uncompensated" elements, 3) "background" with fixed elements. The first and the second areas are determined by a movement detection algorithm [3] and [5]. This detection operates on the low pass prefiltered data in order to limit noises. The classes 1 and 2 are separated after pixel X compensation considering the motion compensated prediction error (dfd). If the dfd is greater than the frame difference (fd) then we consider pel X as a new object element and it is predicted by using its neighbouring pixels in the previous picture. The pel where motion is not detected are considered as the third class elements. In this case, a simple copy is needed. The computation of the innovation in the Kalman filter and in the motion prediction error during the compensation, were achieved by a standard two-dimensional linear interpolation. The initial displacement DE(0) in the KF is taken equal to its prediction computed from the preceding lines'motion estimation, at each scan line except the first one where it is equal to 0.

The Kalman filter gain depends on the COW(i) and COVN(i) which are a priori unknown. For this purpose, one can use an algorithm of estimation of COVW(i) and COVN(i). But this estimation leads to a complex implementation. In order to simplify the computation we assume that PHI, COVW(i) are diagonal such that for any picture element i, PHI=phi.I;COVW(i)=cw0.I and COVN(i)=cn0=sum(fd0(i,j).*fd0(i,j))/M1.M2 with fd0(i,j)=h(FD(i,j))-FD(i,j)) and i=1,M1;j=1,M2. The function h(.) defined the low pass filter. M1=675, and M2=536 define the pictures'size. As for cw0, it is computed from equation (6). If we assume that the state vector D(i) has a known covariance COVD independent of i, such that COVD=cd0.I, then cw0=cd0(1-phi.phi). The block matching used is based on the one developed by Y. Ninomiya [7] utilizing an iterative method. The summary is given by figure 5. The displacement estimation is obtained after n stages. The test scenes consist of the real television frames (2:1 interlaced fields) of M1.M2 samples each, obtained 25 times a second. The "children" test scene was chosen because of various movements of objects. The "Baltimore" scene is a zooming building. The "Secretary" contains head-and-shoulders view of a woman smiling and moving her head up and down, different parts of the scene move differently. The "Car" is very noise scene containing an opening gate with opposite translational movements and a moving car with various speed.

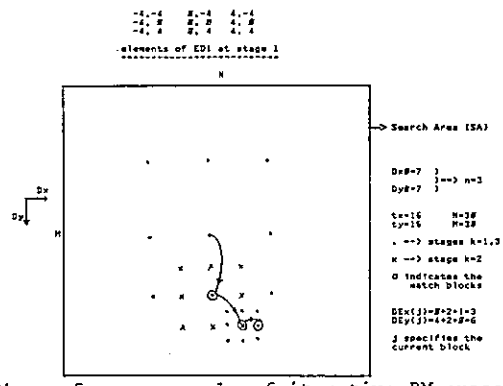


Figure 5 : an example of iterative BM processing

b. Results of simulation and discussion

The four sets of image test sequences were used to compare the KF, BM and HME algorithms. The criteria used are : 1) the root mean square (rms) of the dfd corresponding to one image, 2) the entropy of the dfd, 3) the compensation ratio which measures the number of the compensated picture elements in the moving area. The moving area elements ratio in one picture was called "activity". From table A notice that the improvement on the entropy is within (15 %-30 %) and the rms within (10 %-32 %) when the HME algorithm is used instead of BM or the KF.

test picture	car			children			baltimore			secretary		
algorithm	1	2	3	1	2	3	1	2	3	1	2	3
activity (%)	46			28			39			29		
cr (%)	89	83	86	84	86	84	78	78	76	89	83	98
rms (dfd)	18.1	12.1	8.89	9.88	9.48	7.88	6.26	6.36	5.6	4.98	5.88	3.48
entropy (dfd)	3.94	4.12	3.88	1.81	1.38	0.8	1.57	1.1	0.61	0.91	1.8	0.78

1 --> BM algorithm
2 --> KF "
3 --> HME "

table A : simulations' results

The distributions of the dfd are plotted for the test "car" scene and for each of the three (BM, KF, HME) methods. From figure 6 notices the effects of the hybrid motion estimator (HME) at a significant line 150. Indeed one can see that when used separately, both BM and the KF performed rather poorly, while the HME prediction errors' are quite small. The parameter cd0 was set to 1 and the threshold TH (see section 2.b) was taken to be the fifth part of the mean frame difference of the pictures. The transition matrix parameter phi was chosen equal to 0.95 at the first iteration and 1 after. We found that, with the sequences described above, in the HME process the motion vector given by the KF was chosen 91.9 times in average, out of 100, but the dfd is not much reduced when the KF was used alone. It means that the KF failed only 8.1 times and needed to be reinitialized. We found that the time of convergence of the KF is approximately equal in average to 5 with noiseless synthetical pattern used by Netravali (see [5] in section 2.3).

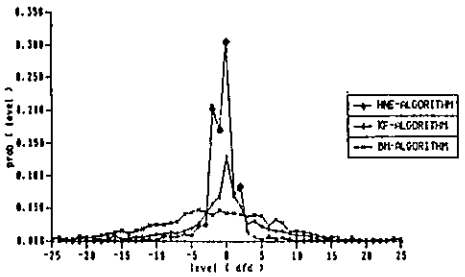


Figure 6 : performance of BM, KF, HME algorithms

4. SUMMARY AND CONCLUSIONS

This paper has presented a hybrid motion estimator (HME) which employs both Kalman filter (KF) formulation and block matching (BM). The improvement of the variance of the motion compensated prediction error was proven. This new motion estimator (HME) can be applied to motion compensated interframe hybrid coders for television bit rate reduction and in all applications where object's displacement measurement is needed.

5. REFERENCES

[1] Giorda, F. and Racciu, A., IEEE Trans. on Com., Sept. 1985, pp. 1002-1004.

[2] Huang, T.S.(ed.), Image sequence processing and dynamic scene analysis (NATO ASI series F, computer and systems sciences n° 2, Springer-Verlag, Berlin, 1983).

[3] Trump, M.R., Movement detection, UK Patent Application GB 2 031 686 A, published Sept. 14, 1980.

[4] Jain, J.R. and Jain, A.K., IEEE Trans. on Com., vol. COM-29, n° 12, Dec. 1981, pp. 1799-1808.

[5] Netravali, A.N. and Robbins, J., BSTJ, vol. 58, n° 3, March 1979, pp. 631-670.

[6] Ninomya, Y. and Othuka, Y., IEEE Trans. on Com., vol. COM-30, n° 1, Jan. 1982, pp. 201-211.

[7] Stuller, J. and Krishnamurphy, G., Computer Vision Graphics and Image Processing, vol. 21, 1983, pp. 169-204.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

NONLINEAR PICTURE ENHANCEMENT TECHNIQES FOR VERTICALLY INTERPOLATED TV-SIGNALS

H. Schröder, H. Elsler, M. Fritsch

University of Dortmund, FRG

In this paper a new concept for nonlinear vertical edge sharpening is introduced. This concept is basing on vertical pre- and postfiltering by which an error-free picture scanning and a flat field reproduction can be achieved. Due to the vertical interpolation process at the receiver vertical enhancement techniques become more efficient and a very high picture sharpness can be achieved. The described nonlinear edge sharpening can favourably be achieved replacing the original edge by a synthetical one which is optimum with respect to the channel including the visual system.

1 Introduction

Nonlinear vertical edge sharpening (crispening) techniques for standard television signals are strongly restricted because of the interlaced scanning scheme. Due to this scanning scheme several deficiencies such as aliasing, line structure distortion and line flicker occur.

In this paper a new concept for nonlinear vertical edge sharpening is introduced, basing on the vertical pre- and postfiltering approach described in /1/, /2/. It will be shown that the restrictions to enhance standard television signals in vertical direction can be overcome and very high picture sharpness can be achieved.

2 Processing of standard interlaced TV-signals

Nonlinear vertical picture processing should be discussed together with picture scanning. Therefore, a short description of the interlaced scanning process is given. Fig. 1a shows the video transmission system from the picture to be transmitted to the receiver including the processing of the viewer's eye. The picture with its three-dimensional brightness function $s(x,y,t)$, is filtered in the camera with a transfer function describing the influence of the optical system and the scanning beam. Subsequently the resulting signal is sampled in the camera with line interlace scanning (Fig. 1b). The complete spatio-temporal reconstruction filtering is performed by the monitor and the viewer's eye. Hereby the monitor is processing only spatially, whereas the processing of the eye is spatio-temporal.

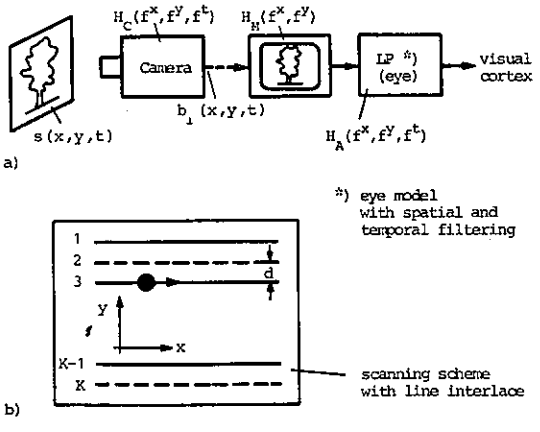


Fig. 1 Video Scanning and transmission

With these assumptions the scanning process can be described analytically /3/. The sampled output signal of the camera is $b_\perp(x,y,t)$. By Fourier transforming $b_\perp(x,a,t)$ twodimensionally into the spatial frequency domain we get for both fields two time dependent spectra $B_\perp(f^x,f^y,t)$ which have to be superposed /3/:

$$B_{i\perp}(f^x,f^y,t)=(1/(2d))\cdot\sum_{m=-\infty}^{\infty} B(f^x,f^y-m/(2d),t_i) \quad (1)$$

$$B_{k\perp}(f^x,f^y,t)1/(2d)\sum_{n=-\infty}^{\infty} B(f^x,f^y-n/(2d),t_k)(-1)^n \quad (2)$$

Fig. 2 shows for two different points of time t_i, t_k the result by a one-dimensional section at $f^x=0$. The repetition frequency is given by the line distance 2d of one field. Heavy field aliasing i. e. spectral overlapping of adjacent spectra and a certain amount of frame aliasing

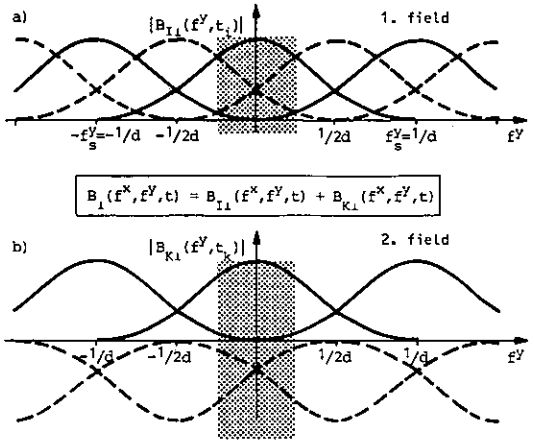


Fig. 2 Interlaced field spectra

i.e. overlapping of spectra periodic with $f_S^y=1/d$ can be seen. The sign of the periodic spectra $B_{k,}(f^x,f^y,t)$ alternate with vertical frequency $1/(2d)$. Therefore, a compensation of the field aliasing seems to be possible by integrating both series by the viewer. This would be true, if both spectra would be perceived with the same amplitude (exactly) and with an perfect integration. Temporal filtering of the eye, however, results in very different perceived amplitudes of the fields at a point of time t_D due to the different times "flashing" the different fields on the monitor /4/.

The result is heavy field aliasing especially for high vertical frequencies. As the sign of the spectra to be compensated change from field to field the field aliasing flickers with 25 Hz, which is very disturbing, as it is well known. From this we can conclude, that processing of interlaced signals has some disadvantages:

(1) Linear frequency response enhancement based on simple field processing (what means taking only line delay elements) has only a small range of influence which is periodic with $1/(2d)$, see dotted area in Fig. 2 for the basic range. Such a processing, is very limited with respect to edge sharpening.
(2) Nonlinear enhancement suffers strongly from interlaced scanning. Edge enhancing always intensifies field aliasing: heavy line flicker and moire distortion occur. The latter is particularly disturbing as it is appearing as a 25 Hz area flicker.

From this we can conclude that the effect of nonlinear vertical edge enhancement is very limited for standard interlaced television signals. In many cases it is even unacceptable because of additional flicker.

3 Processing of vertically interpolated signals

A concept has been introduced /1/, /2/, /3/ of pre- and postfiltering, which is shown in Fig.3 Beginning with a high line number camera the picture signal is sampled with e. g. doubled

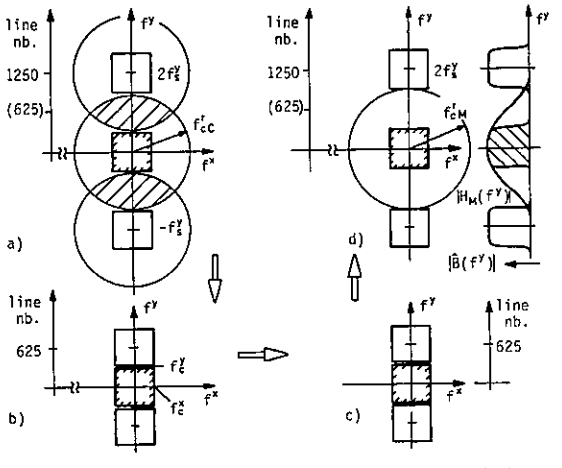


Fig. 3 Vertical pre- and postfiltering /1/

sampling frequency $2\,f_S^y$. Through this the repeated spectra get a larger distance and there is much space for aliasing - which is not within the area to be transmitted.

By vertical and horizontal bandlimitation to f_c^x and f_c^y spaces are generated, which are free of aliasing (Fig. 3a). After a scan conversion to half the line number ($f_S^y=2f_c^y$) the signal is transmitted compatibly (Fig. 3b). In the receiver an interpolation filter is implemented, which converts the signal to the doubled line number again (Fig. 3d). The monitor with its transfer function then reproduces the interpolated television signal i. e. a signal with spectral gaps.

Several advantages of this pre- and postfiltering approach can be seen e. g.:

(1) No aliasing errors, respectively a defined amount of aliasing is transmitted.
(2) The interpolating filter in the receiver enables a reproduction with flat field or a defined amount of line structure.
(3) The product of the transfer function of the pre- and postfilter can be linearly optimized with respect to the camera's and the monitor's frequency reponse.
(4) Spectral gaps enable smart nonlinear enhancement techniques. This will be discussed in detail below.
(6) A line flicker-free reproduction can be achieved although the high line number monitor may operate in an interlaced mode.

For an interpolated TV-signal two reproduction modes can be applied: (1) interlaced or (2) progressive mode:

(1) For an interlaced reproduction the television signal is converted from 1250, 1:1 to 1249, 2:1 /4/. Then two fields with doubled line number are available. No spectral overlapping within the field spectra occurs. Only a small amount of residual line structure and line crawl is visible /5/.
(2) For a progressive reproduction the interpolated signal can be reproduced directly with 1250 lines, 1:1 without 25 Hz flicker.

Nonlinear edge enhancement always mean spectral expansion in the sense that the local spectrum of an edge is expanded corresponding to sharpening the edge. For an interpolated signal there are gaps for such a spectral expansion, which is not the case for a standard TV-signal.

However, filling the gaps enables more picture sharpness (not more resolution) - but also more line structure and in the case of interlaced reproduction more line flicker. Therefore, an efficient nonlinear processing should tend to enhance sharpness and to limit line flicker below a certain nondisturbing level. Very important is the choice of the reproduction mode. The advantages of high line number progressive reproduction are manifest. An interlaced reproduction, however, requires a monitor with only

half the line frequency. For this more economic solution it is particularly important to limit spectral expansion because of line flicker distortion, which occurs when spectral components at f_s^y are generated.

4 Vertical edge enhancement algorithm

Two important concepts for edge enhancement algorithms are well known:
- adding of detail signals to sharpen the edge /6/, /7/, /8/, /9/
- changing the original slope by a synthetical one /8/, /9/.

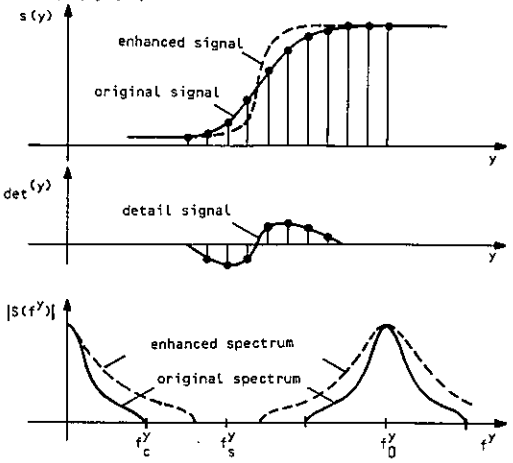


Fig. 4 Nonlinear vertical edge enhancement by adding a detail signal

Fig. 4 shows the first method adding a detail signal to the original signal which is vertically interpolated. The original edge spectrum is expanded due to the enhanced slope. There are some advantages of this concept e. g.:
- low visibility of busy edges if a detail signal is generated, which is proportional to the edge amplitude ("soft decision")
- natural picture impression because of the described proportionality
- simple implementation

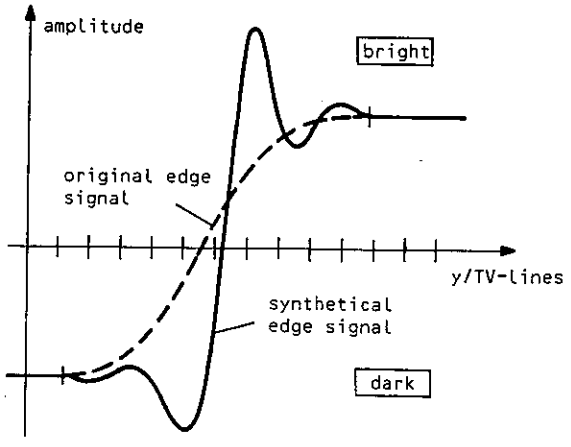


Fig. 5 Synthetical edge signal technique

On the other hand a synthetical edge signal which is reproduced instead of the original edge signal (Fig. 5) offers some other important properties:
- no direct noise addition as it is the case for a noisy detail signal
- however, tendency to busy edges due to misdetection of edges, which makes necessary soft decision
- direct control of spectral expansion
- possibility of an optimum edge shape taking into account nonlinear and linear properties of the complete transmission system including picture scanning in camera and monitor and the properties of the viewer's eye as well.

The latter is very important. It is possible to approximate at least important elements of a picture (i. e. edges, lines, etc.) in a nonlinear sense by this getting optimum signal shape respectively picture quality for those elements. A similar approach for coding has been introduced in /10/.

To get a simple solution for a soft decision a nonlinear enhancement concept has been investigated which is based on a synthetical detail signal added to the original signal. By this, advantages of both methods described above are combined. This concept is shown in Fig. 6, /11/.

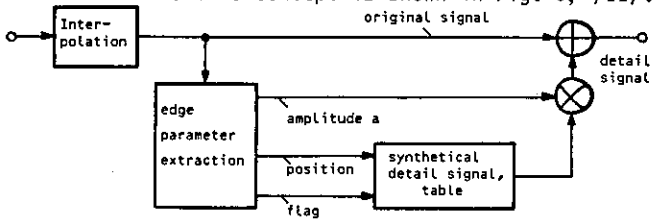


Fig. 6 System for synthetical detail signal

A synthetical detail signal with standardized amplitude is generated by table-look up, weighted by the detail amplitude and added to the original edge signal. Edge parameters are extracted by an edge parameter circuit generating the detail amplitude a, the edge position and an edge recognition flag. To get a simple soft decision algorithm the detail signal is weighted with an appropriate detail amplitude. Therefore, both, soft edges and low amplitude edges will be weighted with a corresponding small detail amplitude a: misdetection of edges then is not visible. In addition a threshold /12/ is implemented for the detail amplitude shown in Fig. 7 to get high stability for noisy signals.

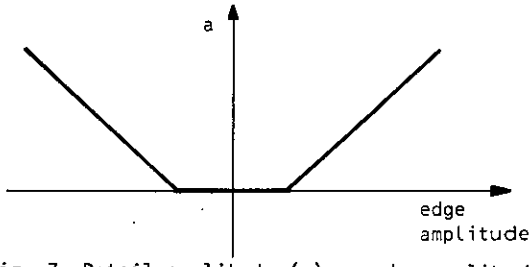


Fig. 7 Detail amplitude (a) on edge amplitude

To get a synthetical edge signal which is optimum with respect to the transmission system linear and nonlinear components have to be taken into account (Fig. 8). For this nonlinear system the following properties hold:
- γ^{-1} pre-correction in the camera does not perfectly compensate monitor's γ due to the bandlimitation by the different filter processes on the channel.
- γ^{-1} in the camera, γ of the monitor and the logarithmic characteristic of the eye (Weber-Fechner law) generate edge signals, which are unsymmetric with respect to echoes. Depending on the steepness of an edge slope the edge position /11/ ist shifted.

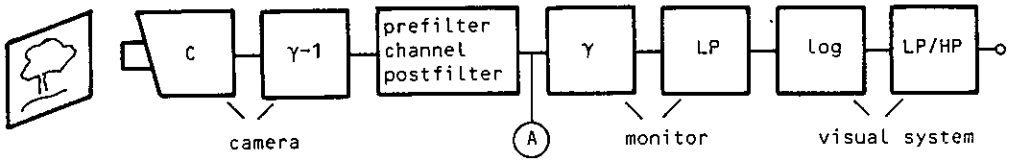


Fig. 8 Video transmission system

For the synthetical edge signal these effects can be taken into account by calculating this signal for position A (Fig. 8) on the basis of a given optimum edge slope at the output of the complete system (Fig. 8) /11/, /13/. Taking into account particularly effects of γ^{-1}, γ is advantageous with respect to line flicker /9/.

To get an optimum edge shape for simplicity a linear lowpass characteristic has been assumed for the complete transmission system including the viewer's eye. The approximated result is shown in Fig. 5. There is a higher positive overload /14/ - due to the smaller sensitivity of the eye for brighter (Weber-Fechner law) and there is a small position difference to the original signal. The latter is a precorrection of nonlinear processing.

The synthetical detail signal now is calculated by taking the difference between the optimum edge shape and the original edge signal, which is assumed as the step response of the pre- and postfilter. By this technique for steep edges nonlinear enhancement yields really optimum synthetical edges whereas for softer edges there is an influence of the original signal.

5 Conclusion

A new concept for nonlinear edge sharpening has been described, which is basing on vertical pre- and postfiltering respectively an interpolated receiver signal. First results are

(1) Picture quality of a pre- and postfiltered television signal can be improved highly with respect to picture sharpness. It can be seen, that the typical somewhat soft impression of vertically pre- and postfiltered television signals /15/ can be improved by edge enhancement techniques. Previous results of picture quality comparison tests showing nearly equal results for progressive scan and vertically pre- and postfiltered signals, have to be completed.

(2) For an interlaced high line number reproduction spectral gaps can be filled up to about 80 % of the gaps without disturbing line flicker in natural scenes. In connection with a pre-filter having a compromise frequency response of about 35 % modulation depth at the Nyquist frequency a very good resolution and sharpness can be achieved.

(3) Picture sharpness and line flicker for an interlaced reproduction are adjustable by the choice of the synthetical edge shape. Both, line flicker and sharpness, then can have the same amount as a real high line number system has with the same reproduction mode. Moirē-distortion, however, at periodic structures can be avoided by a smart edge extraction. Resolution of fine details e.g. in texture areas remain twice for HDTV-systems with twice the line number.

Acknowledgement

The authors would like to thank the "Deutsche Forschungsgemeinschaft" for supporting this work. The authors are very thankful to Prof. Wendland for stimulating this work and helpful discussions and to Mr.Becker for his support/11/.

/1/ Wendland, B.: High Definition Television Studies on compatible Basis with Present Standards. "Television Technology in the 80's",pp. 124-131, SMPTE, Scarsdale, New York, 1981.
/2/ Schröder, H.; Eisler, H.: Planare Vor- und Nachfilterung für Fernsehsignale. ntz-Archiv, vol. 4(82), No.10,pp.303-312.
/3/ Wendland, B.: Zur Theorie der Bildabtastung. ntz-Archiv, vol. 4, No. 10, pp. 293-301.
/4/ Güttner, E.: Uhlenkamp, D.: Verbesserte Wiedergabe von Norm-Fernsehsignalen, ntz-Archiv, vol.4(82), No.10, pp.313-321
/5/ Stollenwerk, F.: Qualitätsvergleich von Zeilensprung- und Vollbildwiedergabe, Frequenz, vol. 37 (1983), No. 11/12.
/6/ Wendt, H.: Verbesserung der Bildschärfe durch einen geschalteten Entzerrer. Fernseh-und Kinotechnik 1971, No. 5, pp. 174-176.
/7/ Zamperoni, P.: Die Verbesserung der Bildqualität bei tiefpaßbegrenzten Fernsehsignalen. Wiss. Ber. AEG-Telefunken 45, No.1/2, pp.36-47.
/8/ Schönfelder, H.: Möglichkeiten der Qualitätsverbesserungen beim heutigen Fernsehsystem. Fernseh- und Kinotechnische Mitteilungen 37 (1983), No. 5, S. 187-196.
(1983), No. 5, S. 187-196
/9/ Jacobsen, H.-M.: Bildschärfeverbesserung mittels digitaler Verarbeitung im PAL-Farbfernsehempfänger. Dissertation, Technische Universität Braunschweig, 1983
/10/ Wendland, B.: Optimale Irrelevanzreduktion durch Codierung synthetischer Quellensignale. NTG-Fachber. Bd. 65, S. 125 ff. Berlin, 1978.
/11/ Becker, P.: Nichtlineare Kantenversteilerung von Videosignalen zur Erhöhung der Bildschärfe in vertikaler Richtung. Diploma thesis at the University of Dortmund
/12/ Okada, K.: A Digital Contour Corrector for a HDTV-Camera, NHK Report 311/85
/13/ Schröder, H.: Improvement of subjective picture sharpness of a bandlimited television channel by linear methods. Siemens Forsch.-, Entw.-Ber., vol. 5 (1976), No.2, pp. 57-60
/14/ Sakata, H.: Picture Quality Compensation System based on Brightness Polarity. SMPTE Journal, Febr. 1985, pp. 190-199
/15/ Wendland, B.; Schröder, H.: On Picture Quality of Some Television Signal Processing Techniques, SMPTE Journal, Oct. 1984.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

THE APPLICATION OF A TRANSLATION INVARIANT TRANSFORM FOR LOW BITRATE VIDEO CODING

R.H.J.M.Plompen *, J.G.P.Groenveld *, J. Biemond ** and F.Booman *

*
Dr. Neher laboratories
Digital Video Communication Group
Dutch Telecom, P.O.Box 421
2260 AK Leidschendam ,The Netherlands

**
Delft University of technology
Information Theory Group
Dept. of Electrical Eng., Mekelweg 4
2628 CD Delft, The Netherlands

ABSTRACT

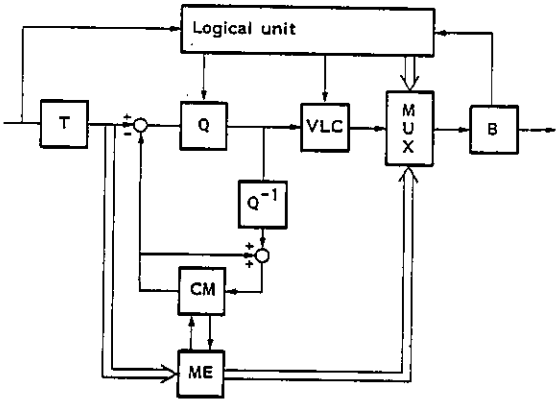
For very low-bitrate coding, it is necessarry to remove intra/interframe redundancy. From the standpoint of visual perception it is desirable to strive after a good quality for still pictures. For a sequence of images smooth motion is even as important as spatial resolution. However for a sequence of images some degradation is acceptable due to the characteristics of the human observer.
In a hybrid coding scheme, i.e. transform coding in combination with an interframe prediction, intraframe aswell as interframe redundancy reduction techniques are applied. The prediction error, the difference between the actual and the coded block, is the orthogonal projection of consecutive frames. The prediction error can be decreased further using compensation along the motion trajectory. The motion trajectory can be calculated using displacement estimation algorithms. In this paper only a block-matching algorithm is considered.
In literature several displacement estimation algorithms have been proposed. Algorithms based on a straight forward block base implementation will minimize the prediction error, known as displaced block difference. They can be characterized as best-match algorithms.
In this paper a transform domain oriented motion estimation is explained. Properties of the translation invariance and frequency weighting of the transform coefficients within the motion estimation are described.
The improvement of the subjective picture quality using the proposed motion compensation is verified using computer simulations.

USED CONFIGURATION

In the hybrid coding scheme the estimation is usually calculated in the pixel domain, whereas the actual coding takes place in the transform domain. This requires at least two transforms i.e. a forward and an inverse transform. The configuration described in this paper is based on only one transform (the discrete cosine) i.e. the incoming image is transformed using square blocks. In the transform domain the problem to overcome is the independent processing of the blocks.

An analysis of the input signal gives a rough estimate for the allowable number of bits to spend. The status is monitored and the status parameters are converted into control parameters. The buffer fullness is further used to compute the stepsize for the quantizer.

Figure 1 shows the hybrid coding scheme used for the simulations experiments. The decoding has been performed by utilizing a similar strategy .



T : Transform
ME : Motion Estimation
LU : Logical unit
VLC : Variable length Coder
Q : Quantiser
B : Buffer
MUX : Multiplexer

figure 1 : hybrid coding scheme

A change detector, CD, calculated in the transform domain, makes a distinction between significantly changed and nonchanged blocks, assuming the processing of a full frame takes place in blocks. The change detector threshold is adapted to the quantizer choice.

The prediction error is decreased further by applying motion compensation.

The motion estimation is calculated in the transform domain and not in the pixel domain. The residuals are quantized and variable length coded (VLC).

A logical unit controls the coder, i.e. the codec is adapted to the non-stationarity of the input signals and buffer state. A rate equalizing buffer has been used because variable length coding and run length coding is used.

The sensitivity of the motion compensation can be enhanced by the introduction of a frequency weighting function. By means of so called pairs of displacement matrices denoted by H_Δ and $H_{\Delta-N}$, translation of an object beyond the block can be estimated within the transform domain. Shaping the displacement matrices according to the visibility function given in [5] results in a better estimation.

INTRODUCTION DISPLACEMENT MATRIX

The displacement operator is explained first in the time domain in order to obtain a better understanding of the manipulations in the transform domain. Let h be a nilpotent operator of index N. To prevent discontinuities (e.g. no gaps between shifted blocks) the nilpotent operators are always used in pairs with indices Δ and $\Delta-N$. Then h_Δ and $h_{\Delta-N}$ have a matrix representation of the form:

$$h_1 = \begin{bmatrix} 0 & I_{N-1} \\ 0 & 0 \end{bmatrix} \qquad h_{1-N} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

The major properties of h are;

$$h_1(\Delta) = h_1^{\Delta} \quad \text{e.g. } h_1(2) = h^2{}_1 = h_2, \tag{2a}$$

$$h_1^N = 0 \ , \tag{2b}$$

$$h_{\Delta-N} = h_N^t{}_{-\Delta} \ , \tag{2c}$$

$$f(q)\, h_x \rightarrow \text{horizontal shift} \quad \begin{matrix} x > 0 \text{ to the right} \\ x < 0 \text{ to the left} \end{matrix} \tag{2d}$$

$$h_y\, f(q) \rightarrow \text{vertical shift} \quad \begin{matrix} y > 0 \text{ up} \\ y < 0 \text{ down} \end{matrix} \tag{2e}$$

Let f(q-1,t-1) , f(q,t-1) and f(q+1,t-1) (figure (2)) be three sub-blocks of size N x N in the previous frame, and assume translation only with D = x and x>0.
Then :

$$f(q,t-1,D) = h_x\, f(q,t-1) + h_{x-N}\, f(q-1,t-1) \tag{3}$$

where f(q,t-1,D) is the compensated translated sub-block q. The formulation shown in (3) means a shift to the right of block q over x positions and a shift to the left of block q-1 over x-N, see figure 2.

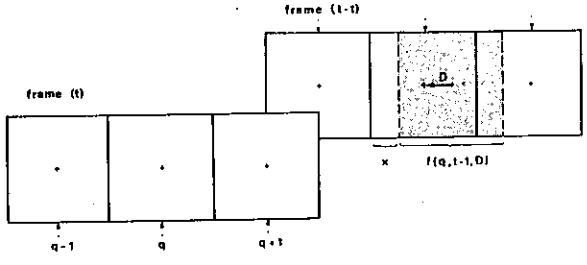


figure 2 neighbouring blocks.

The sum of the pairs of displacement matrices Δ and $\Delta-N$ is taken to get cyclic matrices. Using the new defined operator h^{cl} the blocks q and q-1 become a cylinder and the energy is kept within the block.
The operator h^{cl} rotates the cylinder along its axis. To obtain the displaced sub-block another matrix e(trunc) has to be introduced which has the form;

$$e_N^\Delta = \begin{bmatrix} I_{N-\Delta} & 0 \\ 0 & 0 \end{bmatrix} \tag{4}$$

A combination of both matrices $h_N^{c\Delta}$ and e_N^Δ, results in the same displacement matrix h_Δ

$$h_\Delta = e_N^\Delta \quad h_N^\Delta \tag{5}$$

Substitute (5) in (3)

$$f(q,t-1,D) = f(q,t-1)\, e_N^\Delta\, h_N^{c\Delta} + f(q-1,t-1)\, h_N^{c\Delta}\, e_N^{N-\Delta} \tag{6}$$

Given the properties of the unitary transformation matrix T, the method described can also be used in the transform domain.

$$T(fh) = FH \tag{7}$$

Capital characters denote the calculation in the transform domain i.e. h , f, e and H, F, E. Due to the separability of the transform the displacement matrix becomes:

$$H_\Delta = T_c \quad e_N^\Delta \quad h_N^{c\Delta} \quad T_r^t \qquad (8)$$

with; T_c , T_r operators on columns and rows respectively.

The translation invariant matrix becomes;

$$H_N^{c\,\Delta} = H_\Delta + H_{\Delta - N} \qquad (9)$$

because (7) and (8),

$$H_\Delta = E_N^\Delta \quad H_N^{c\Delta} \qquad (10)$$

The displacement matrix h (and H) is non-singular. The inverse operation does not exist; data is shifted out of the considered sub-block q . The displacement matrix in the pixel domain contains a lot of zeroes, on the other hand the transform domain displacement matrices can be decomposed with a shift-in-place algorithm.

FREQUENCY WEIGHTING

After the transform of a block in frame t denoted by f(q,t) the obtained matrix F(q,t) contains coefficients C(u,v). Two possible frequency weighting methods are to be distinguished:

- additive
- multiplicative

ADDITIVE WEIGHTING

For the change detection additive weighting is used. Each coefficient is compared with a weight $T_a(u,v)$:

$$E[u,v,D] = \left|\Delta F[u,v,t]\right| - T_a(u,v) \qquad (11)$$

With :

$E[\,]$ = the prediction error
$T_a()$ = weighting function

$\Delta F()$ = coefficients under consideration.
= F[q,t] - F[q,t-1,D]

$\Delta F[u,v,t]$ = $\Delta F[q,t]$ indicating coefficients within block

This weighting function is such that all the components have an equal contribution to a decision criterion. Only positive differences are taken into account i.e. $\left|F\{q(u,v)\}\right| > T_a(u,v)$. The displaced block difference DBD is the minimum over the search area. In the case of the brute force search method the global minimum is defined by:

$$DBD = \min_{SW} \left\{ \sum_{u=1}^{N} \sum_{v=1}^{N} E[(u,v),D] \right\}$$

with SW = search area.

In the case $\left| F\{q(u,v)\}\right| < T_a(u,v)$ the weighting function does not influence the error . Then the results obtained in the pixel domain and the transform domain are the same.

MULTIPLICATIVE WEIGHTING

In order to obtain a search algorithm sensitive to predominated structures a weighting function in the transform domain T_a is introduced. The transform coefficients are weighted, this weighting is equivalent to a scaling. The scaling is calculated only in the encoder .

$$E\,[u,v,D] = T_m(u,v)\; F[q(u,v),D)$$

This approach is very interesting due to possible manipulation of the coefficients e.g. filtering. The coefficients itself are shaped i.e. scaled.

For example a lowpass filtering. Let us consider a number of samples f(i) with i = 1,2,..,64. After applying a transform we obtain the coefficients F(i). Coefficients with a low index correspond to low-sequency elements. (The sequency of a transform basis function is defined as the number of zero crossings. It is normally used instead of "frequency" for non-sinusoidal transforms.) Let $T_m(i)$ be the weighting function:

$T_m(i)=1$ i= 1,2,..,32
$T_m(i)=0$ i= 33,.., 64

now the filtered elements become:

$$F^*(i) = T_m(i)\; F(i) \qquad (12)$$

In the case of the described configuration $T_m(i)$ is equivalent to E(trunc). In order to have a smoother filtering $T_m(i)$ has to be shaped. For the estimation and the quantization H_Δ and $H_{\Delta-N}$ become $T_m\; H_\Delta$ and $T_m\; H_{\Delta-N}$,

with T_m as the scaling matrix.

EXPERIMENTS AND RESULTS

In order to compare the performance of the proposed estimation two sequences are used i.e. a splitscreen scene with a hard switch (Split and Trevor)* and a sequence with a girl behaving naturally in front of a camera (Alexis).
The blocksize of the transform and the motion compensation is 8 x 8 pixels. The bitrate for video only is 300 kbit/s. The estimation in the pixel domain is compared to the estimation in the transform domain whereas the mean square error is utilized as opimization criterion.

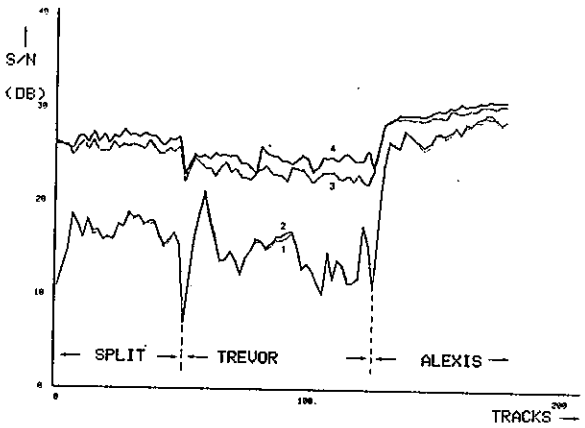


figure 3 results comparison

The odd numbered curves show the result of the estimation in the pixel domain, the even ones with the new transform domain oriented estimation. Curves 1 and 2, represent the frame differences and 3,4 the quantized displaced frame difference. The coding gain in the Alexis part differs from the Trevor part, this is due to the content and the quality of the source material. It is shown that the proposed method results in a higher coding gain for sequences with a low S/N ratio and moderate motion. The advantage of the proposed method is the possibility of implicit filtering. The influence of the noise can be annihilated by applying lowpass filtering with a specific weighting function. The weighting function can be adapted to the characteristic of the human visual system [5].

ACKNOWLEDGEMENT

The assistance of Prof. dr.ir. D.E.Boekee of Delft University of technology and the support of the Dr. Neher laboratories in various aspects of the work reported here is gratefully acknowledged.

* The test sequences are provided by the British Telecom Research Laboratories on behalf of the COST 211-bis project.

REFERENCES

[1] Alireza F. Faryar. Sarah A. Rajala
SUBSAMPLING AND INTERPOLATION IN VIDEO SEQUENCE CODING.
Video Teleconferencing Algorithm CCSPTR-85/16
Center for communication and processing

[2] R.H.J.M. Plompen, B.F. Schuurink and J. Biemond
A NEW MOTION-COMPENSATED TRANSFORM CODING SCHEME
Proceedings ICASSP 85 International conference
IEEE Acoustics, Speech and Signal Processing
March 1985 Vol 1, pp. 371-374

[3] J.A. Stuller and A.N. Netravali
Transform domain estimation
BSTJ vol 58 3 march 1979 pp 619-688

[4] D.F. Elliot and Rao K.M.
FAST TRANSFORMS ALGORITHMS ANALYSIS, APPLICATION
Academic Press 1984

[5] James L. Mannos David J. Sakrison
THE EFFECT OF A VISUAL FIDELITY CRITERION ON THE ENCODING OF IMAGES.
IEEE Trans. on Information theory,
Vol IT-20 no. 4, JULY 1974

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOME REVERSIBLE IMAGE OPERATORS FROM THE POINT OF VIEW OF CELLULAR AUTOMATA

P. Zamperoni
Institut für Nachrichtentechnik, Technische Universität Braunschweig (FRG)

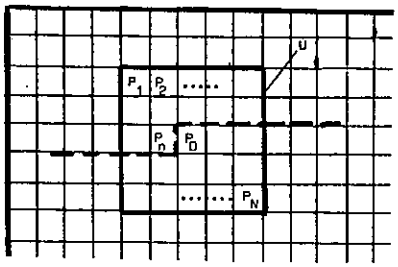
1. REVERSIBILITY CONSIDERATIONS AND REVERSIBLE OPERATOR STRUCTURES

Local image operators and cellular automata can be considered under several aspects in an unified look and described by means of a transition function:

$$P_0 \longrightarrow Q_0 = f(P_0 , P_1 \ldots P_N) = f(U) \qquad (1)$$

giving the new state Q_0 as a function of the old states $P_0 \ldots P_N$ of the N+1 cells comprised in a neighbourhood U of the actual cell P_0, as shown on fig. 1.

Figure 1



For image operators P_i denotes a grey value, and the transition function (1) assumes mostly a compact analytical form. In an unified approach image operators underlie to the same constraint as cellular automata, namely:

$$f(U) \in \{1 \ldots k\} \text{ or: } f(U) \in S_k \qquad (2)$$

where the digits 1..k are the elements of the state set S_k. Let us consider now the problem of reversibility. Most cellular automata and image operators are not reversible, i.e. the previous state P_0 of a cell can not be deducted from its actual state Q_0 and from the states of its neighbours $Q_1 \ldots Q_N$. An original image can not be reconstructed after having been processed, for instance, with a maximum operator. The property of reversibility lends itself to different interpretations, depending upon the imposed constraints. The reversibility problem has been treated in all generality in /1/, where necessary conditions for constructing reversible transition functions are derived. In /2/ reversibility is understood as the possibility of backtracking the previous state R_0 of the central cell of U, knowing the cell's states f(U) and Q_0 at two consecutive time steps. The following transition function is proposed:

$$Q_{0\,t+1} = f(U) - R_{0\,t} \qquad (3)$$

$t-1$: time steps

meeting sufficient conditions for reversibility. As stated in /2/, any f(U), without restriction of its arguments to some particular subset of $\{P_0 \ldots P_N\}$ meets this condition. The concept of reversibility, upon which the present work is based, differs somewhat from the above-exposed one, from which it is derived, and is probably more realistic, as far as practical applications to image processing are concerned. Here reversibility must subsist under the constraint of disposing of informations about cell states in only one time step, namely the present one. Under this constraint let us consider the following transition function, derived from (3):

$$Q_0 = f(P_0 , P_1 \ldots P_N) - P_0 \qquad (4)$$

where f may be a non-linear function, as for instance maximum, median, majority, and so on. With reference to fig.1, a sufficient condition for the reversibility of (4) is that f is of the type:

$$f(U) = f(P_1 \ldots P_n) \quad \text{with } n = \frac{N}{2} \text{ for square windows} \qquad (5)$$

Under this condition the inverse transition function can be written as:

$$P_0 = f(P_1 \ldots P_n) - Q_0 \qquad (6)$$

The constraint (5) allows to put into evidence the term P_0, which does not appear any more among the arguments of the non-linear function f. Further, the arguments of f in (6) are all known: they are the reconstructed original cell states (grey values) of already processed cells. The condition is, of course, that the original cell states of the first $(\sqrt{N+1} - 1)/2$ scan rows and columns are given as initial conditions. Notice that the transition function (4) under the constraint (5) represents a causal non-recursive filter. Its inverse (6) is causal and recursive, in analogy with the linear systems.

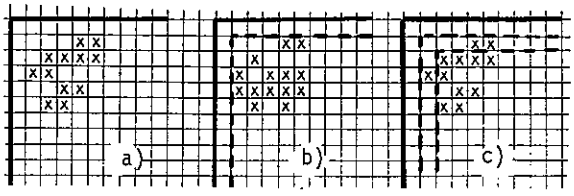
2. SOME EXAMPLES OF CAUSAL REVERSIBLE OPERATORS

The considerations of the previous section point out that causality, i.e. the restriction of the argument set to input cell states occurring in time before the output, constitutes a sufficient reversibility condition for transition functions of the type of (4). This will be illustrated in this section by aid of some examples. A window of 3x3 cells with N=8 and n=4, as shown on figure 2d, will be considered unless otherwise indicated.

Example 1 (see fig. 2) $S_k = \{0 , 1\}$ k=2

$$Q_0 = \text{majority}(P_1 \ldots P_4) - P_0 \quad \text{direct op.} \qquad (7)$$

$$P_0 = \text{majority}(P_1 \ldots P_4) - Q_0 \quad \text{inverse op.} \qquad (8)$$



S_k:(blank=0, X=1)

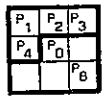


Figure 2
a) Initial configuration
b) Operator result
c) Inverse transformation

In case of parity (n=4) the majority is arbitrarily attributed to the state 1. The result of the inverse transformation (fig. 2c) is identical with the initial configuration of fig. 2a. The recursive inverse operator must be initialized by giving the original cell states at the border of the cellular space (between dotted lines in fig. 2c).
The next example utilizes the same processing window and a larger state set S_k:{0,1,2,3},k=4,

$$Q_0 = R^{II}(P_1 \ldots P_4) - P_0 \quad \text{direct operator} \quad (9)$$

$$P_0 = R^{II}(P_1 \ldots P_4) - Q_0 \quad \text{inverse operator} \quad (10)$$

where R^{II} represents the ordered rank function yielding the second largest value. This Example 2 is shown on fig. 3. Figs. 3a and 3c are identical, since the operator is reversible.

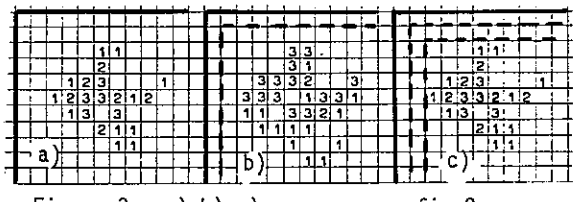


Figure 3 a),b),c): same as on fig.2

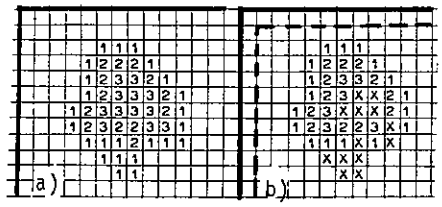
A more general operator structure, of the type:

$$Q_0 = f(P_1 \ldots P_n) + c\,P_0 \quad (11)$$

can comply with the reversibility requirement only if $c = \pm 1$. In fact, the state set S_k: {0, 1...k-1}, which is automorph with respect to the modulo-k sum and subtraction, has not this property as far as multiplication and division are concerned. From a general point of view, automorphism is required for any operation on a term of (11) involving P_0.

A further generalization of sufficient reversibility conditions can be attained by observing that the relation between the cell states P_0 and Q_0 does not need to be expressed by an analytical formula, provided that the state set is automorph with respect to this relation. The following Example 3 illustrates an operator, whose transition function is given in an implicit form. The state set S_k: {blank=0,1,2,3,X}, with k=4+1, is extended by including the additional state X, denoting the case in which a cell has the same state as the majority of its

Figure 4
a),b): as on fig.2



neighbours. The transition function is:

$$Q_0 = \begin{cases} X \text{ if } P_0 = \text{majority}^{\S}(P_1 \ldots P_4) \\ P_0 \text{ otherwise} \end{cases} \quad \text{dir.op.} \quad (12)$$

$$P_0 = \begin{cases} \text{majority}^{\S}(P_1 \ldots P_4) \text{ if } Q_0 = X \\ Q_0 \text{ otherwise} \end{cases} \quad \text{inv.op.} \quad (13)$$

§: in case of parity the majority is arbitrarily attributed to the state with lowest digit

A further alternative to the general form (11) of the transition function for k=2, S_k: {0,1} is provided by a logical function of P_0 and of the other arguments, as:

$$Q_0 = g(P_k\ ,\ P_0) \quad \text{with } P_k = h(P_1 \ldots P_n) \quad (14)$$

and g,h are logical functions. In this case the reversibility is assured if and only if the truth table of the function (14) contains all the possible combinations between the states of Q_0 and P_k. It can be shown that the only non-trivial cases, in which this condition is met, are if g is the antivalence or the coincidence between P_k and P_0.

Concluding this section, the Example 4 is given below, which is more closely related to image processing applications. Here reversibility means restoration of an image impaired by a non-linear distortion, in this case by hysteresis combined with time-lag (due, for instance, to the image pickup device), approached by the transition function:

$$Q_0 = f(P_1\ ,\ P_2) + P_0 \quad (15)$$

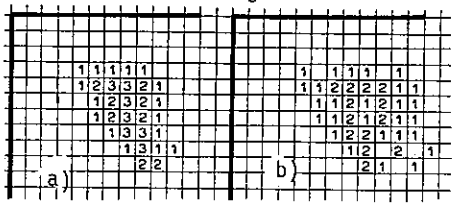
where:
$$f(P_1\ ,\ P_2) = \begin{cases} 0 \text{ if } P_1 = P_2 \\ +1 \text{ if } P_1 > P_2 \\ -1 \text{ if } P_1 < P_2 \end{cases} \quad (16)$$

defined in the two-cell processing window shown on fig. 5c. The inverse operator, performing the image restoration, is given by:

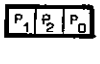
$$P_0 = Q_0 - f(P_1\ ,\ P_2) \quad (17)$$

Fig. 5 shows the application of this operator to a cellular field with S_k: {0,1,2,3}, k=4. The effect of the grey-tone smearing and lag, evident on fig. 5b, is cancelled by the restoration process, which restitutes the original pattern 5a.

Figure 5



a),b) as in fig.2



c)

3. REVERSIBILITY CONDITIONS FOR NON-CAUSAL OPERATORS AND EXAMPLES

The circumstance that causality, as exposed at the beginning of section 2, is required for ensuring reversibility under the assumptions made here, poses sensible limitations to the variety of operator structures, which can fulfil the reversibility conditions. For image processing and for simulating the evolution of two-dimensional cellular automata it would be useful to use processing windows, which lie fairly symmetrically round about the actual cell. Given the constraints exposed in section 1, the question arises if the equivalent of a symmetrical window can be obtained by performing the cascade of two reversible end causal operators, O_1 and O_2, in two opposite scan directions, as shown on fig.6.

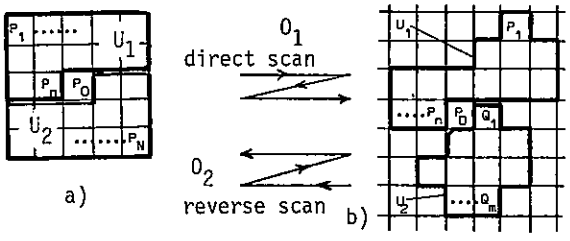


Figure 6

It is evident that for O_2, with processing window U_2 and in reverse scan, analog causality and reversibility considerations can be made as for U_1 in sections 1 and 2. The problem is now to find sufficient reversibility conditions for the cascade $O_1 * O_2$ of the two operators:

$$O_1 : \quad Q_0 = f(P_1 \ldots P_n) - P_0 \qquad \text{and} \qquad (18)$$

$$O_2 : \quad R_0 = g(Q_1 \ldots Q_m) - Q_0 \qquad (19)$$

each one reversible in its appropriate scan mode. Fig. 6 illustrates the general situation and the notations used. Writing down explicitely the cascaded transition function of $O_1 * O_2$, and introducing the notation:

P_j^i : neighbour P_j of Q_i in $U_1(Q_i)$

and $f_i = f(P_1^i \ldots P_n^i)$ for $i = 1 \ldots m$ (20)

one obtains from (18) and (19):

$$R_0 = g\,(f_1 - P_0^1) \ldots (f_m - P_0^m) - f(P_1 \ldots P_n) + P_0 \qquad (21)$$

A sufficient condition for reversibility is:

$$\forall i,j: \; P_j^i \not\equiv P_0 \quad \text{with} \quad 1 \leqslant i \leqslant m \,,\; 1 \leqslant j \leqslant n \qquad (22)$$

This ensures that P_0 does not appear among the arguments of the non-linear and (in the general case) not univocally reversible function f. The window shape of fig. 1 conflicts with this requirement. Therefore no symmetrical processing window can be simulated by cascading operators of the type described in the examples 1 to 3. The reversibility condition will be now illustrated by aid of an example, in which both functions f and g are represented by the maximum function.

Example 5 (see fig. 7) $S_k : \{0,1,2,3\}$ $k=4$
The processing window is chosen with n=m=2. The cells belonging to the windows $U_1(Q_1)$ and $U_1(Q_2)$, marked with X in fig. 7f, do not include P_0. The transition functions of the direct operators are given by:

$$O_1 : \quad Q_0 = \max(P_1 , P_2) - P_0 \qquad (23)$$

$$O_2 : \quad R_0 = \max(Q_1 , Q_2) - Q_0 \qquad (24)$$

and those of the inverse operators by:

$$O_2^{-1} : \quad Q_0 = \max(Q_1 , Q_2) - R_0 \qquad (25)$$

$$O_1^{-1} : \quad P_0 = \max(P_1 , P_2) - Q_0 \qquad (26)$$

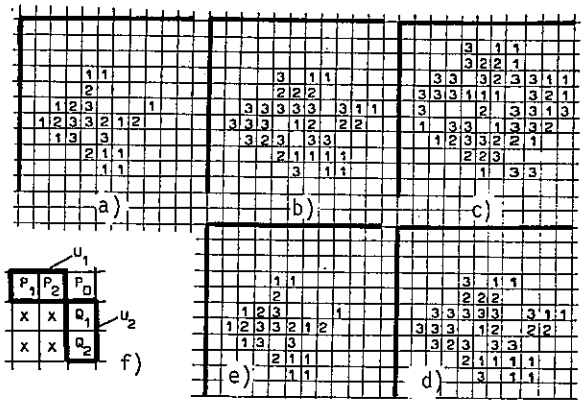


Figure 7

The figures 7a to 7e illustrate the successive steps, leading from the original P-pattern of fig. 7a through $O_1 * O_2$ to the transformed R-pattern of fig. 7c, and then through $O_2^{-1} * O_1^{-1}$ back to the original pattern of fig. 7e. The overall processing window can be regarded as the convolution of U_1 and U_2.

The example 5 achieves a rather poor realization of a two-dimensional processing window aiming at a spatial symmetry around the actual cell P_0, namely the cell complex P_1, P_2, Q_1, Q_2 and the Xs of figure 7f. Can a better symmetry be attained by a proper choice of U_1 and U_2 ? The figures 8a to 8c give an answer to this question.

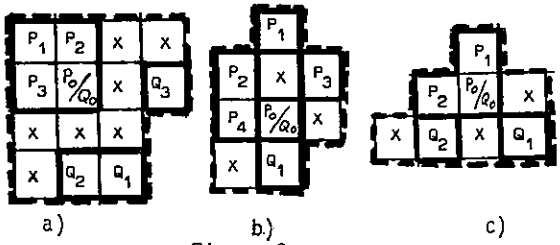


Figure 8

Fig. 8b represents the best approximation of an ideal 3x3 window. The drawback of the windows shown on fig. 8b is that O_2 must be of rather

trivial nature, since it involves only one argument, i.e. the cell Q_1 .

In fig. 9 is given an example of the performance of the nearly-symmetrical processing window of fig. 8a.

Example 6 (see fig. 9)

S_k: {blank=0,1,2,3} k=4

The direct operators are:

$$O_1 : \quad Q_0 = \max(P_1 , P_2 , P_3) - P_0 \qquad (27)$$

$$O_2 : \quad R_0 = \max(Q_1 , Q_2 , Q_3) + Q_0 \qquad (28)$$

and the inverse operators are:

$$O_2^{-1} : \quad Q_0 = R_0 - \max(Q_1 , Q_2 , Q_3) \qquad (29)$$

$$O_1^{-1} : \quad P_0 = \max(P_1 , P_2 , P_3) - Q_0 \qquad (30)$$

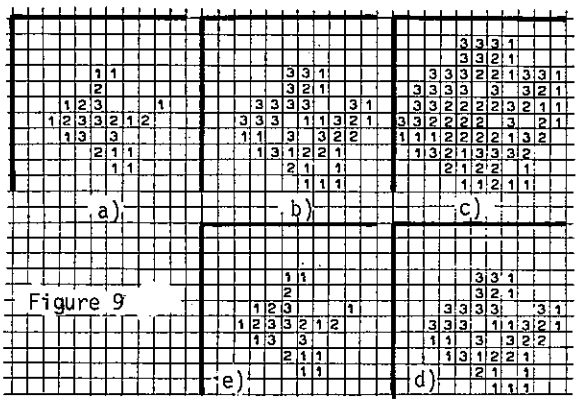


Figure 9

The term $+Q_0$ in O_2 has been chosen with the aim of compensating the term $-P_0$ of O_1 , in order to confer to the cascade $O_1 * O_2$ the overall character of a maximum operator. This character can be recognized by considering the fig. 9c, taking account of the fact that pattern-internal blanks result from a modulo-4 sum or subtraction, and correspond therefore to the grey level 4. The peculiarity of $O_1 * O_2$ is that it is a reversible maximum operator, a quite strange specimen in the operator world.

4. CONCLUSION

The price for meeting the reversibility requirements is a severe cut into the broad set of non-linear operators of widespread use for image processing purposes, or into the broad set of all possible cellular automata. Nevertheless, as pointed out by the examples shown, a fair stock of useful operators seems to remain at our disposal for quite a number of concrete applications. An interesting particular case arises when an image-impairing transformation can be decomposed into a cascade of reversible operators. In this case the restoration can be performed by means of the corresponding inverse operators. If we have a single operator instead of a cascade, this can be accomplished by a recursive structure in real time. Some of the terrain lost in the operator landscape can be won back, without giving up the reversibility, if we cascade two operators with opposite scan directions. However, the proposed condition (22) constitutes a serious limitation to the processing window shapes, which can be meaningfully implemented, taking account of the point-symmetrical statistical properties of natural images. As far as cellular automata are concerned, more degrees of freedom seem to be given.

REFERENCES

/1/ Toffoli, T., Cellular automata mechanics, Ph.D. Thesis, University of Michigan, Computer Science Dept., 1977.

/2/ Margolus, N., Physics-like models of computation, in: Farmer, D., Toffoli,T. and Wolfram, S. (eds.), Cellular auotmata (North-Holland, Amsterdam, 1984) pp. 81-95.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN OPTIMAL ALGORITHM FOR COMPUTING THE RELATIVE CONVEX HULL OF A SET OF POINTS IN A POLYGON

Godfried T. Toussaint

School of Computer Science
McGill University
Montreal, Canada

Let $P = (p_1,p_2,\ldots,p_n)$ be a simple polygon with n vertices which contains another simple polygon $Q = (q_1,q_2,\ldots,q_n)$. The *relative convex hull* of Q given P, denoted by $CH(Q|P)$, is the minimum-perimeter polygon containing Q constrained to lie in P. Sklansky and Kibler were the first to present an algorithm to compute $CH(Q|P)$ for a restricted case of this problem, but did not provide a complexity analysis. Here we show that their algorithm runs in $O(n^2)$ time. We also point out that using existing tools from computational geometry the unrestricted case can be solved in $O(n)$ time. Next, we consider the more general problem of computing $CH(S|P)$ where $S = \{x_1,x_2,\ldots,x_n\}$ is a set of n points in P. We present an algorithm for computing $CH(S|P)$ in $O(n \log n)$ time. Since $\Omega(n \log n)$ is a lower bound for this problem, it follows that our algorithm is optimal to within a constant factor.

1. INTRODUCTION

Let $P = (p_1,p_2,\ldots,p_n)$ be a simple polygon with n vertices which contains another simple polygon $Q = (q_1,q_2,\ldots,q_n)$. Given two points a and b in a simple polygon R the *geodesic path* between a and b is defined as the shortest path that connects a and b and also is contained in R. A polygon $R \subseteq P$ is *convex relative to P* if the goedesic path, in P, between every pair of points in R is also contained in R. The *convex hull* of polygon $Q \subseteq P$ *relative to P*, denoted by $CH(Q|P)$, is defined as the intersection of all sets *convex relative to P* which also *contain* Q. Note that this definition is not restricted to polygons contained in P. The same applies to sets of polygons or a set of points taking the place of polygon Q. Intuitively, $CH(Q|P)$ is the minimum perimeter polygon that contains Q and is constrained to lie in P. Relative convex hulls, also known as *minimum-perimeter polygons* [1], have been studied for some time in image processing and computer vision [1]-[3], and more recently have been applied to visibility [4]-[6] and collision avoidance problems in robotics [7]-[8].

In this paper we consider the following two *relative convex hull* problems:

Problem-1: Given a polygon Q contained in P, find $CH(Q|P)$.

Problem-2: Let $S = \{x_1,x_2,\ldots,x_n\}$ be a set of n points contained in P. It is required to compute $CH(S|P)$.

Sklansky and Kibler [2] were the first to present an algorithm for computing $CH(Q|P)$ for a restricted case of Problem-1. They restricted themselves to analysing *"normal complexes on an acute mosaic"*. In essence this means that in our terminology we are not only given two nested polygons P and Q, but the region in between P and Q is given as a partition into convex polygons with the property that the union of each pair of adjacent polygons (i.e. polygons sharing a side) is also a convex polygon. No complexity analysis is provided in [2]. In this paper we show that Algorithm-M of Sklansky and Kibler [2] runs in $O(n^2)$ time in the worst case. More recently, El Gindy [9] obtained an $O(n \log n)$ algorithm to solve the unrestricted case of Problem-1. Here we point out that using recent results from computational geometry, Problem-1 can be solved in $O(n)$ time without the above restrictions.

In the remainder of the paper we consider the more general problem of finding the $CH(S|P)$, where $S = \{x_1,x_2,\ldots,x_n\}$ is a set of n points in P. In Figure 1 the $CH(S|P)$ is illustrated in dashed lines. Since $CH(\{Q\}|P)$, where $\{Q\}$ is a set or r polygons $Q_1,Q_2,\ldots,Q_r$ with a total of n vertices, is equivalent to $CH(\text{vertices of } \{Q\}|P)$, it follows that $CH(S|P)$ also includes this problem as a special case thus generalizing the work of Sklansky and Kibler [2] and El Gindy [9]. We present an algorithm for computing $CH(S|P)$ in $O(n \log n)$ time in the worst case. Since we can show that $\Omega(n \log n)$ is a lower bound for this problem it follows that our algorithm is optimal to within a constant factor.

2. COMPUTING THE RELATIVE CONVEX HULL OF ONE POLYGON INSIDE ANOTHER

2.1. The Algorithm of Sklansky and Kibler

Sklansky and Kibler [2] presented an algorithm (Algorithm M) to compute CH(Q|P) when the region between Q and P is given as a partition into convex polygons with the property that the union of each pair of adjacent polygons is also convex. For details of how this restriction comes about the reader is referred to [2]. Also, in the interest of brevity, we do not describe Algorithm M in detail here. No complexity analysis of Algorithm M is provided in [2]. We now show by an example that Algorithm M may require $O(n^2)$ operations in the worst case. Let the region between P and Q be triangulated as illustrated in Figure 2 where P and Q are given in counterclockwise order. If we make sure that the y coordinates of Q's vertices increase slowly enough then the unions of adjacent triangles will be convex quadrilaterals and the restrictions considered in [2] will be satisfied. In Figure 2, the diagonals d_i are the "windows" in the terminology of [2]. Algorithm M starts at some anchor vertex known to be on CH(Q|P), say q_1 and traverses the "windows" in order, building a *"visibility cone"* which decreases in angle at each step until it disappears. For example, referring to Figure 2, when traversing "window" d_1 the "visibility cone" is determined by angle $p_1q_1q_2$. At the next step, "window" d_2 reduces the visibility cone to angle p_2,q_1,q_2, and so on. When a window is no longer in sight from q_1 Algorithm M starts again from a new anchor vertex which is the last vertex on the appropriate side of the final cone. The straight line segments between successive anchor vertices form CH(Q|P).

Now, in Figure 2 we make sure that (1) all vertices p_i, i=1,2,...,n-1 are visible from q_j for $j < i$, (2) the vertices of Q are monotonically increasing in y coordinate, and (3) p_n lies to the left of the directed line through q_{n-2} and q_{n-1}. This construction implies that for each "anchor" vertex O(n) "windows" must be looked at before a "window" disappears. Furthermore, at each iteration the "anchor" vertex advances by only one vertex on Q. Therefore $O(n^2)$ operations are required.

2.2 Solving Problem-1 in Linear Time

Let the *annular polygon* defined as the region between P and Q, along with its boundary, be denoted by ANN(P-Q). It is easy to see that CH(Q|P) must pass through any extreme vertex of Q. Let q_{ymax} be the vertex of Q with maximum y coordinate. It follows that $q_{ymax} \in CH(Q|P)$. It is also easy to see that there must exist a vertex p* of P, visible from q_{ymax} (i.e., the line segment $[p^* \; q_{ymax}]$ must lie in ANN(P-Q)) with y coordinate greater than that of q_{ymax}. It is straightforward to verify that p* can be found in O(n) time. It follows that by cutting ANN(P-Q) at $[p^* \; q_{ymax}]$ and distinguishing between two copies of q_{ymax}, say q_L to the left and q_R to the right of the cutting segment, we can convert Problem-1 in linear time to a shortest path problem between two points in a simple polygon. In this case the two points are q_L and q_R.

Let the shortest path (*geodesic path*) between two points a and b in P be denoted by GP(a,b|P) where the direction is from a to b. Geodesic paths find application in many areas such as image processing [10], [11], operations research [12], visibility problems in graphs [6], and robotics [7]. Recently, Chazelle [13] and Lee & Preparata [12], independently discovered the same O(n log n) algorithm for computing GP(a,b|P). Both of these algorithms first triangulate P in O(n log n) time and then find the shortest path in O(n) time. Another algorithm due to El Gindy [9] computes GP(a,b|P) without first triangulating P. In [9] P is first decomposed into *monotone* polygons in O(n log n) time and subsequently the shortest path is found in linear time. It follows from the above discussion that there exist at least two methods for solving Problem-1 in O(n log n) time without the restrictions imposed by Sklansky and Kibler [2]. However, we can do even better than this. In the above algorithms that first triangulate P, only the triangulation step requires O(n log n) time. But very recently Tarjan and Van Wyk [14] have obtained a linear-time triangulation algorithm for simple polygons. Therefore Problem-1 can be solved in O(n) time.

3. COMPUTING THE RELATIVE CONVEX HULL OF A SET OF POINTS IN A POLYGON

In this section we provide a high-level description of Algorithm-RCH(S|P) for computing the relative convex hull of S given P. We only define new terms for the sake of brevity. It is assumed the reader is familiar with the basic notions and definitions concerning triangulations, shortest paths, and point-location algorithms [4], [12]-[13], [15].

Definition: Let C_1 and C_2 be two *oriented*, possibly self-intersecting, curves. We say that C_1 and C_2 have a *proper crossing* if, as we follow C_1 from its starting point we encounter a region where C_2 intersects C_1 and actually *switches* from one side of C_1 to the other.

Definition: A closed polygonal path C is a *polygonal circuit* (alternately, a *weakly-simple* polygon) if every pair of distinct points on C partitions C into two polygonal chains C_1 and C_2 that have no *proper crossings*.

Definition: The *relative convex hull* of S in P is the *minimum-perimeter weakly-simple* polygon containing S and constrained to lie in P.

Note that a *weakly simple polygon* is a slight generalization of the notion of a simple polygon

to allow some vertices and edges to be used more than once. Thus it makes sense to speak of its *interior* and *exterior*. In fact, its interior is *simply*, but not necessarily *polygonally connected* [17]. The important point of *weakly simple* polygons in this context is that many geometric structures and algorithms (such as triangulations and convex hulls) designed for *simple* polygons are also valid for *weakly-simple* polygons without increasing the order of complexity.

The basic idea for Algorithm-RCH(S|P) is to convert this problem in O(n log n) time to an instance of Problem-1 by first computing some *weakly-simple* polygon, denoted by WSP(S|P), that lies in P and contains S. Then the problem is solved by computing RCH(WSP(S|P)|P).

ALGORITHM-RCH(S|P)

Input: A simple polygon $P = (p_1,p_2,\ldots,p_n)$ with vertices specified by their cartesian coordinates in order and a set of points $S = \{x_1,x_2,\ldots,x_n\}$ lying in P also specified in cartesian coordinates.

Output: The convex hull of S relative to P, RCH(S|P).

Begin

Step 1: Triangulate P to obtain T(P).

Step 2: Compute the dual tree of T(P), DT(T(p)).

Step 3: Preprocess T(P) to support O(log n) time point-location queries.

Step 4: Determine the triangles in which S lies.

Step 5: For each triangle T_i of T(P) containing points $S_i \subseteq S$, compute the convex hull of S_i, $CH(S_i)$.

Step 6: For each set S_i lying in T_i compute the *connecting* vertices of $CH(S_i)$ lying closest to the lines colinear with the edges of T_i which are diagonals in T(P).

Step 7: Using a *tree-walk search* of the dual tree DT(T(P)), compute the shortest paths between consecutive *connecting* vertices in the order in which they are encountered.

Step 8: Concatenate the shortest paths computed in Step 7 with the appropriate subchains of the $CH(S_i)$ to yield the *weakly simple* polygon WSP(S|P).

Step 9: Triangulate ANN(P-WSP(S|P))

Step 10: Compute RCH(WSP(S|P)|P) by solving shortest path problem-1 in region ANN(P-WSP(S|P)).

End.

The correctness of ALGORITHM-RCH(S|P) follows from the previous discussion and the fact that Steps 5-8 do indeed yield the *weakly-simple* polygon WSP(S|P). This can be readily verified by the reader. Consider now the complexity of this algorithm. Steps 1 and 2 can be done in O(n) time with algorithms in [14] and [12], respectively. Step 3 can be done in O(n log n) time using Kirkpatrick's algorithm [15]. Since for each point $x_i \in S$ we must spend O(log n) time to locate a containing triangle, Step 4 requires O(n log n) time. Convex hulls in Step 5 can be computed using a variety of algorithms [18] in O(n log n) time. In Step 6 each set S_i contains at most *three connecting* vertices. Therefore they can all be computed in O(n) time. In Step 7 we compute several shortest paths between pairs of *connecting* vertices. It is well known that given a *sleeve* [12] the shortest path between two points can be computed in time proportional to the cardinality of the sleeve. By performing a *tree-walk search* we ensure that all the adjacent pairs of connecting vertices lie in *sleeves*. Furthermore, it is easy to see that the sum of the lengths of *all* the sleeves encountered is no more than twice the length of the *dual tree* which is O(n). Therefore Step 7 can be done in linear time. Step 8 is trivially linear. Step 9 is as Step 2, and finally Step 10 can be done in linear time as discussed in the previous section. We therefore have the following theorem.

Theorem: ALGORITHM-RCH(S|P) computes the *relative convex hull* of a set of n points in a simple n-gon in O(n log n) time.

It is known that Ω(n log n) is a lower bound to finding the ordinary convex hull of n points [19]. By discarding P and constructing a square that contains S we can convert Problem-2 in linear time to an ordinary convex hull problem. Therefore Ω(n log n) is also a lower bound on Problem-2 and ALGORITHM-RCH(S|P) is optimal.

ACKNOWLEDGEMENT

This paper was written while the author was visiting the Courant Institute of Mathematical Sciences of New York University in April, 1986. The author is grateful to Richard Pollack, Richard Cole and Micha Sharir for discussions on this problem.

[1] J. Sklansky, R.L. Chazin, and B.J. Hansen, "Minimum perimeter polygons of digitized silhouettes", *IEEE Trans. Computers*, Vol. C-21, March 1972, pp. 260-268.

[2] J. Sklansky and D.F. Kibler, "A theory of nonuniformly digitized binary pictures", *IEEE Trans. Systems, Man, & Cybernetics*, Vol. SMC-6, Sept. 1976, pp. 637-647.

[3] D.H. Ballard and C.M. Brown, *Computer Vision*, Prentice-Hall, Inc., 1982.

[4] G.T. Toussaint, "Shortest path solves edge-to-edge visibility in a polygon", *Pattern Recognition Letters*, in press.
[5] G.T. Toussaint, "A linear-time algorithm for solving the strong hidden-line problem in a simple polygon", Tech. Rept. SOCS-86.2, School of Computer Science, McGill University, January 1986.
[6] D. Avis, T. Gum, and G.T. Toussaint, "Visibility between two edges of a simple polygon", Tech. Rept. SOCS-85.20, School of Computer Science, McGill University, October 1985.
[7] G.T. Toussaint, "Shortest path solves translation separability of polygons", Tech. Rept. SOCS-85.27, School of Computer Science, McGill University, October 1985.
[8] B.K. Bhattacharya and G.T. Toussaint, "A linear algorithm for determining translation separability of two simple polygons", Tech. Rept. SOCS-86.1, School of Computer Science, McGill University, January 1986.
[9] H.A. El Gindy, "Hierarchical decomposition of polygons with applications", Ph.D. thesis, School of Computer Science, McGill University, May 1985.
[10] C. Lantuejoul and F. Maisonneuve, "Geodesic methods in quantitative image analysis", *Pattern Recognition*, Vol. 17, 1984, pp. 177-187.
[11] T. Asano and G.T. Toussaint, "Computing the geodesic center of a simple polygon", Tech. Rept. , School of Computer Science, McGill University, December 1985.
[12] D.T. Lee and F.P. Preparata, "Euclidean shortest paths in the presence of rectilinear barriers", *Networks*, Vol. 14, 1984, pp. 393-410.
[13] B. Chazelle, "A theorem on polygon cutting with applications", *23rd Annual IEEE Symposium on Foundations of Computer Science*, 1982, pp. 339-349.
[14] R.E. Tarjan and C.J. Van Wyk, "A linear-time algorithm for triangulating simple polygons", *Proceedings STOC*, May 1986.
[15] D. Kirkpatrick, "Optimal search in planar subdivisions", *SIAM J. Computing*, Vol. 12, No. 1, February 1983, pp. 28-35.
[16] G. Toussaint, "Complexity, convexity, and unimodality", *International J. of Computer and Information Sciences*, 1985.
[17] L.M. Kelly, Ed., *The Geometry of Metric and Linear Spaces*, Springer-Verlag, 1975.
[18] G.T. Toussaint, "A historical note on convex hull finding algorithms", *Pattern Recognition Letters*, Vol. 3, January 1985, pp. 21-28.
[19] D. Avis, "On the complexity of finding the convex hull of a set of points", *Journal of Discrete Mathematics*.

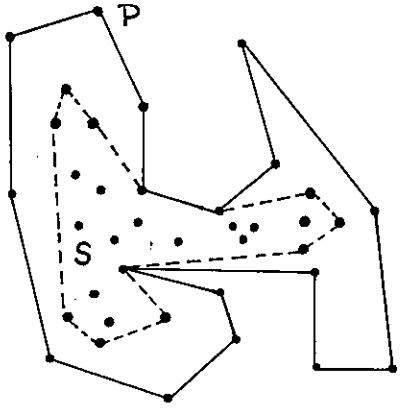


Fig. 1

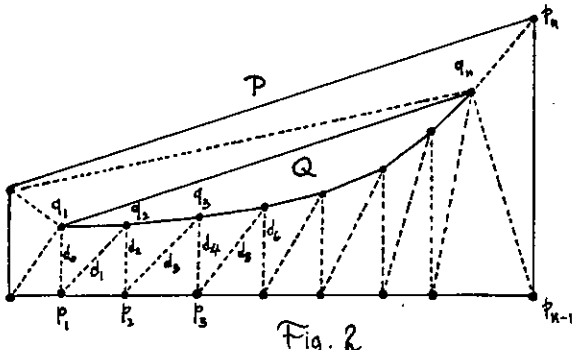


Fig. 2

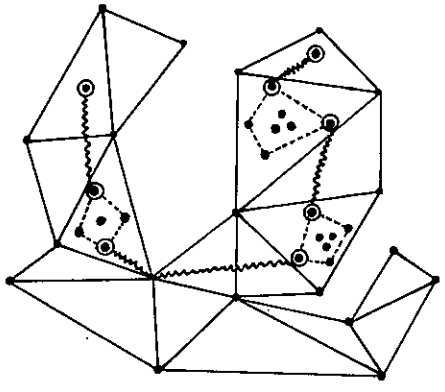


Fig. 3

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Sequential and Parallel Implementation of the Cylindrical Multivalued Transform

Y. J. Tejwani

Department of Computer Science
Southern Illinois University at Carbondale
Carbondale, Illinois 62901

Abstract: The implementation of the cylindrical multivalued transform is discussed in this paper. The transform coefficient are obtained for a few objects.

SECTION I

Theory and Basic Definitions

Definition I: The two dimensional cylindrical m-valued Rademacher function $\Psi_{np}(2,r,r\theta)$ are defined in terms of the one dimensional functions $\Psi(1,x)$, [1], as

$$\Psi_{00}(2,r,r\theta) = \Psi_0(1,r^2) \oplus \Psi_0(1,\theta)$$

$$= i \oplus \mu$$

$$\Psi_{np}(2,r,\theta) = \Psi_{00}(2, m^{n/2}r,\ m^{p+n/2}r\theta)$$

where

$$\frac{2\pi\mu}{m} + 2\pi\zeta \leq \theta < \frac{2\pi(\mu+1)}{m} + 2\pi\zeta \qquad (1)$$

$$\sqrt{k+i/m} \leq r < \sqrt{k+(i+1)/m}$$
$\zeta, k = 0,1,2,.....$
$\mu, i = 0,1,2,...,(m-1)$
$p,n = 0,1,2,...$
$\oplus$ = addition mod m (2)

Definition II: The two dimensional cylindrical m-valued pre-Walsh functions $\xi_{np}(2,r,r\theta)$ are defined in terms of the one dimensional pre-Walsh functions $\xi_p(1,x)$, [1], and the two dimentional cylindrical m-valued Rademacher functions $\Psi_{np}(2,r,r\theta)$ as follows

$$\xi_{00}(2,r,r\theta) = \xi_0(1,r^2) \oplus \xi_0(1,\theta) = 0$$

for p,n = 0 (3)

$$\xi_{n0}(2,r,r\theta) = \xi_n(1,r^2) \oplus \xi_0(1,\theta) = \xi_n(1,r^2)$$

for p = 0 (4)

$$\xi_{0p}(2,r,r\theta) = \xi_0(1,r^2) \oplus \xi_p(1,\theta) = \xi_p(1,\theta)$$

for n = 0 (5)

and

$$\xi_{np}(2,r,r\theta) = \Psi_{n1,p1}(2,r,r\theta) \oplus \ldots \oplus \Psi_{nk,pk}(2,r,r\theta) \qquad (6)$$

where

$n = m^{n1} + m^{n2} + \ldots + m^{nk}$

$p = m^{p1} + m^{p2} + \ldots + m^{pk}$ and

$\oplus$ is addition modm for $n \geq 1$, $p \geq 1$.

Definition III. The two dimensional cylindrical m-valued Walsh functions $\delta(\xi_{np}(2,r,r\theta))$ are defined as a mapping $\delta(i) = a_i$ of integers i between 0 and m-1 into a set of real numbers a's (a-coefficients) such that the orthogonality constraints are satisfied, i.e.,

$$\frac{1}{\pi}\int_0^{2\pi}\int_0^{1} \delta((\xi_{np}(2,r,r\theta)))\ \delta((\xi_{\varsigma q}(2,r,r\theta)))\, dr\, d(r\theta) \quad \text{for} \quad \begin{matrix} 0 \leq (n,\varsigma) \leq m-1 \\ 0 \leq (p,q) \leq m-1 \end{matrix} \qquad (7)$$

or

$$\frac{1}{\pi}\int_0^{2\pi}\int_0^{1} \delta_{np}(r,r\theta)\, \delta_{\varsigma q}(r,r\theta)\, dr\, d(r\theta) \quad \text{for} \quad \begin{matrix} 0 \leq (n,\varsigma) \leq m-1 \\ 0 \leq (p,q) \leq m-1 \end{matrix} \qquad (8)$$

where $\delta_{np}(r,r\theta) = \delta(\xi_{np}(2,r,r\theta))$

Independent Constraints

The number of constraints defined by equation (8) above is equal to $\{(1/2)m(m-1)\}^2$. However, not all of the constraints are independent. It has been shown [2] that the number of independent constraints is (m+1)/2. The constraining equations are as follows.

1) $\sum_{k=0}^{(m-1)} a_k = 0$ (first constraint) (9)

2) $\sum_{t=0}^{m-1} a_t a_{k\oplus t} = 0$ (((m-1)/2) constraints) (10)

The above constraints may be used to evaluate the "a" coefficients.

Completeness

The system of orthogonal functions $\delta_{np}(r,r\theta)$'s has been shown [2] to be complete in two dimensional functional space $C_f[0\leq\theta r^2\leq 1]$ of piecewise constant functions, where each piece is an integral multiple of $(1/m^{2n})$.

Now, since the $\delta_{np}(r,r\theta)$'s form a complete orthogonal system in C_f any piecewise continuous function $g(r,r\theta)$ can be expressed as a series, i.e.,

$$g(r,r\theta) = \sum_{j=0}^{\infty}\sum_{i=0}^{\infty} c(i,j)\ \delta_{ij}(r,r\theta) \qquad (11)$$

Therefore,

$$c(i,j) = \frac{\int\int_{a\leq\theta r^2\leq b} g(r,r\theta)\ \delta_{ij}(r,r\theta)\ dr\ dr\theta}{\int\int_{a\leq\theta r^2\leq b} \delta_{ij}(r,r\theta)^2\ dr\ dr\theta} \qquad (12)$$

where [a,b] is the interval/area over which $g(r,r\theta)$ is defined.

SECTION II

Implementation

a) Parallel

The computation of the a-coefficients in definition III is exactly the same as in the one dimensional case [1]. Once these coefficients have been evaluated they may be stored in a table as shown in figure 1.a. The radial distances $r_{0n'}, r_{1n'}, r_{2n'}, \ldots$ over which the a-coefficients may assume constant values may be evaluated by the following equation for a given value of m and n'.

$$\sqrt{k+(i/m)} \leq r m^{n'/2} < \sqrt{k+(i+1)/m} \qquad (13)$$

where k, n'=0,1,2,... , and i=0,1,2,...(m-1). Similarly the angles $\theta_{p'}, \theta_{1p'}, \ldots$ over which the a-coefficients may assume constant values may be evaluated for a given value of m and p' by the following equation.

$$(2\pi\mu/m)+2\pi\zeta \leq \theta m^{p'} < 2\pi((\mu+1)/m)+2\pi\zeta \qquad (14)$$

where ζ, p'=0,1,2,...., and μ=0,1,2,...(m-1).

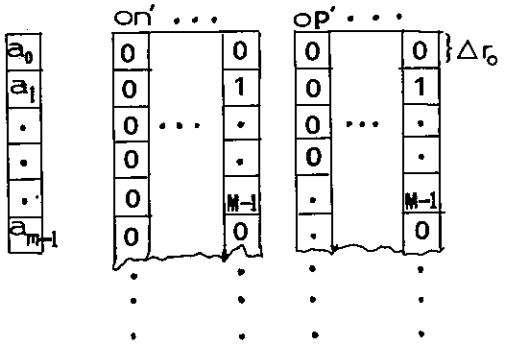


FIG 1a

Though in theory all values should be evaluated, in practice the values which result in $\Delta r_{0n'}$ or ($\Delta\theta_{p'}$)*(Maximum Linear dimension/2) being less than the resolution are not meaningful. The values of r and θ obtained are stored in a two dimensional table. Where each column contains a quantized range entry and Rademacher value entry over it.

Now let, $n=m^{n0'}+\ldots+m^{n'1}$ (15)

$p=m^{p0'}+\ldots+m^{p'}$ (16)

and define the function f:x→z* as follows

$$f(x)=\underbrace{z_0 z_0 \ldots z_0}_{(m-1)} \ \ldots\ \underbrace{z_k z_k \ldots z_k}_{(m-1)} \qquad (17)$$

f maps x satisfying equation (15) and (16) respectively into a string of $(m-1)^k$ bits. 'k' is the order of the highest Rademacher function allowed in the system within the limits of resolution. If $n\geq 1$ and $p\geq 1$ then the total number of z_i bits that are allowed to take on a value one is equal to the number of exponents in equation (15) or (16) which are equal to 'i' for a given value of n or p respectively. If x=0 then the entire string f(x) is zero.

The outputs of the encoder 'f' shown in figure 1.b is passed through a series of adders. The output of each adder indicates the number of times any Rademacher function occurs in the summation in equation (3) through (6). This number is multiplied by the values taken by Rademacher function from tables in figure 1.a and stored in the set of tables Nn' and Pp'. The contents of Nn' and Pp' tables respectively, are added module m and the resultant stored in one dimensional registers Nn and Pp respectively.

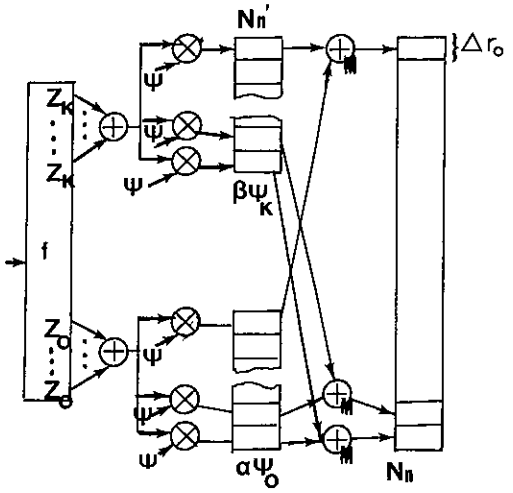


FIGURE 1.b DIAGRAM FOR Pp IS SIMILAR

The one dimensional registers Nn and Pp are combined to obtain a two dimensional table 'NP'. The range entries are combined using a

logical AND and the value entries over the combined range is obtained using a module m addition. These values are then transformed into the 'a' values using the δ mapping. See figure 1.c. The c[n,p] coefficient is then obtained by multiplying by the input and dividing by the energy in the δ_{np} function over the region.

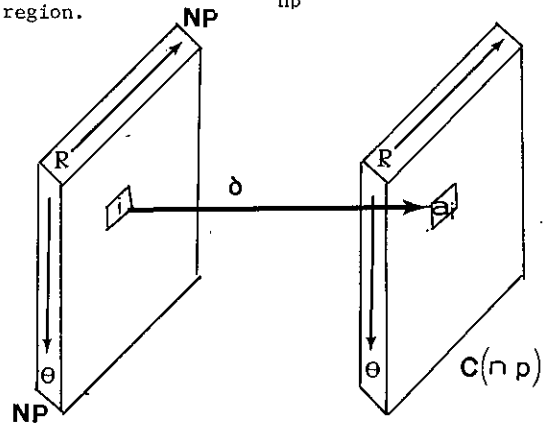


FIG 1C

Thus the parallel implementation may be written in a psuedo-language { a version of concurrent Pascal with vector operations} as follows.

```
program      ParallelCMLT;
Construct the tables shown in figure 1.a
   A[0:m-1];  {array to store a-coefficients}
   T1[1:MaxRow, 1:(2*MaxHarmonic)];
     {MaxRow=(linear dimension)/2* Resolution}
   T2[1:MaxRow, 1:(2*MaxHarmonic)];
   INPUT2:=RTHETA(INPUT[1:MaxRow,1:MaxRow]);
   {Convert the input to r,r   description}
procedure     Coefficients (n,p);
  begin
  cobegin
  Z:=f(n);   {Z and Y and 1by(m-1)^k vectors }
  Y:=f(p)    {see figure 1.b}
  end
  cobegin
  KNO:=+Z[,0:m-1];  {add the first m bits of Z
                     vector }
  :
  KNK:=+Z[,ϑ:ϑ+m-1];
  KPO:=+Y[,0:m-1];
  :
  KPK:=+Y[,ϑ:ϑ+m-1]
  end
  cobegin
  NKSIO[,1]:=T1[,1]; NKSIO[,2]:=T1[,2]*KNO;
  {obtain the contribution of each Rademacher
   function}
                  :
  NKSIK[,1]:=T1[,(2*MaxHarmonic-1)];
  NKSIO[,2]:=T1[,(2*MaxHarmonic)]*KNK;
  PKSIO[,1]:=T2[,1]; PKSIO[,2]:=T2[,2]*KPO;
                  :
  end
  cobegin
  {add all the contributing Rademacher functions}
  Nn[,2]:=NKSIO[,2]+ ... +NKSIK[,2];
  Np[,2]:=PKSIO[,2]+ ... + PKSIK[,2]
  end

  NP2[1:MaxRow,1:MaxRow]:=Nn[,2] M_0D(m) Mp[,2];
  {M_0D(m) takes the cross product of two vec-
   tors and adds the element pairs module m}
  NP2:=DELTA(NP2);
  {results in two dimensional cylindrical Walsh
   function}

  cobegin
  MUL:=INPUT2#NP2; {#=elementwise multiplica-
                    tion}
  ENERGY:=NP2#NP2
  end
  cobegin
  NUM:=(++MUL)/(MaxRow)^2;
  DEN:=(++ENERGY)/(MaxRow)^2
  {add the rows and columns and normalize it}
  end
  C[n,p]:=NUM/DEN
  end; {Procedure coefficient}

begin
  {Main Program}
  {Call procedure Coefficient NXN times in
     parallel}
end
```

b) Sequential

The sequential implementation is similar to the parallel implementation except that steps taken in parallel are implemented in sequence. For lack of space the procedure is not described here.

SECTION III

Transform of Some Continuous Objects

In this section the transform coefficients are evaluated for some continuous objects. The relation between the size and these coefficients is discussed.

In the discussion that follows let R_0, R_1, ..., R_k... and $\theta_0, \theta_1, \ldots, \theta_k$... be intervals over which the $\delta(n,p)$ function, may assume constant values. There are m such values $\{a_0, \ldots a_{m-1}\}$. The values takes over each of the m intervals between ((m(kDIVm)+(m-1)) and (m(kDIVm)) in a permutation of these a-coefficients. Let $b(j)=a_i$ be the value taken in the jth interval such that j+(m(kDIVm))=k.

a) Circular Object

Consider the circular object (shown in Figure 2.a).

$$g(r)=1 \text{ for } a \leq r < Rc \ , \ 0 \leq \theta \leq 2\pi \qquad (18)$$

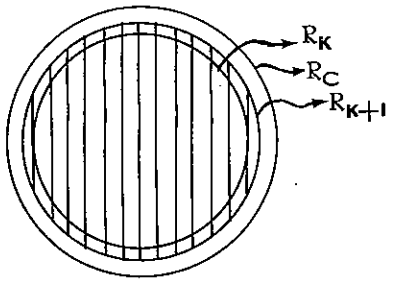


FIG. 2.a

The transform coefficients for this case obtained by using equation (12) are as follows,

$$c[0,0]=\pi R_c^2 / \pi a_0 = \text{Area} / \pi a_0 \quad (19)$$

$$c[0,p]= 0 \quad (20)$$

$$c[n,p]= 0 \qquad n\geq 1 \text{ and } p\geq 1 \quad (21)$$

$$c[n,0]= \frac{\pi b(x)(R_c^2-R_k^2)+\sum_{j=0}^{(k-s-1)} \pi\ b(j)(R_{s+j+1}^2-R_{s+j}^2)}{(\gamma+1)(\pi/m)\sum_{i=0}^{(m-1)} a_i^2} \quad (22)$$

where $\gamma=kDIVm$, $s=m*\gamma$ and $R_k\leq R_c<R_{k+1}$

A plot of the C[1,0] coefficient as a function of the area is shown in figure 2b.

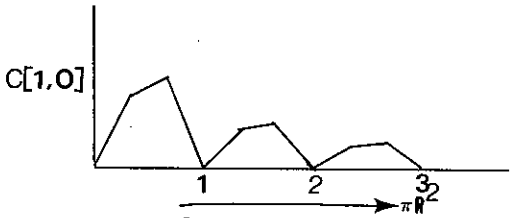


FIG 2b

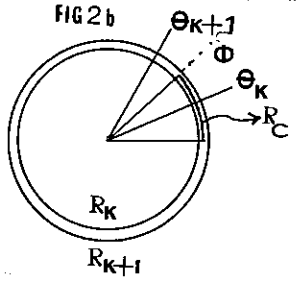


FIG 2c

b) Angular Region

Let Φ be the angle made by the angular region as shown in figure 2.c. Using equation (12) the coefficients evaluate as follows.

$$c[0,0]=(\Phi R_c^2)/\Phi a_0 \quad (23)$$

$$c[n,0]= \frac{(\frac{b(x)\Phi}{2\pi})(R_c^2-R_k^2)+\sum_{j=0}^{(k-s-1)}(\frac{b(j)\Phi}{2\pi})(R_{s+j+1}^2-R_{s+j}^2)}{(\gamma+1)(\frac{1}{m})\sum_{i=0}^{(m-1)} a_i^2} \quad (24)$$

$$c[0,p]= \frac{\frac{b(y)R_c^2}{2\pi}(\Phi-\theta_{k'})+(\frac{R_c^2}{2\pi})\sum_{j=0}^{(k'-j-1)} b(j)(\theta_{s'+j+1}-\theta_{s'+j})}{(\gamma+1)(\frac{1}{m})\sum_{i=0}^{(m-1)} a_i^2} \quad (25)$$

$$c[n,p]= \frac{(\frac{b(x,y)}{2\pi})(\Phi-\theta_k)(R_c^2-R_k^2)}{(\gamma+1)(\frac{1}{m})\sum_{i=0}^{(m-1)} a_i^2} + \frac{\sum_{t=0}^{(k'-s'-1)}\sum_{j=0}^{(k-s-1)}(\frac{b(j,t)}{2\pi})(\theta_{s'+t+1}-\theta_{s'+t})(R_{s+j+1}^2-R_{s+j}^2)}{(\gamma+1)(\frac{1}{m})\sum_{i=0}^{(m-1)} a_i^2} \quad (26)$$

For $n\geq 1$ and $p\geq 1$.

where $\gamma=kDIVm$, $\gamma'=k'DIVm$
$s=m*\gamma$, $s'=m*\gamma'$
$x=kMODm$, $y=k'MODm$
$R_k\leq R_c<R_{k+1}$, $\theta_{k'}\leq \Phi < \theta_{k'+1}$ and
$b(x,y)$ is two dimensional version of $b(j)$.

SECTION IV

The implementation of the CMLT was discussed and the transform coefficients were obtained for a few objects. The applications and other results will be discussed in a forth coming paper.

References

[1] Lieber, M.E. and Roesser, R.P.,"Multiple Real Valued Walsh Functions," Rine,D.C., (eds), Computer Science and Multivalued Logic, (North Holland) pp. 535-548.

[2] Tejwani, Y.J. "A Cylindrical Multivalued Logic Transform,"Proceeding of SPIE, Vol. 579, Intelligent Robots and Computer Vision.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CONDITIONING OF LOCAL IMAGE SIGNAL TO NOISE RATIO

D. Bosman, W. Bakker
BSC, Dept. of Electrical Engg.
Twente University of Technology
7500 AE Enschede, Netherlands.

With conventional video cameras the local signal to noise ratio (SNR) varies strongly with illuminance. This paper describes algorithms which control local SNR to a constant level at the expense of varying spatial mapping properties. Good large step response is preserved.

1. INTRODUCTION

Image features to be analysed or extracted often are local gradual and sudden jumps in gray level, irrespective of the background level in the region where they occur. In machine vision with changing environments, e.g. with moving objects and tool parts, the local background level can vary appreciably both in time and space. We assume that consecutive images are obtained by freezing the temporal situations to exclude movement blur; the variations of local background level from image to image remains.
The video signal to be processed is obtained from the local jumps in radiation. With passive objects and backgrounds, the sources of interest are local changes in reflectance ρ; unwanted are changes in local illuminance E_i. The relative reflectance $\Delta\rho/\rho$ is not contaminated with fluctuations in E_i provided the spatial extent in $\Delta\rho$ is smaller than that in E_i. Thus we wish to establish an interval scale in $\Delta\rho/\rho$ and have to accept responses of the processing action by sharp variations in illuminance.
The camera signal E_t is proportional to ρE_i+E_n, where E_n is the noise contribution of the camera transformed to the input. Assume that E_n is wide sense stationary, spatially uncorrelated, with constant standard deviation σ_n. The magnitude of σ_n limits the resolution in ρE_i. Let the minimum interval in ρ be $\Delta\rho_{min} = 3\,\sigma_n$, and the max. signal to noise ratio SNR_{max} of the camera $SNR_{max} = 300$; the range over which E_i can vary: a factor 10. Then the range in ρ (number of $\Delta\rho$ intervals) equals 10, which is considered too small. The situation can be improved if, by means of suitable processing, σ_n can be controlled to become proportional to E_i. This is only possible by spatial averaging.

This paper addresses a solution to this problem based on vision research [1,2]. The method employs adaptive spatial averaging (causing lower spatial cut-off frequency in darker parts of the image) while preserving a good response to large steps.

2. ANALYSIS OF THE PROBLEM

Assumptions:
a) The illuminance E_t of the target of the camera is the sum of a uniform background E_o of varying magnitude but constant over the area of local averaging, and a pattern E_p: $E_t = E_o+E_p$. The pattern signal E_p consists of a spatially random part E_w of constant variance density σ_w^2 and local correlated gradients E_s which constitute the signal of interest.
b) The camera receptors are uniformly distributed over the target, contributing spatially uncorrelated noise to the signal equivalent to random input illuminance E_n of constant variance density σ_n^2.
E_t and E_n are in $[W/m^2]$.
c) The receptor outputs are convolved with a PSF where locally weighed receptor signals are added to form a new output located at the center of the PSF. The form of the PSF is irrelevant, but we introduce here the restriction that the PSF is circular or quadrant symmetric.

Analysis:
The PSF is built up from concentric round or square rings. In the implementation of the algorithm, these are one pel wide. The centre ring is solid, of base area dA_o, the k-th ring has base area dA_k. The height h_k of the k-th ring is constant over the whole area dA_k.

The output f_k of the k-th ring is $f_k = E_t dA_k h_k =$

$$= h_k dA_k (E_o + \bar{E}_s + \bar{E}_w + \bar{E}_n) \quad (1)$$

where $\bar{E}_p$, $\bar{E}_w$ and $\bar{E}_n$ are the contributions of E_s, E_w and $\bar{E}_n$ averaged over the area dA_k. For large dA_k, E_w and E_n tend to zero.
The power P_k in the signal (excluding noise) of the k-th ring:

$$P(f_k) = (E_o + \bar{E}_s)^2 (h_k dA_k)^2 + \sigma_w^2 h_k^2 dA_k \quad (2a)$$

The noise power in the k-th ring

$$P_n = \sigma_n^2 h_k^2 dA_k \quad (2b)$$

For our purpose the random signal power and the random noise power are indistinguisable, so that one may write for the signal to noise ratio in the k-th ring

$$SNR(k) = \frac{E_o^2 (1+\bar{g})^2 (h_k dA_k)^2}{\sigma_t^2 h_k^2 dA_k} \quad (3)$$

where $\bar{g} = \frac{\bar{E}_s}{E_o}$; $\sigma_t^2 = \sigma_n^2 + \sigma_w^2$.

In the total PSF the signal gain is given by the total volume $\int h_k dA_k$:

$P(f) = E_o^2 (1+\bar{g}^2) (\int h_k dA_k)^2$; the noise power is obtained by adding all contributions of the noise density:

$$P_{n,k} = \sigma_t^2 \int h_k^2 dA_k .$$

Thus the SNR of the PSF is

$$SNR(f) = \frac{E_o^2 (1+\bar{g})^2 (\int h_k dA_k)^2}{\sigma_t^2 \int h_k^2 dA_k} \quad (4a)$$

In areas where there are no deterministic signal features such as edges, (4a) becomes

$$SNR(f, \bar{g} = 0) = \frac{E_o^2 (\int h_k dA_k)^2}{\sigma_t^2 \int h_k^2 dA_k} \quad (4b)$$

The requirement is that $SNR(f,g = 0) = K^2$, a constant, irrespective of E_o:

$$K^2 \sigma_t^2 \int h_k^2 dA^k = E_o^2 (\int h_k dA_k)^2 \quad (5)$$

Two cases are considered in detail. The first is uniform PSF (cylinder, its height can still be a function of E_o: $h_k = C_1 h(E_o)$). The second is Gaussian PSF with standard deviation $\sigma = \ell\sigma_o$ (σ_o being the smallest s.d. associated with maximum illuminance of the camera target). The height of the k-th ring is

$$h_k = C_1 . h(E_o) \exp -k^2/2\ell^2\sigma_o^2 ,$$

its area $dA_k = 2\pi k dk$.

2.1. Uniform PSF

Expression (5) yields

$$A_t = \int dA_k = \frac{K^2 \sigma_t^2}{E_o^2} \quad (6a)$$

or: the base area of the PSF must be inversely proportional to the square of the background illuminance E_o. The PSF "signal" output f:

$$f = \int E_t h_k dA_k = \bar{E}_t C_n . h(E_o) \frac{K^2 \sigma_t^2}{E_o^2}$$

where $\bar{E}_t$ is the total signal averaged over the base area of the PSF. The random fluctuations are attenuated by $(A_t)^{1/2}$ so that $\bar{E}_t = E_o(1+1/K) + \bar{E}_s$. Putting

$C_1 . h(E_o) = \frac{E_o}{K^2}$, the PSF output f becomes

$$f = (\frac{1+K}{K} + \bar{g}) \sigma_t^2 \quad (6b)$$

which is an interval scale with displacement $(1+K)/K$.

2.2. Gaussian PSF

For the PSF with normal distribution the effective base area (aperture) is according to expression (5):

$$A_t = \frac{(\int h_k dA_k)^2}{\int h_k^2 dA_k} = \frac{K^2 \sigma_t^2}{E_o^2} \quad (7a)$$

The effective base area A_t of the PSF is obtained from $A_t = (\int h_k dA_k)^2 / \int h_k^2 dA_k) = 4\pi\ell^2\sigma_o^2$

so that $\ell = \frac{K\sigma_t}{2E_o\sigma_o\sqrt{\pi}}$ (7b)

With $h_k = C_1 h(E_o) \exp -k^2/\ell^2\sigma_o^2$,

the effective height (contrast gain) of the

PSF becomes $h_t = 0.5\ C_1h(E_o)$. The output of the PSF, determined by the input signal weighed by its volume, is then:

$$f = 0.5C_1h(E_o)A_t = K^2\sigma_t{}^2C_1h(E_o)/2E_o{}^2.$$

Putting again $C_1h(E_o) = E_o/K^2$, the output becomes

$$f = 0.5\ (\frac{1+K}{K}\bar{g})\ \sigma_t{}^2 \qquad (7c)$$

For small disturbances in a constant background (linearisation condition) the cut-off frequency can be defined. Using (7a) and $A_t = 4\pi\ell^2\sigma_o{}^2$ one obtains for the Fourier transform of the PSF

$$M(r) = 0.5\ \exp-(rE_o)^2/2(K\sigma_t)^2 \qquad (8a)$$

r being the spatial frequency.
The -3db frequency r(-3) is obtained from $(rE_o)^2/2(K\sigma_t)^2 = \ell n\ u\sqrt{2}$ or

$$r(-3) = 0.83\ K\sigma_t/E_o \qquad (8b)$$

Thus the cut-off frequency is inversely proportional to E_o. In 3) it is shown that the response for large steps is considerably better than can be expected from (8b).

2.3. Algorithm gain

It can be shown that the condition A_t inversely proportional to $E_o{}^2$ is necessary and sufficient to satisfy (5) for every realiseable form of PSF.
In the expressions (6b), (7c) and (9) the volume V of the PSF is inversely proportional to E_o, meaning that strong gray level compression results. The output image shows mainly the changes $\bar{g}$. As the function $h(E_o)$ can be chosen at will, it is also possible to obtain output images which copy the spatial gray level distribution of the input image, e.g $h(E_o) = E_o{}^2/K^2$.
In general, $h(E_o)$ can be any function of E_o; $h(E_o) = C_2(E_o)^n$.
Then the PSF output f will be

$$f = \int E_t h_k dA_k = \int E_t dV = \bar{E}_t V \quad \text{with } E_t \text{ the weighed}$$

average of E_t. Fluctuations in f are given by

$$\left|\frac{\Delta f}{f}\right| = \left|\frac{\Delta \bar{E}}{\bar{E}}\right| + \left|\frac{\Delta V}{V}\right|$$

The term $\left|\frac{\Delta \bar{V}}{\bar{V}}\right|$ consists of two contributions, one equal to $n\ \frac{\Delta \bar{E}}{\bar{E}}$ and one due to volume noise caused by quantisation errors. Ignoring the latter, it is seen that n is a gain factor.

3. THE ALGORITHMS

The signal that controls the area of the base of the PSF must not be too noisy itself; therefore one would want a separate noise filter (e.g. averager) to derive the base area control signal or, even more logical, one may use the noise filtering properties of the PSF itself, leading to a recursive implementation of the algorithm. In figure 1a and 1b the flow diagrams of the two types are shown.

Type 1 (forward averaging, figure 1a) works well on images with relatively low contrast ratio (< 3, figure 2a), but for high contrast images it filters too much in dark regions (figure 2b). In fact, the growth of the effective area for dark regions can cause negative contrast effects.
Type 2 (recursive averaging figure 1b) is the better one. A light edge embedded in a dark region automatically decreases the PSA base area, avoiding negative contrast effects and yielding less blur. However, being recursive it consumes more computation time. Recursive filters can exhibit instability problems, especially when image dependent. Our analysis has not yet progressed to complete understanding of the phenomenon. Of course it is always possible to incorporate a damping mechanism.
In figures 3a and 3b the responses on respectively a gray level wedge, and on a typical image are shown.

REFERENCES

[1] Cornsweet, T.N., Mach bands without inhibition: intensity dependent positive pointspread models of early visual processing. ARVO meeting (1984).

[2] Bosman, D., Boterenbrood, H. and Van Huijstee, H.L.M., Two non-linear image enhancement algoritms, in Gelsema, E.S. and Kanal, L.N. Pattern Recognition in Practice II (North Holland, Amsterdam) (1986).

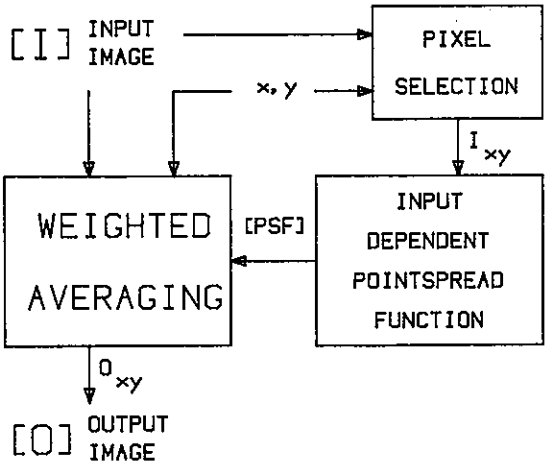


Figure 1a Forward signal dependent averager.

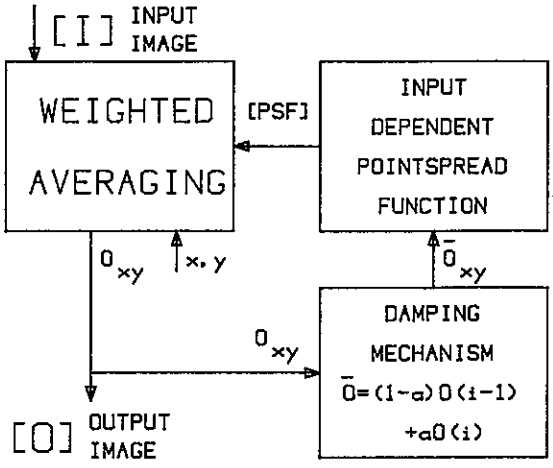


Figure 1b Recursive signal dependent averager.

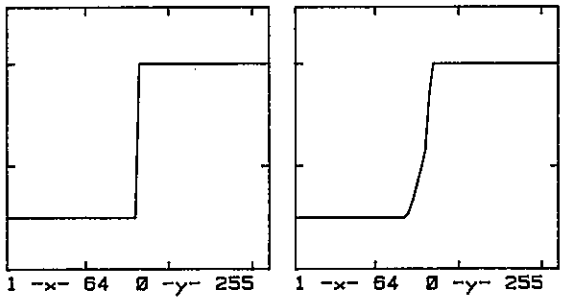


Figure 2a Left: Input image step from gray level 50 to 200.
Right: Step response of type 1 filter.

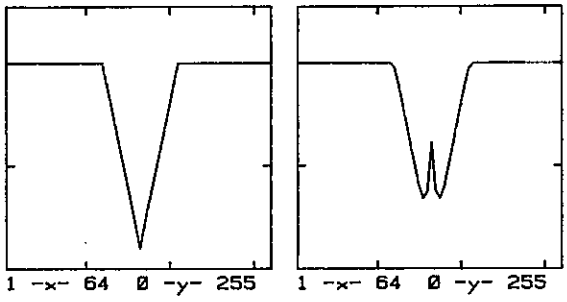


Figure 2b Left: Input image high contrast valley.
Right: Output of type 1 showing negative contrast effects in the dark region (due to wide average areas).

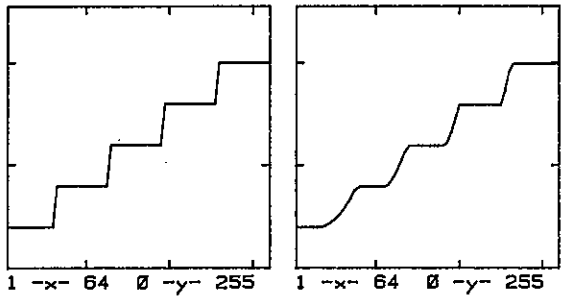


Figure 3a Left: Input showing a gray level wedge.
Right: Output of type 2 filter.

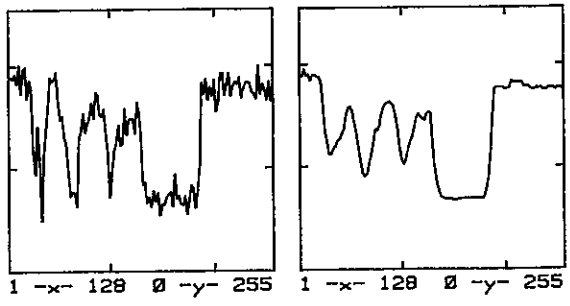


Figure 3b Left: Typical cross-section of a noisy image.
Right: Response of type 2 filter.
Note the difference in noise levels in the high and low gray levels.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMAGE PROCESSING SYSTEM /IPS/ FOR RECOGNITION AND ANALYZING OBJECTS

Ryszard S. CHORAŚ

Institute of Telecommunications and Electrical Engineering
Technological Academy
85-763 Bydgoszcz, Poland

Image Processing System /IPS/ here presented for recognition and analyzing object is able to extract objects of a image and to recognize them in a suitable way for practical applications. The system consists of the following main components: preprocessor, feature extractors/ analyser, structural processor and classifier. We describe some components of this IPS and some image processing techniques which are useful in the analysis of object.

1. INTRODUCTION

The problem of object recognition and analyzing has received considerable attention in the past several years. Several procedures have been proposed, e.g. Duda [2], Niemann [5], Rosenfeld [6], for object analyzing. Image Processing System /IPS/ here presented is able to extract objects and to recognize them in a suitable way for practical applications.

A general block diagram of the IPS is shown in Figure 1. The system consists of the following main components: preprocessor, feature extractors/analyser, structural processor and classifier. In the system described an object is recognized by comparing its parameters with those of various models. The input object and a model feature are defined to be similar if the difference between the object and model parameters is less than or equal to some constant. Associated with each feature is a list of parameters. If the data for the located outlines of the object and model in the Hough space belong to the error ellipse region then object and model are defined to be similar.

In this paper we describe some components of this IPS and some image processing techniques which are useful in the analysis of object.

2. PREPROCESSING

The generation of edge outlines of objects within an image characterizes significant attributes which can be utilized in feature extraction. The first step in the preprocessing is edge detection. Some of the simplest edge detectors in the spatial domain are based on the computation of the gradient operators([2],[6]). These operators perform well in the absence noise but poorly in the presence of noise. The effect of noise is reduced by median filtering Huang[4].

The image information is represented as a function of two variables (i,j). If $M = \{1,2,..i,..m\}$ and $N = \{1,2,..j,..n\}$ are the spatial domains, then $D = M \times N$ is the set of resolution cells and the digital image F is a function which assigns some graytone value $G \in \{0,1,..L\}$ to each and every resolution cell, e.g. $F : M \times N \longrightarrow G$.
Formally

$$D = \left\{(i,j) \;\middle|\; i \in M,\ j \in N\right\} \tag{1}$$

and

$$F = \left\{f(i,j) \;\middle|\; (i,j) \in D \text{ and } f(i,j) \in G\right\}$$

The median filtering operator is given by

$$y(k,l) = \underset{i,j \in w}{\text{median}} \left\{f(i,j)\right\} \tag{2}$$

$$Y = \left\{y(k,l)\right\} \qquad \frac{H-1}{2} < k \leqslant M - \frac{H-1}{2}$$

$$\frac{H-1}{2} < l \leqslant N - \frac{H-1}{2}$$

where Y is the output of an $H \times H$ median filter and w is chosen as

$$w = \left\{(i,j) \;\middle|\; k - \frac{H-1}{2} \leqslant i \leqslant k + \frac{H-1}{2}\right\}$$

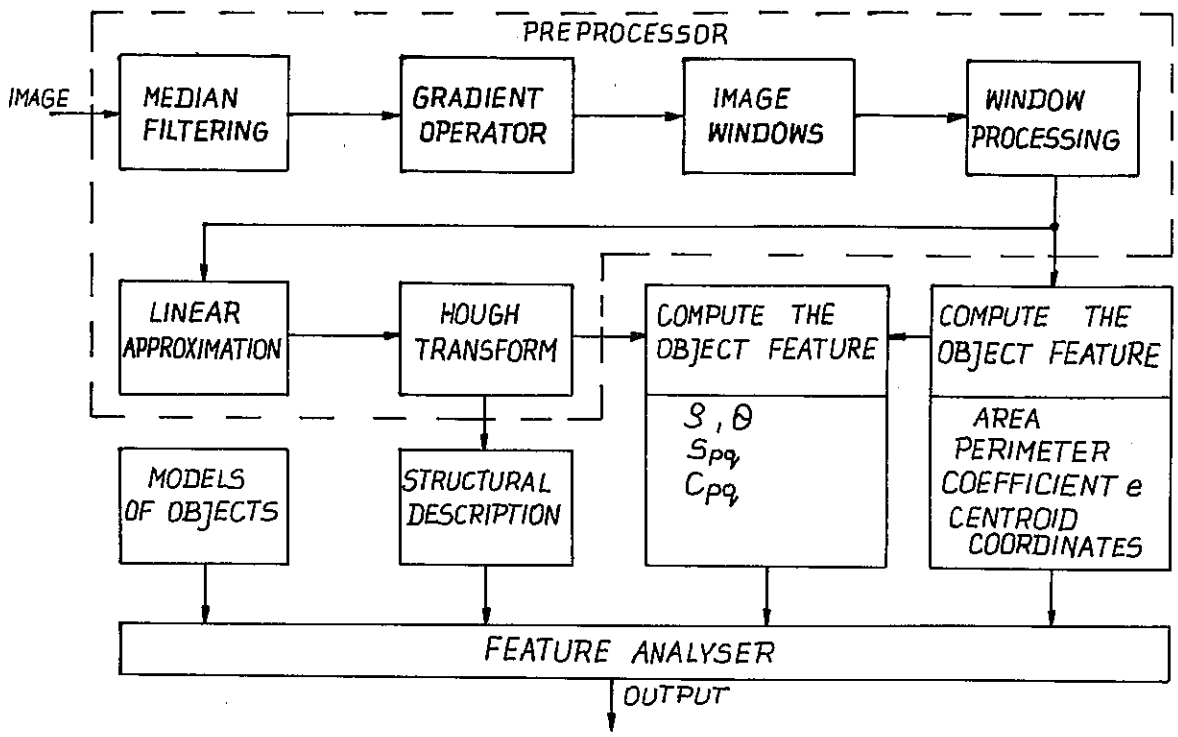


Figure 1

The edges are extracted by an gradient operator :

$$P = \left\{ p(i,j) \;\middle|\; \begin{array}{l} p(i,j) = 0 \quad j = \left\{1, \ldots \frac{H-1}{2}\right\} \\ p(i,j) = 0 \quad j = \left\{\frac{2n-H+1}{2}, \ldots n\right\} \\ p(i,j) = 0 \quad i = \left\{1, \ldots \frac{H-1}{2}\right\} \\ p(i,j) = 0 \quad i = \left\{\frac{2m-H+1}{2}, \ldots m\right\} \\ p(i,j) = \left(2p(i,j) - p(i,j-1) - p(i,j+1)\right) - \left(2p(i,j) - p(i-1,j) - p(i+1,j)\right) \end{array} \right. \quad (3)$$

where P is the image gradient.
A binary image is extracted from gradient field :

$$\begin{array}{l} p(i,j) = 1 \text{ if } p(i,j) \geq tr \\ p(i,j) = 0 \text{ otherwise} \end{array} \quad (4)$$

where tr is a fixed threshold value.

At the same time preprocessor selects points of interest where objects are likely to be located. Interest points are carried out using histogram, and windowing operation is then performed which produces window regions centered about the interest points.

Further, a edge operator (3) is used to compute the edge values of the pixels within the object window. A threshold is applied to keep only the dominant edges within the window. At this stage, the line segments produced may be of two, three or four pixels width, depending on the threshold and spatial distribution of the image feature on the pixel.

The strategy for edge thinning is based on a fast parallel algorithm defined in the following way Zhang[4] :
The contour point p1 is deleted from the digital pattern if it satisfies the following conditions

(a) $2 \leq B(p1) \leq 6$
(b) $A(p1) = 1$
(c) $p2 \cdot p4 \cdot p6 = 0$
(d) $p4 \cdot p6 \cdot p8 = 0$

where A(p1) is the number of 01 patterns in the set eight neighbors of p1 (Figure 2) , and B(p1) is the number of nonzero neighbors of p1 that is

$$B(p1) = p2 + p3 + \ldots + p9 \quad (5)$$

If any conditions is not satisfied, then A(p1) = 2 and p1 is not deleted from the image. Next, conditions (a) and

(b) remain the same and the rest conditions are changed as follows

(c') p2 • p4 • p8 = 0

(d') p2 • p6 • p8 = 0

i-1	p9	p2	p3
i	p8	p1	p4
i+1	p7	p6	p5
	j-1	j	j+1

Figure 2

For the efficient representation of edge lines we used piecewise linear curves. Figure 3 illustrates this concept. If the distance from each point of the edge line to the current piecewise linear segment is less or equal than ε , then this segment approximates the given points. If d_j is greater than ε , then are searching new segments.

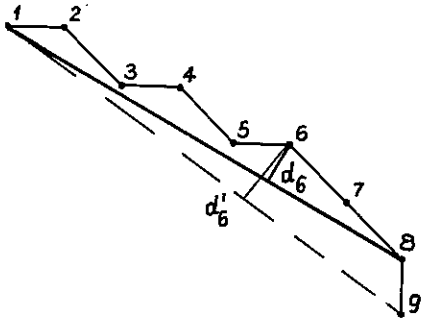


Figure 3

The Hough transform is a linear transform originally developed for line detection in digital pictures. This parametric transform is briefly described next. Referring to Figure 4, a point (i,j) in a two dimensional image space is mapped into a sinusoidal curve in the parameter space (ρ,θ) by

$$\rho = i\cdot\cos\theta + j\cdot\sin\theta \ , \quad 0 \leqslant \theta < \pi \qquad (6)$$

where ρ and θ are the parameters of the normal form of the line on which (i,j) resides. Collinear points in the image space are mapped into curves that share a common point in the paramerer space. Figure 4 depicts the curves for three such collinear points and their common point in the parameter space. In the discrete case, the parameter space is considered to be a two dimensional array of accumulators.

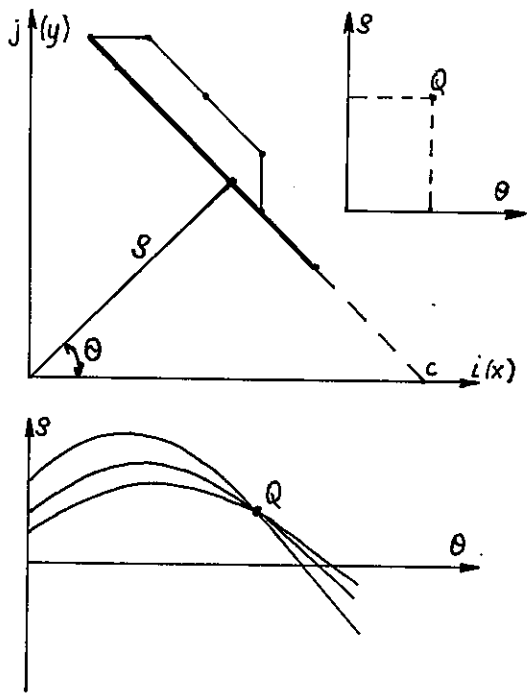


Figure 4

The parameters ρ,θ are defined as follows

$$\rho^2 = \frac{(y_2 x_1 - y_1 x_2)^2}{[(x_2 - x_1)^2 + (y_2 - y_1)^2]}$$

$$\theta = \text{arc tg} \ \frac{(x_2 - x_1)}{(y_2 - y_1)} \qquad (7)$$

3. EXTRACTION OF THE FEATURES

The extraction of the features and their further classification are obligatory steps in the analysis of objects. Here, we formally define and listed several feature functions which can be calculated for each object. These functions are listed below:

(i) Area

$$A = \sum_{i=1}^{m} \delta_i \qquad (8)$$

where $\delta_i = \delta_{i1} + \delta_{i2} + \cdots + \delta_{ir}$

δ_{ir} - number of elements $p(i,j) = 1$ in the r-series,
r - number of series in the i-th lines of image.

(ii) Perimeter

$$Ob = \sum z_{ij} \ , \ (i,j) \in \text{object} \quad (9)$$

where z_{ij} is number of elements $p(\ ,\ )$ which $p(\ ,\ ) \in N_4(i,j)$ and $p(\ ,\ ) \notin$ object.

(iv) Coefficient e

$$e = \frac{\text{max. chord of object}}{\text{min. chord of object}} \quad (10)$$

(iiv) Centroid coordinates

$$X_0 = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} i \cdot p(i,j)}{\sum_{i=1}^{m}\sum_{j=1}^{n} p(i,j)} \quad (11)$$

$$Y_0 = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} j \cdot p(i,j)}{\sum_{i=1}^{m}\sum_{j=1}^{n} p(i,j)}$$

(vi) Moments invariants

$$C_{pq} = \frac{S'_{pq} \cdot \sin(q\theta) + C'_{pq} \cdot \cos(q\theta)}{\sum_{\rho}\sum_{\theta} p(\rho,\theta)} \quad (12)$$

$$S_{pq} = \frac{S'_{pq} \cdot \cos(q\theta) - C'_{pq} \cdot \sin(q\theta)}{\sum_{\rho}\sum_{\theta} p(\rho,\theta)}$$

where

$$S'_{pq} = \sum_{\rho}\sum_{\theta} \rho^p \cdot \sin(q\theta) \cdot p(\rho,\theta) \cdot \rho$$
$$C'_{pq} = \sum_{\rho}\sum_{\theta} \rho^p \cdot \cos(q\theta) \cdot p(\rho,\theta) \cdot \rho \quad (13)$$

$p = 0,1,2\ldots . \ ; \ 0 \leqslant q \leqslant p$

and

$$\Theta = \text{arc tg}\left(S'_{pq} \ / \ C'_{pq}\right) \quad (14)$$

These moments are not affected by changes in object size and orientation.

(vii) (ρ,θ) Parameter .

If $P(v)$ be the v-th feature of the object and $\hat{P}(vm)$ be the v- th feature of the model and $d(v)$ be the accptable distance thus

$$M(v) = \begin{cases} 1 & \text{if } \left|P(v) - \hat{P}(vm)\right| \leqslant d(v) \\ 0 & \text{otherwise.} \end{cases} \quad (15)$$

We define the confidence coefficient C as

$$C = \frac{\sum_{v} M(v) \cdot W(v)}{W(v)} \quad (16)$$

where $W(v)$ is the weight of the v-feature The object is recognized with any model if C = max.

4. CONCLUSIONS

Image Processing System for recognition and analyzing objects has been described along with features which we propose to use for recognizing and analyzing objects in our model. IPS has been developed and tested by computer simulation. IPS may be used in the vision system for industrial robots and targets recognition system

REFERENCES

[1] Choraś,R.S.,Image and Vision Computing 1984 31
[2] Duda,R.O. and Hart,P.E., Pattern Classification and Scene Analysis J.Wiley London,1973 .
[3] Choraś,R.S.,Private Communications
[4] Huang,T.S., Two-Dimensional Digital Signal Processing II Springer, Berlin, 1981 .
[5] Niemann,H., Pattern Analysis Springer Berlin, 1981 .
[6] Rosenfeld,A. and Kak,A.C., Digital Picture Processing Academic, NY, 1982 .
[7] Zhang,T.Y. and Suen,C.Y., Com ACM 1984 236 .

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EXPERT SYSTEMS FOR IMAGE PROCESSING : AN OVERVIEW

Takashi MATSUYAMA

Department of Information Engineering
Faculty of Engineering
Tohoku University
Sendai, Miyagi 980
JAPAN

1. INTRODUCTION

A variety of image processing algorithms have been devised in the history of digital image processing, and a number of programs for image processing have been accumulated in software packages[1]. Although such software packages greatly facilitate the use of digital image processing techniques in various application areas, it is not so easy to make full use of the packages, especially for those with little experience in digital image processing; various knowledge and know-hows are reqired to realize effective image analysis.

The followings are popular problems encountered in developing image analysis programs.

1)Determination of Algorithms and Operators
Since there are many different algorithms (operators) for a specific purpose (e.g. operators for edge detection), one has to select an appropriate one considering image quality, purposes of image analysis, and characteristics of algorithms.

2)Selection of Parameters
Moreover, many algorithms have parameters to be adjusted, so that how to determine appropriate parameters is another problem.

3)Combination of Algorithms and Operators
It is often necessary to combine several fundamental algorithms to perform a meaningful task. For example, a popular way to extract regions from an image is to apply edge detection -> edge linking -> closed boundary detection. To attain effective combination, know-hows about image processing is required.

4)Trial-and-Error Experiments
Usually it is very hard to estimate a priori the performance of an algorithm for a given image, so that one has to repeat trial-and-error experiments by modifying parameters (and sometimes algorithms).

Recently, several expert systems for image processing have been proposed to solve these problems[2-6]. They incorporate Artificial Intelligence techniques to represent and use know-hows about image processing techniques. Since there are many levels of know-hows and purposes of image processing, each expert system has a different objective:

(1)Improve man-machine interface in interactive image processing systems[2]
(2)Automatic generation of image processing programs[3]
(3)Rule-based design system for image segmentation algorithms[4]
(4)Automatic image segmentation experts for object detection[5][6]

Note that (3) is different from the others in the sense that it is an expert system to develop a new algorithm, while the others are for effective use of existing image processing programs.

Expert systems of type (4) play a very important role as low level vision modules in image understanding systems: they are activated by high level object recognition modules to locate missing objects and verify the existence of hypothesized objects. The effectiveness of such top-down goal-directed analysis using a low level vision expert is described in detail in [8].

This paper overviews these expert systems and discusses their future possibilities and problems.

2. CONSULTATION SYSTEM FOR IMAGE PROCESSING

A user of interactive image processing systems is usually reqired to select a command from a large number of varieties and to specify appropriate arguments. Although several HELP facilities are available, one has to refer to user's manuals to see detailed usage of commands. A consultation system for image processing uses such information as knowledge source to help a user select an appropriate command. Since the command and argument specification is done under the system guidance, the man-machine interface of the system can be greatly improved.

In [2] a prototype of such consultation system is described. First a user specifies his purpose (e.g. enhancement, segmentation and so on) and properties of an image to be processed (e.g. color/B&W, noise level, contrast etc.). The system reasons about a processing plan based on the specification given by a user. Then, it selects an appropriate command and its arguments through

conversation on detailed user's objective and image quality. The command and argument selection is controlled by a set of production rules representing various know-hows about image processing.

Processing results are displayed on a monitor step by step, and a user can ask the system to modify command and/or arguments according to his evaluation. Such modification is also controlled by production rules to find good alternatives.

This type of expert systems are very helpful for those with little experience in digital image processing.

3. AUTOMATIC GENERATION OF IMAGE PROCESSING PROGRAMS

Currently, several transportable software packages for image processing are available. For example, SPIDER[1] is a FORTRAN subroutine library for image processing containing over 300 subroutines. Characteristics of program modules (subroutines) in a package, such as data types of arguments of a subroutine, are usually written in a manual. Using such information about program modules as knowledge source, we can construct an "automatic programming system" which fabricates a complex program by combining program modules in a package. A user of the system has only to write an abstract program specification without knowing about the details of the program.

Sakaue and Tamura[3] proposed an automatic program generation system using SPIDER. The system generates a complete (main) program from a given command sequence. Fig. 1(a) shows an input command sequence, which implies that
1)for an image in the standard format(SFDI), G, compute its histogram(HIST1),
2)find a threshold value from the histogram(THDS2),
3)apply binarization using the thresold(SLTH1),
4)apply connected component labeling(CLAB),
5)remove tiny regions(ERSR3),
6)and compute compactness measures for resultant regions(CRCL1).
Each command corresponds to a name of a subroutine in SPIDER.

The system stores information about meanings of each argument of every subroutine in SPIDER,

```
SFDI                    HIST1
   G                     $1 in  GRYN
HIST1                    $2 out HIST
  in G                   =
THDS2                     $1_3
SLTH1                     $2_1
  in G
CLAB
ERSR3
CRCL1
```

(a) command sequence

(b) syntactic and semantic constraints

Fig. 1 Constraints on Subroutine Arguments[3]

such as input/output discrimination, data type, and semantic usage (e.g. image data, histogram, property table etc.). For example, Fig. 1(b) shows the information about HIST1, where the first line reads "the first argument($1) is an input argument and its data type is Gray Picture", and the last three lines imply that the third attribute of the first argument (i.e. the number of gray levels of the input picture) must be equal to the first attribute of the second argument (i.e. the size of an array for the output histogram).

Based on such information, the system determines real arguments for each subroutine, asks a user to specify missing parameters, and generates a complete main program consisting of a set of data declarations and a sequence of subroutine calls.

Although one has to specify which subroutines to use, one need not know their detailed syntactic and semantic structures.

4. RULE-BASED IMAGE SEGMENTATION SYSTEM

Usually, algorithms for image segmentation use many heuristics to split/merge regions and lines into meaningful ones: "merge neighboring small regions with similar properties", and so on. While filtering operators such as smoothing and edge detection can be designed based on well defined mathematics, the incorporation of heuristics into segmentation algorithms is inevitable (at least under the state-of-the-art of image processing). Therefore, to design a segmentation algorithm with high performance, we have to repeat trial-and-error experiments to test the effectiveness of incorporated heuristics.

A rule-based segmentation system was proposed in [4] to facilitate such experiments, where various heuristics for region and line segmentation are represented by a set of production rules (Fig. 2). By representing the heuristics explicitly, one can easily modify them to find really effective ones.

A condition part of a rule is described in terms of constraints on attributes of regions and lines and their mutual spatial relations. An action part specifies a merge/split operation on regions and lines. Besides rules for such segmentation procedures, the system stores a set of meta rules to control the analysis. The mode of the system is switched by a meta rule from region analysis to line analysis and vice versa.

Although the execution time is slow in this rule-based segmentation system, its flexibility enables the fast development of effective segmentation algorithms.

5. AUTOMATIC IMAGE SEGMENTATION SYSTEM

In image understanding of complex scenes,

top-down goal-directed image segmentation is often required to correct errors incurred by initial (bottom-up) segmentation as well as to verify the existence of a hypothesized object[8].

In [5] we proposed an automatic image segmentation expert which extracts image features (such as regions and lines) according to a specification given by a user.

Suppose we want to extract line segments from a gray picture. Gray Picture -(Edge Detection)-> Edge Picture -(Thresholding)-> Edge Point -(Linking)-> Line Segment would be a typical analysis process to satisfy the objective. We call Gray Picture, Edge Picture, Edge Point, and Line Segment in the above example "Image Features", and Edge Detection, Thresholding, and Linking "Transfer Process". That is, an image feature denotes a type of information extractable from a raw image data, and a transfer process analyzes its input image feature to generate its output image feature. Usually, to extract a specific image feature from a raw picture, we have to combine several different transfer processes as shown in the above example. We call such ordered sequence of transfer processes "Process Sequence".

The system uses these image features and transfer processes as fundamental entities to represent the knowledge about image segmentation. Fig. 3 illustrates the network structure used by the system, where each ellipse denotes an image feature and a directed arc a transfer process.

Fig. 4 shows an example of the processing by the system. The goal specification given to the system is as follows(Fig. 4(b)):
Find rectangle(s) in the image shown in Fig. 4(a) whose area size is in between 100 and 400 and which is located in the window (upper-left and lower-right corners : (165,89) and (204,115)).

The system first reasons about which process sequence is best to extract a rectangle. This reasoning is done by searching for the most promissing path in the network connecting Gray Picture and Rectangle. The search process is guided by production rules associated with each node and arc in the network. The rules represent knowledge to determine which path is promissing under a given goal and properties of an image to be processed. In this example, the process sequence shown in Fig. 4(c) was selected.

Then, the system activates the transfer processes included in the selected sequence one by one (Fig. 4(d)). In this example, however, the transfer process POLYGON-TO-RECTANGLE has failed. This implies that the extracted polygon cannot be considered as a rectangle. In case of such failure, the system activates "failure rules" associated with a failed transfer process, which suggest alternative analysis processes. In the example, the failure rule associated with POLYGON-TO-RECTANGLE suggested that retry the same process sequence by changing a threshold value for binarization(Fig. 4(e)). By this modification, a rectangle satisfying the goal specification was successfully extracted(Fig. 4(f)).

This expert system was originally developed as a low level vision expert (LLVE) in an image understanding system SIGMA[8], where LLVE is activated by an object recognition module to verify the existence of hypothesized objects. The incorporation of LLVE enables clear separation between domain dependent knowledge for object recognition and domain independent knowledge for image processing.

6. DISCUSSION

Four types of expert systems for image

IF: (1) The REGION SIZE is HIGH
(2) The REGION AVERAGE GRADIENT is NOT LOW
(3) The REGION HISTOGRAM is BIMODAL
THEN: (1) SPLIT the REGION according to the HISTOGRAM

METARULE
IF: (1) Previous PROCESS was REGIONS
(2) Previous PROCESS was ACTIVE
THEN: (1) Match the REGION analysis rules.

Fig. 2 Production Rules for Segmentation[4]

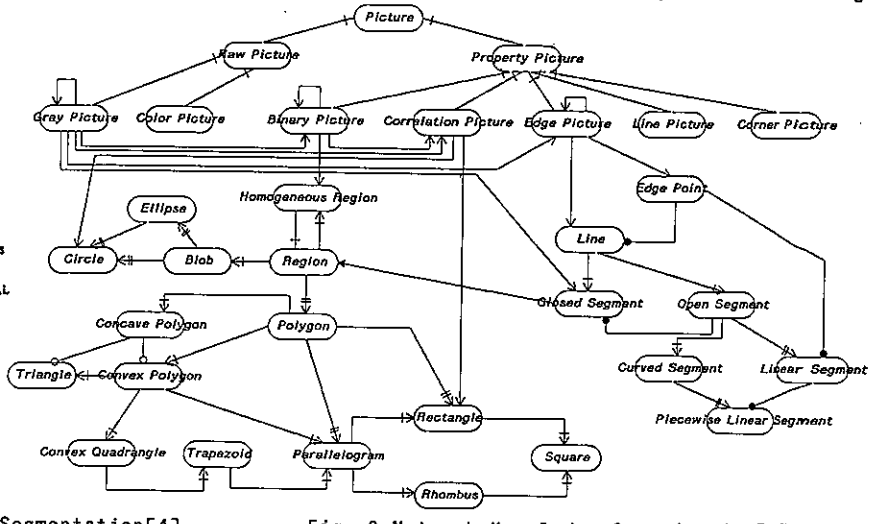


Fig. 3 Network Knowledge Organization[5]

processing have been surveyed. They are very useful to organize image processing algorithms and heuristics in a unified system as well as to develop versatile image processing and understanding systems. However, the systems described in this paper are first prototypes, and have many problems to be solved:

(1)Description of Image Quality and Knowledge

All systems use production rules to represent heuristics and know-hows about image processing techniques. However, vocabularies used to write the rules are very limited and it is very hard to verbally describe image quality, shape features and spatial relations.

(2)Knowledge about Analysis Strategies

There have been proposed many analysis strategies to increase the performace of image processing: processing based on pyramid data structures, and combination of edge-based and region-based analyses. How to incorporate such analysis strategies into expert systems is a future problem.

As for problem (1), Hasegawa et al[6] proposed an automatic segmentation system, where a goal specification is given by a sample picture representing regions and lines to be extracted rather than by a symbolic word such as rectangle. This allows the system to extract complex image features and to evaluate processing results accurately by calculating their similarity to the sample picture.

As for more flexible combination of fundamental algorithms, we proposed an interactive image processing system with versatile algorithm combinations[7]. For example, a command

OPTIMIZE [CONTOUR(BINARY(DATA1,*)),40,60,2 by BINARY(GRAD(DATA1),3) at EFUNC]

implies

1)first apply edge detection (GRAD) to image DATA1, binarize the result by threshold 3, and make this a reference picture, 2)then apply binarization to DATA1 by changing threshold values from 40 to 60 by 2, 3)finally select one from the binarized pictures which is most consistent with the reference picture obtained by 1). The consistency is evaluated by function EFUC, which enumerate the number of edge points in the reference picture coincident with pixels on region boundaries in each binarized picture.

Command OPTIMIZE is a meta command to combine fundamental commands and enables a user to incorporate a parameter optimization scheme into image processing.

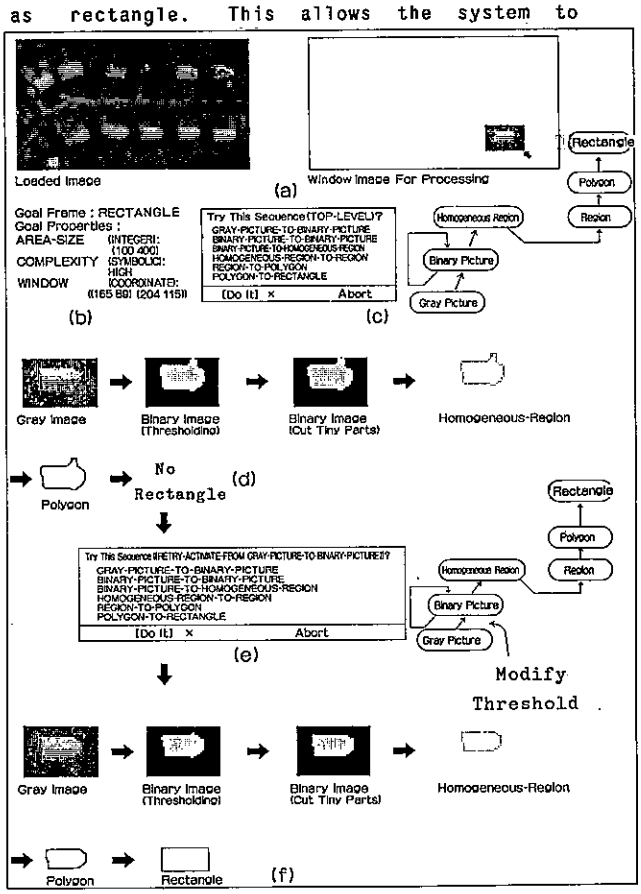


Fig. 4 Automatic Extraction of Rectangle[5]

ACKNOWLEDGEMENT

The author would like to appreciate helpful comments and discussions by Prof. M. Nagao of Kyoto University and Prof. T. Ito of Tohoku University.

[REFERENCE]

[1]H.Tamura et al: Design and Implementation of SPIDER, CVGIP, Vol.23, pp.273-294, 1983

[2]N.Sueda: An Expert System for Image Processing, Image Technology and Information Display, Vol.17, No.9, pp.19-22, 1985(in Japanese)

[3]K.Sakaue and H.Tamura: Automatic Generation of Image Processing Programs, Proc. of CVPR, pp.189-192, 1985.

[4]A.M.Nadif and M.D.Levine: Low Level Image Segmentation:An Expert System, IEEE Trans., Vol.PAMI-6, No.5, pp.555-577, 1984

[5]T.Matsuyama and M.Ozaki: LLVE:An Expert System for Top-Down Image Segmentation, Journal of IPS Japan, Vol.27, pp.191-204, 1986(in Japanese)

[6]J.Hasegawa et al: Automatic Construction of Image Processing Procedures by Sample-Figure-Presentation, Tech. Rep. of the Professional Group of IECE Japan,PRL85-38, 1985(in Japanese)

[7]N. Murayama, T.Matsuyama and T.Ito: Combination of Image Processing Operators, National Convention Record of IECE Japan, 1236, March 1986(in Japanese)

[8]T.Matsuyama and V.Hwang: SIGMA:A Framework for Image Understanding, Proc. of 9th IJCAI, pp.908-915, 1985

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

STATISTICAL DETERMINATION OF THE SPATIAL QUANTIZATION ERROR IN SAMPLED CONTOURS*

Reinhard Janssen

AEG Aktiengesellschaft, Research Center
7900 Ulm, Federal Republic of Germany

Many segmentation problems in image analysis can be solved by uniform thresholding. The contours separating points above and below the threshold will undergo a spatial quantization if the image is digitized. The paper describes the quantization error in terms of image statistics. The image function is modeled as a two-dimensional random process. Linear approximations of the contours as well as the image function yield relations between the sampling error (area) and parameters of the random process.

1 INTRODUCTION

The quantization of the image $z(x,y)$ in both luminance and coordinates is a prerequisite of any digital image processing. In many practical instances, a bilevel quantization (1) of the function values will be adequate for image analysis purposes.

$$b(x,y) = \begin{cases} 1 & \text{if } z(x,y) \geq z_0 \\ 0 & \text{if } z(x,y) < z_0 \end{cases} \qquad (1)$$

Superimposing a square grid on the x-y plane we obtain a matrix of luminance samples. Each pixel $z(x_i,y_k)$, or $b(x_i,y_k)$ respectively, represents a square domain D_{ik}. Due to the sampling, the contours ($z=z_0$), which separate "background" (b=0) and "foreground" (b=1), undergo a spatial quantization. If no a-priori information on the function $z(x,y)$ is available, the quantized contour is assumed to consist of edges of the square domains.

The Nyquist sampling theorem describes the error encountered when reconstructing $z(x,y)$ from its samples $z(x_i,y_k)$. However there is no equivalent rule for sampling binary images (Otterloo and Gerbrands [1]) as only the difference between the quantized contour and underlying continuous contour is of interest. The appropiate measure for this error depends on the application (Proffitt and Rosen [2]). Henceforth we consider as quantization error the area enclosed by the two curves.

The picture (Fig. 1) and its thresholded version (Fig. 2) are considered quasi-continuous in space. Under-sampling and zero-th order interpolation lead to Fig. 3. The difference between Fig. 2 and 3 illustrates the quantization error in question (Fig. 4).

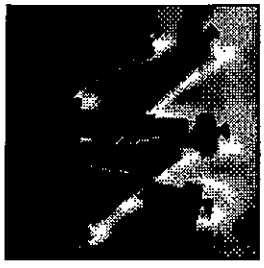


Figure 1

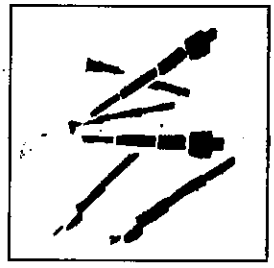


Figure 2

Figure 3

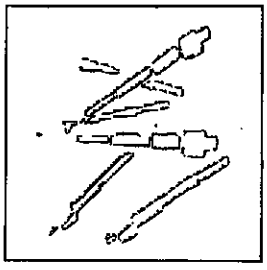


Figure 4

The paper describes two procedures to obtain relations between the contour quantization error and the image statistics. While the sampling theorem gives the overall "energy" of the reconstruction error, the considered quantization error only depends on the image function's behaviour close to the contours. Therefore we apply models for the image function and contour for those sampling domains which represent boundary pixels in the digitized bilevel picture. First we summarize how Panda [3] expressed the expected contour length in terms of image statistics. A linear approximation of the contour then yields a relation between expected contour length and quantization error. In another approach we use a bilinear approximation of $z(x,y)$ in the neighbourhood of contour pixels in order to compute the error directly from the image statistics.

*This work was supported by DFG (Germany) under contract Ha624.

2 ESTIMATION OF CONTOUR LENGTH

Panda [3] assumes that the continuous image function $z(x,y)$ is a stationary 2-D random process which can be represented as a trigonometric polynomial

$$z(x,y) = \sum_i a_i \cos(k_i x + l_i y + \varphi_i), \tag{2}$$

where k_i and l_i are the horizontal and vertical wave numbers, a_i the random amplitude, and φ_i the random phase, uniformly distributed in $[0,2\pi)$.

Under these assumptions the expected density of contour points $(z=z_0)$ per unit area can be calculated exactly for the isotropic case. In general, $E(l)$ is found to be approximately

$$E(l) \approx \frac{A}{\pi}\sqrt{\frac{m_z(2,0)+m_z(0,2)}{m_z(0,0)}}\exp[-\frac{z_0-E(z)}{2m_z(0,0)}], \tag{3}$$

where A is the area covered by the image, and $E(z)$ is the mean of the marginal pdf. For real processes the moments $m_z(i,k)$ of the spectral density function can be expressed by partial derivatives of $z(x,y)$.

$$(-1)^{\frac{d-c}{2}} m_z(a+b,c+d-a-b) = E\left(\frac{\partial^c z}{\partial x^a \partial y^{c-a}} \cdot \frac{\partial^d z}{\partial x^b \partial y^{d-b}}\right)$$

3 ESTIMATION OF QUANTIZATION ERROR

3.1 *Linear Contour Approximation*

Figure 5 shows the square domain of one pixel. The sampling interval T is normalized to unit.

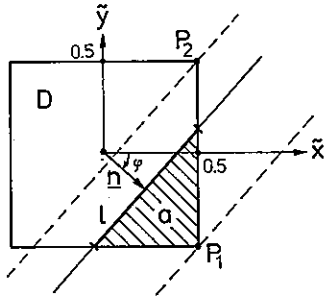


Figure 5

We assume that the contour only once intersects the pixel domain. This contour segment is approximated by a straight line with normal vector $\underline{n}$.

$$\tilde{x}\cos\varphi + \tilde{y}\sin\varphi = n \tag{5}$$

With $0 \le \varphi < \pi/4$ the line passes the corner points P_1 and P_2 if

$$n_1 = \frac{1}{2}(\cos\varphi - \sin\varphi), \tag{6}$$

$$n_2 = \frac{1}{2}(\cos\varphi + \sin\varphi). \tag{7}$$

The contour length l and quantization error a (hashed region) related to that pixel are

$$l(n,\varphi) = \begin{cases} \dfrac{1}{\cos\varphi} & 0 \le n \le n_1 \\ \dfrac{n_2-n}{\sin\varphi\cos\varphi} & n_1 \le n \le n_2 \end{cases} \text{ if }, \tag{8}$$

$$a(n,\varphi) = \begin{cases} \dfrac{1}{2}\tan\varphi + \dfrac{n_1-n}{\cos\varphi} & 0 \le n \le n_1 \\ \dfrac{(n_2-n)^2}{2\sin\varphi\cos\varphi} & n_1 \le n \le n_2 \end{cases} \text{ if }. \tag{9}$$

If all values of n are equally probable (compare with [2]), the expectations of l and a, assuming event C that the line intersects the pixel square, are calculated by integrating Eq. (8) and (9) over all normals and angles.

$$E(l|C) = \frac{\pi}{4} \tag{10}$$

$$E(a|C) = \frac{1}{12}\left(\sqrt{2} + \ln\tan\frac{3}{8}\pi\right) \tag{11}$$

The unknown probability of C can be eliminated from the total expectations

$$E(l) = E(l|C)\,P(C) + E(l|\bar{C})\,[1-P(C)], \tag{12}$$

$$E(a) = E(a|C)\,P(C) + E(a|\bar{C})\,[1-P(C)], \tag{13}$$

$$E(l|\bar{C}) = E(a|\bar{C}) = 0, \tag{14}$$

yielding a relation between the quantization error and contour length per unit area.

$$E(a) = \frac{1}{3\pi}\left(\sqrt{2} + \ln\tan\frac{3}{8}\pi\right) E(l) \tag{15}$$

With respect to the normalization, the expected total area A is proportional to the product of the expected total contour length and the sampling interval T.

$$E(A) \sim E(L)\cdot T \tag{16}$$

3.2 *Bilinear Approximation of Image Function*

More accurate results can be achieved by modeling the image function $z(x,y)$ rather than the contour path. The most simple approximation of $z(x,y)$ in the vicinity of x_{ik} is linear in x and y:

$$\hat{z}(x,y) = [n\ m_x\ m_y] \left[\begin{bmatrix} 1 \\ x \\ y \end{bmatrix} - \begin{bmatrix} 0 \\ x_i \\ y_k \end{bmatrix} \right] = \underline{k}'(\underline{x}-\underline{x}_{ik}) \quad (17)$$

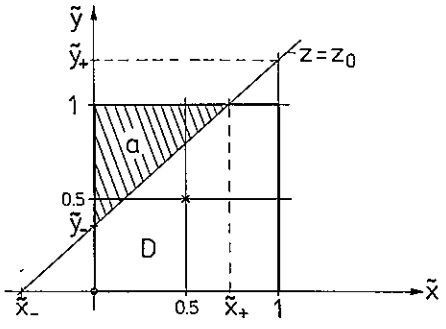


Figure 6

Figure 6 again shows a normalized square pixel D. The contour ($z=z_0$) intersects the grid lines at

$$\tilde{x}_{\pm} = \frac{1}{m_x}(z_0 - n \pm \tfrac{1}{2} m_y) + \tfrac{1}{2} \quad \text{if } m_x \neq 0, \quad (18)$$

$$\tilde{y}_{\pm} = \frac{1}{m_y}(z_0 - n \pm \tfrac{1}{2} m_x) + \tfrac{1}{2} \quad \text{if } m_y \neq 0. \quad (19)$$

The hashed area a in figure 6 approximates the quantization error. Function $a(\underline{k})$ holds the symmetry relations

$$a(n-z_0, m_x, m_y) = a(|n-z_0|, |m_x|, |m_y|), \quad (20)$$

$$a(n, m_x, m_y) = a(n, m_y, m_x). \quad (21)$$

Thus the calculation of $a(\underline{k})$ can be confined to those $\underline{k}$ where $n-z_0, m_x, m_y \geq 0$ and $m_x \geq m_y$. For the remaining $\underline{k}$ we distinguish the cases F0-F2 (Fig. 7).

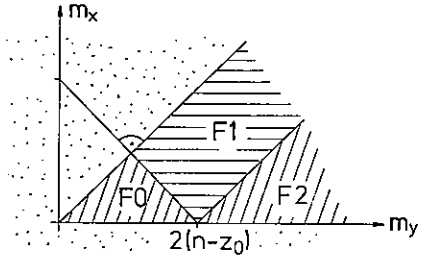


Figure 7

case	line g intersects ...	error area
F0	no edges of D	$a = 0$
F1	opposite edges of D	$a = \frac{1}{2}(\tilde{x}_- + \tilde{x}_+)$
F2	abutting edges of D	$a = \frac{1}{2}\tilde{x}_- \cdot \tilde{y}_+$

We derive the model parameter $\underline{k}_{ik}$ for each sampling domain D_{ik} by minimizing the functional

$$J(\underline{k}_{ik}) = \iint_{D_{ik}} |z(x,y) - \hat{z}(x,y)|^2 \, dxdy. \quad (22)$$

The expected quantization error can be calculated from the pdf $p(\underline{k})$.

$$E(a) = \iiint a(\underline{k})\, p(\underline{k})\, d\underline{k}. \quad (23)$$

If the image function is modeled as a stationary 2-D random process, the parameters $\underline{k}$ will be normally distributed. For the isotropic case we conclude that E(a) can be approximately expressed by a polynomial of the variances of $\underline{k}$ if the image is thresholded with $z_0 = E(n)$.

$$E(a) \approx \sum_{i+k+l \leq 1} c_{ikl}\, \sigma_n^i\, \sigma_{m_x}^k\, \sigma_{m_y}^l. \quad (24)$$

4 EXPERIMENTS

Finally we compare some results with data measured from digital test pictures. Fig. 8 shows an industrial scene. The corresponding contours for $z_0 = E(z)$ are represented by Fig. 9. The second test image (Fig. 10) is a computer-generated white noise process [5], having a normal marginal pdf. Various normal processes can be derived from the noise signal by linear filtering. Figure 11 results from applying a recursive low-pass filter whose Z-transform has double poles at 0.785. Figures 12 and 13 show power spectrum and contour picture belonging to Fig. 11. The images are 256 X 256 pixels in size. The luminance scale has 256 distinct grey levels.

Figure 8

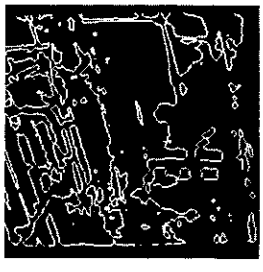


Figure 9

The first test image (Fig. 8) is thresholded at several levels. Sampling the image again, we obtain a matrix of N_S X N_S samples. Figure 14 shows three quantities plotted against the frequency p_0 of "background" pixels, where p_0 is a monotonic function of the threshold level. The number A_0 of pixels in which the original and the under-sampled image differ, is a *measurement* of the quantization error. From Eq. (15) we derive the *estimates* A_1 and A_2.

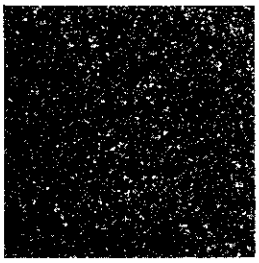


Figure 10

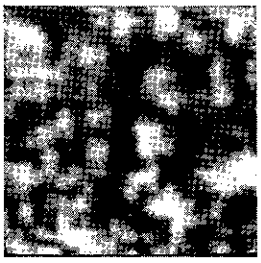


Figure 11

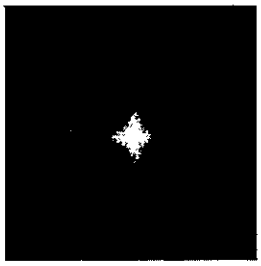


Figure 12

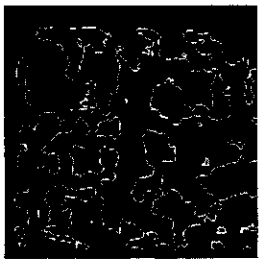


Figure 13

In case of A_1, the contour length is measured according to [2] while Panda's results are used to calculate E(l) in case of A_2. The moments of the spectral density are estimated by substituting spatial averages for ensemble averages in Eq. (4). Since A_0 and A_1 agree well with each other, the linear contour approximation seems to be a reasonable approach. The estimate A_2 will perform well only if the picture meets the assumptions made in Section 2.

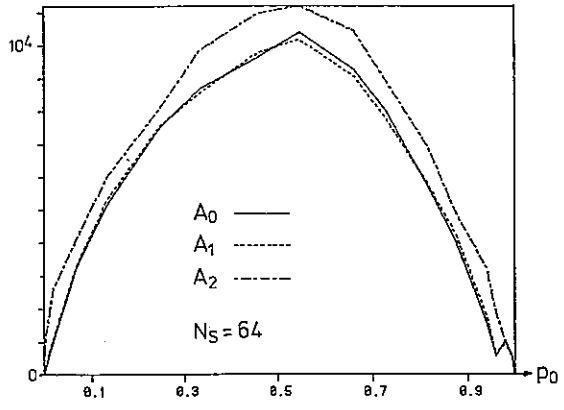


Figure 14

Figures 15 and 16 show measurements and predictions of the quantization error according to Section 3.2 for various under-sampling rates (N_S=8,16,32,64,128). A_1 is the sum of $a(\underline{k}_{ik})$. We obtain A_2 from Eq. (23), where $p(\underline{k})$ is a normal pdf fitted on the measured distribution of $\underline{k}$. According to Eq. (24), A_3 is calculated only from the variances of the three model parameters, i.e. mean luminance, vertical and horizontal slope within pixel domains.

The results for the test images assist the assumption of a normal pdf for $p(\underline{k})$ as well as the applicability of Eq. (24). The measured error and the three estimates considerably differ from each other for low values of N_S because the bilinear approximation becomes inadequate. As was to be expected, A_2 is closer to A_1 for the pseudo-random image (Fig. 16) than it is for the industrial scene (Fig. 15). Nevertheless, modeling the image function yields more accurate results than modeling the contours.

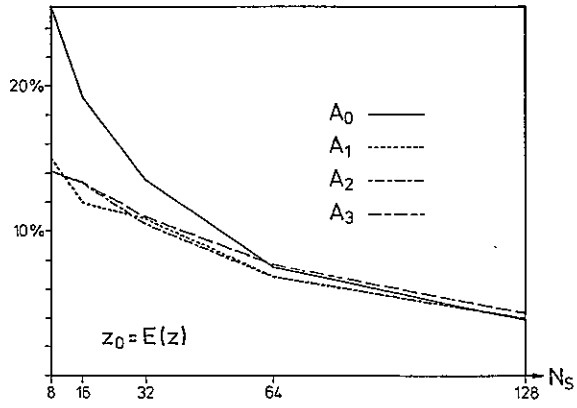


Figure 15

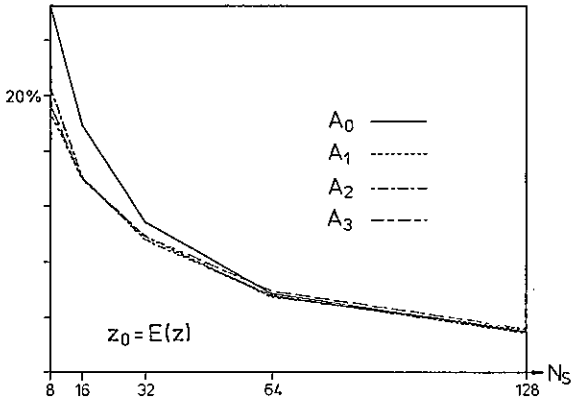


Figure 16

REFERENCES

[1] Otterloo, P.J. and Gerbrands, J.J., Information & Control *39* (1978), pp. 87.
[2] Proffitt, D. and Rosen, D., Computer Graphics & Image Processing *10* (1979), pp. 318.
[3] Panda, D.P., CG & IP *8* (1978), pp. 334.
[4] Ellis, T.J. and Proffitt, D., CG & IP *10* (1979), pp. 333.
[5] Franklin, J.N., Mathematics Computation *17* (1963), pp. 28.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

M.10

EDGE FILTERING IN IMAGE SYNTHESIS WITH Z-BUFFER METHOD

A. BRUNO
I.N.S.A.
Laboratoire d'Automatique
20, Av. des Buttes de Coësmes
35043 RENNES Cédex - FRANCE

D. BARBA
IRESTE
3 Rue du Maréchal Joffre
44041 NANTES Cédex 01 - FRANCE

This paper deals with a 2-D adaptive filtering technique specially developped to remove aliasing artifacts at edges in computer generated images. This edge filtering is designed in the context of image synthesis with depth-buffer method and image facet model. Filtering of facet edges is realized by technique called "pixel integration" and is performed only when all the polygons are writting in the frame memory, just at the time of displaying. This technique is independant of the facet sorting order, gives good results and is very cost effective.

1. INTRODUCTION

We propose here a 2-D adaptive filtering technique specially developped to remove aliasing artifacts which occur at edges. This edge filtering is designed in the context of image synthesis with depth buffer method and image facet model. The scene modelling is based upon objects described by a set of plane polygons called facets on which some optic parameters (reflectance coefficients, ...) and aspect parameters (textures, ...) are defined. This gives after image formation (projection of the scene into the image plane) a two component image modelling. The first component called "plateau component" contains informations about local mean values and abrupt changes of these local mean values (edges). This component is supposed to be locally constant or involving simple (linear) variations inside each facet. The other component or "texture component" represent the local variations of the image signal apart from its mean value. This component is obtained by mapping on the image facet with some memoryless transformation the content of one specific look-up table made with a normalized digital texture or by using a mathematical model for computing on line the texture component. The aliasing defects of the texture signal need the use of some filtering methods well fitted to the texture model, but they will not be discussed here. We will focuse our discussion on the edge filtering.

The aliasing effects come from subsampling in the digital processing and displaying while the image signal may have high activity in some areas : not only at edges but in some textured areas too. Anti-aliasing methods can be divided into two groups. In the first one, the principle used is to increase the spatial sampling density of the computered pixels. Aliasing effects vanish at the price of a higher number of arithmetic operations (quadratic law). In the second groupe, one reduces the signal bandwidth by using some low-pass filtering technique before displaying. CROW and CATMULL |1|, |2| have shown that the last techniques are more interesting in term of computer time saving than oversampling technique at the same image quality. The spread of the finite impulse response filters is very small (1x1 or 2x2 pixels) and the impulse responses are usually taken uniform. Others responses such as gaussian or pyramidal ones have been tested. They give better performances in term of visual quality but are computationaly more complex to use than the uniform ones.

Another type of problems is connected to the algorithm used in the removal of the hidden objects in image synthesis and interacts with the antialiasing techniques.

In order to properly compute the color at a pixel, every visiblefacet which intersects the spatial domain attached to this pixel must be taken into account. In the initial version of image synthesis with depth-buffer, the informations are inadequate for determining wether two facets interacting with one pixel are contignous each other or not. BLOMENTHAL |3| solved this problem in part by memorizing supplementary informations at every pixel (the number of the facet). This method is not a general one yet, moreover some defects go on to exist. The only way to get exactly the color at one pixel is to compute it only once at all. We present here after the basic principle of the antialiasing processing for only one polygonal facet. Generalization of the entire algorithm will be discussed after.

2. ONE FACET EDGE FILTERING

In its principle the technique is very simple. Filtering of facet edges is realized by technique called "pixel integration". For sake of simplicity, we shall make the following assumptions :

- the sampling grid in the image plane is orthonormal,
- the finite impulse response of the antialia-

sing filtering $F_{m,n}$ associated with each sample $P_{m,n}$ is constant over its support : a square with unit lenght side and centered at this sample.

Then, the coefficient λ used in the filtering of the grey level at one particular pixel is directly given by the intersection area between the support of the corresponding filter $F_{m,n}$ and the projection of the facet in the image plane.

2.1. Notations

Let Q_k be one convex polygonal plane facet displayed in the image plane. The rows of the image correspond to the Y axis and the columns to the X axis (figure 1).

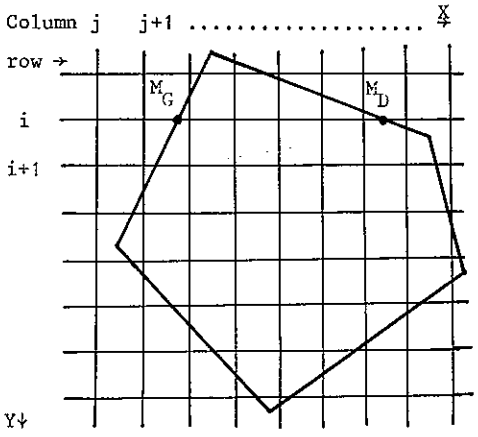


Figure 1
Projection of one facet in the image plane

Let be :
IY : number of the pending row
XG : abscissa of the left hand edge facet at row IY
XD : abscissa of the right hand edge facet at row IY
ZG : depth of the left hand point M_G (XG, IY)
ZD : depth of the right hand point M_D (XD, IY)
OG : grey level at point M_G
OD : grey level at point M_D
DXG : variation of the abscissa XG between two successive rows
DXD : variation of the abscissa XD between two successive rows
DZG : variation of the depth ZG between two successive rows
DZD : variation of the depth ZD between two successive rows
DOG : variation of the grey level OG between two successive rows
DOD : variation of the grey level OD between two successive rows

Without any antialiasing filtering, the grey level and depth at point M with integer abscissa IX are given by relations (1-a) and (1-b)

$$O(IX) = OG + (IX-XG)(OD-OG)/(XD-XG) \qquad (1\text{-a})$$

$$Z(IX) = ZG + (IX-XG)(ZD-ZG)/(XD-XG) \qquad (1\text{-b})$$

2.2. Left-hand edge filtering

Let us assume that no vertex of the polygon is present on the pending row at the left-hand edge.According to the fact that wether $|DXG|>1$ or not, we get two sets of formula for computing the left hand coefficient λ_G of that edge filtering.

Let be : XGD = XG - |DXG/2|
XGF = XG + |DXG/2|
DMUG = |1/DXG|
IXGD = [XGD + 0.5]
IXGF = [XGF + 0.5]

where [.] is the integer part of .

2.2.1. $|DXG| > 1$

Figure 2 shows the set of pixels which must be filtered

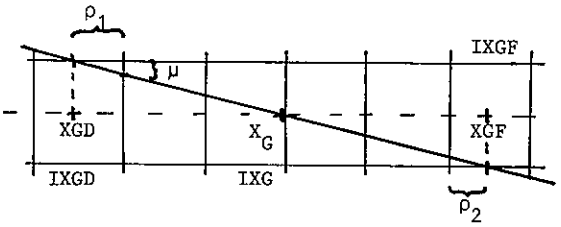


Figure 2 - Left-hand edge filtering

If we define : ρ_1 = IXGD + 0.5 - XGD
ρ_2 = XGF - (IXGF - 0.5)
μ = ρ_1 . DMUG

every pixel of abscissa I between IXGD and IXGF must be filtering respectively with the coefficients λ_G.

If : I = IXGD, then $\lambda_G(IXGD) = (\rho_1.\mu)/2$

$I \in [IXGD+1, IXGF-1]$, then $\lambda_G(I)=\lambda_G(I-1)+DMUG$

I=IXGF, then $\lambda_G(IXGF)=1-(\rho_2^2.DMUG)/2$

2.2.2. $|DXG| < 1$

The left-hand edge crosses only one or two pixels. One gets the following relations depending on the fact that wether IXGF is different of IXGD or not (figure 3).

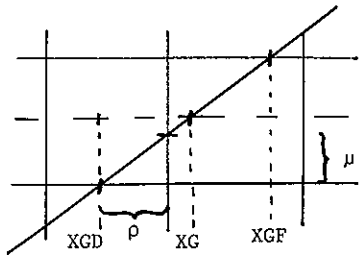


Figure 3

If IXGF ≠ IXGD

We define : $\rho = IXGD + 0.5 - XGD$
$\mu = \rho.DMUG$

For : I = IXGD, then $\lambda_G(IXGD) = (\rho.\mu)/2$
I = IXGF, then $\lambda_G(IXGF) = (IXGF+0.5-XG) - \lambda_G(IXGD)$

If : IXGF = IXGD, then $\lambda_G(IXGD) = IXG+0.5-XG$

2.3. General case

The design of the right hand edge filtering is quite similar to the left hand side. The entire processing is performed both at the same time but with the following precautions. A priori, left hand and right hand edges filtering are independant for every pixel which doesn't belong both to the two spatial filtering areas. In the contrary, one needs to take into account these fact as it is shown on figure 4.

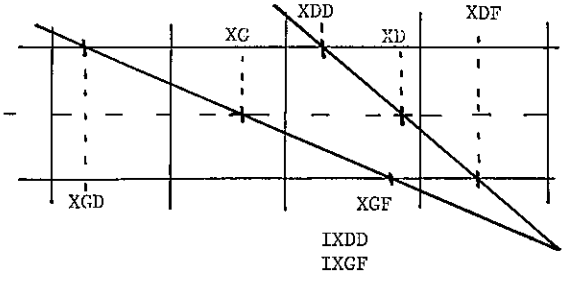


Figure 4 - Edge filtering overlapping area

The overlapping zone corresponds to the segment [IXDD, IXGF] with IXDD≤IXGF. For these pixels, the intersection area between the support of the filter and the facet is given by the relation

$$\lambda = \lambda_G + \lambda_D - 1$$

We notice that this formula is always true, what ever the position of the pixel is, because :

$\forall I \in [IXGF + 1, IXDF]$, then $\lambda_G(I) = 1$

$\forall I \in [IXGD, IXDD-1]$, then $\lambda_D(I) = 1$

and finally $\forall I \in [IXGD, IXGF]$, then

$$\lambda(I) = \lambda_G(I) + \lambda_D(I) - 1$$

Moreover, as the computation of the two parameters λ_G and λ_D are independant, they can be calculated by two different processors working in parallel. In the next paragraph, we shall show that in practice no multiplication are needed to compute the filtering coefficients λ. Before show it, an adaptation of this filtering technique must be used to filter correctly the first and last rows of one facet. We modify the ends of the left hand and right hand filtering zones according the sign of the edge slopes in order to avoid excessive extensions. For example, at the first row, we limit the left hand filtering zone to begin at IXG abscissa if the left edge slope is positive.

2.4. Filtering determination by look-up table

In order to compute the different filtering coefficients λ_D and λ_D, one generally needs two real multiplications besides others additions or substractions. Owing to the fact that the required precision isn't important, it is possible to quantize them very coarsely. More precisely, as the coefficients λ always correspond to the area of a triangle or a trapezium, it is possible to get their area by reading a predefined look-up table with two input addresses. Figure 5 shows the trapezium case. Knowing the two quantizing values μ_i and μ_{i+1}, one get the area of the corresponding trapezium by only one reading cycle of ROM.

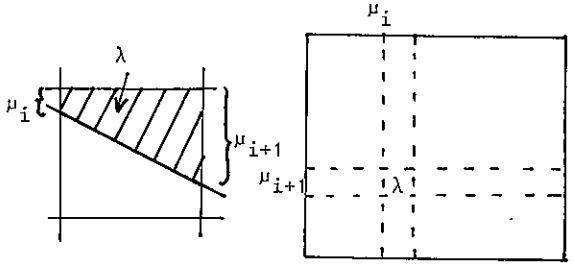


Figure 5
Area of a trapezium by look-up table reading

The different formula show that the filtering coefficient λ is given by one out of the four following operations :

$$\lambda = \rho.\mu/2$$
$$\lambda = (\mu_1 + \mu_2)/2$$
$$\lambda = 1 - \rho.\mu/2$$
$$\lambda = 1 - (\mu_1 + \mu_2)/2$$

and the maximum speed is got by using four different look-up tables. Some tests have been made with 4,8 and 16 quantization levels of μ and ρ. In normal viewing conditions, we noticed some visual quality difference between 4 and 8 levels but almost no difference between 8 and 16

levels. So only four 64-words ROM are necessary to compute λ_G and λ_D from ρ and μ as we can see at the image on figure 6. For comparison, we displayed the same image without edge filtering on figure 7.

3. IMAGE SYNTHESIS WITH DEPTH-BUFFER AND EDGE FILTERING

The basic principle of image synthesis with depth-buffer method is as the following. For every pixel, one memorizes the depth z between the observer and the corresponding point on the polygon (facet) (in the viewing space). The display of a new point of a facet is depending of the result of the test of its depth with the one with the same picture coordinates which has been previously memorized into the depth-memory. If the new point is closer to the observer, its color informations and depth must be memorized. It doesn't in the opposite case.

This procedure remains the same for every pixels completely interior to the facet. It must be modified for the pixels belonging to edge filtering zones. In that case, in order to compute exactly the color, one needs to know the intersection zone for every facet, their color and depth. That can be done only by setting a list at every pixel which memorizes :

- the color
- the shape of the intersection (not only the coefficient λ)
- the depth

A depth sorting algorithm allows to classify the facets according their distance, close to far, and to compute recursively the coefficient λ. That method gives good results in simulation but is not well suited for real-time image synthesis. Moreover, implementation of this technique in hardware isn't easy because of the need of dynamical memory allocation of the lists. Therefore, we have kept one simpler method which have nothing of these drawbacks.

This method lies in taking into account only two facets to compute the color at edges : first, the one which covers the center of the pixel and second, the adjacent one which is of minimal depth and in the perpendicular direction with the edge. When 3 facets cover in part the pixel both at the same time, we make some errors which is not really annoying. In fact, if the error isn't small then the difference of color (luminance and hue) between two facets is high and produces one important reduction in error visibility (masking effect in the visual system).

In order to perform this algorithm, the needed informations for every pixel P to be filtered are :

- the color at one pixel $P' \in V(P)$ when V(.) is the 8- neighbourhood of
- the filtering coefficient at P

These informations are got in 3 distinct steps. The first step is to compute for every pixel P from a facet :

- the color
- the depth
- the filtering coefficient λ (as described before)
- one indicator (JTYP) showing wether P belong to the internal or to the external edge filtering zone
- one indicator (IPIX) pointing out the associated pixel P'.

From P, pixel P' is in the perpendicular direction with the edge and if P is an internal edge filtering pixel, P' is an external one and conversely.

Depending of the result of the tests of old (depth memory) and new depth at pixel P and P', in the second step we modify completely, or only in part, or not at all the content of the memories.

Finally, when all the facets have been processed by steps 1 and 2, the last step makes out edge filtering by using both the colour informations at pixel P and P' and the filtering coefficient. Pixel P' is pointed out by the indicator IPIX associated to P. If IPIX equal zero, then $P' \equiv P$ and the colour of P isn't changed. In the opposite case, the modified colour information is given by :

$$C(P) = \lambda(P)\, C(P) + (1-\lambda(P))\, C(P')$$

This technique gives good results as we could have seen on the image shown figure 6.

4. CONCLUSION

Edge filtering is needed only for pixels where two or more facets are simultaneously present. In general, the colour obtained is dependant of the facet presentation order and important defects sometimes exist. The principle of the method we proposed is to memorize at every pixel the response of the pixel filtering, the colour value and the depth for every facet covering partially the pixel. Then, we have all the necessary information for an exact computation of the colour. We show, however, that the memory cost is important and can be drastically reduced if we memorize the information only for the two facets which filtering responses are maximum. Filtered and non-filtered images show that this balance is acceptable for most applications and is quite performant and cost effective for real time image synthesis with special processors.

ACKNOWLEDGMENT : this research was supported by the Centre Commun d'Etudes de Télédiffusion et Télécommunications under contract C 1138 W

REFERENCES

|1| FC GROW
"Filtering in hidden surface algorithms"
Ph. Thesis, Univ. of Utah, March 1976

|2| E. CATMULL
"A hidden surface algorithm with anti-aliasing"
Computer graphics, vol.2, N°3, July 1978

|3| J. BLOMENTHAL
"Edge inference with applications to anti-aliasing"
Computer graphics, vol.17, N°3, July 1983

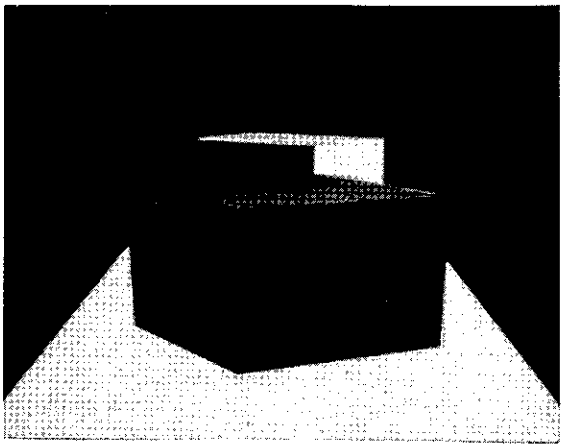


Figure 6 - Cube with edge filtering

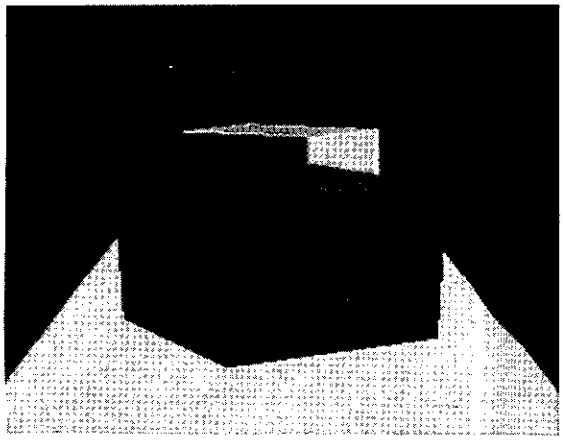


Figure 7 - Cube without edge filtering

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CENTRAL SYMMETRY MODELING

Josef Bigün, Gösta H. Granlund

Linköping University. Department of Electrical Engineering. S–581 83 Linköping, Sweden.

A definition of central symmetry for local neighborhoods of 2-D images is given. A complete ONset of centrally symmetric basis functions is proposed. The local neighborhoods are expanded in this basis. The behavior of coefficient spectrum obtained by this expansion is proposed to be the foundation of central symmetry parameters of the neighborhoods. Specifically examination of two such behaviors are proposed: Point concentration and line concentration of the energy spectrum. Moreover, the study of these types of behaviors of the spectrum are shown to be possible to do in the spatial domain.

INTRODUCTION

There is a long list of operators that detect the existence of linear symmetry in a local neighborhood. Most of them measure linear symmetry in the sense of lines and edges. But there is very little done to model central symmetry. Perhaps it is because images of objects in nature, are usually more irregular than circles. Nevertheless, we believe that this is one of the symmetries which human beings utilize in early vision. It seems that central symmetry should be an additional symmetry model. The fact that circularly symmetric shapes like rotating fans, diverging rays, circularly propagating water waves... e.t.c. are observed as phosphenes when low frequency magnetic fields are applied to the temples of a subject, [1], [2], supports this belief. Moreover many manufactured objects are locally concentrated and have closed rounded boundaries. Many natural objects in low resolution images may exhibit this property like cells seen under a microscope. Conceivable application areas are object counting, classification as well as image coding and enhancement for certain types of images, possessing local central symmetry property. But first we should have an intuitive feeling about what kind of patterns are called centrally symmetric in our terminology, since it is otherwise quite a vague concept.

DEFINITION: We will call local neighborhoods centrally symmetric if the locus of iso-gray values constitute parallel lines in local polar coordinates:

$$\varphi = k_1 r + k_2 \qquad r \geq 0$$

for some constants k_1 and k_2. if the locus of iso-gray values are not curves, that is when they are regions then the borders of these regions are considered as locus. We will assume that the boundary of the neighborhood is a circle, and the origin of coordinates is the center of this circle.

DEFINITION: $\mathcal{C}(\Omega)$ is the space of complex valued functions which are continuous on Ω except on a subset of Ω with zero measure. Ω is a circle with the radius R.

DEFINITION: (f,g) is the scalar product for $f,g \in \mathcal{C}(\Omega)$ with

$$(f,g) \triangleq \frac{1}{|\Omega|}\int_\Omega \frac{1}{r} f^*(\bar{r})g(\bar{r})d\Omega$$

with $r = |\bar{r}|$ and:

$$|\Omega| = \int_\Omega \frac{1}{r} d\Omega.$$

Consider the functions, see Figure 1),

$$\Psi_{mn}(\bar{r}) = e^{i(mwr+n\varphi)} \tag{1}$$

with $w = \frac{2\pi}{R}$ and $m,n \in Z$. $\mathcal{C}(\Omega)$ is a Hilbert space with the following scalar product

$$\frac{1}{2\pi R}\int_0^{2\pi}\int_0^R \bar{f}(r,\varphi)g(r,\varphi)drd\varphi.$$

Moreover $\{\Psi_{mn}\}_{m,n\in Z}$ is dense in $\mathcal{C}(\Omega)$, which follows from the Fourier series expansion theory on a rectangle, [3]. But this scalar product is the same scalar product defined earlier with, Ω, being a circle. By that we have established that $\mathcal{C}(\Omega)$ is a Hilbert space with the scalar product given in the definition. Now let us consider the neighborhood Ω, around an examined point in an image. Assume that the polar coordinates, $r = |\bar{r}|$ and $\varphi = arg(\bar{r})$, referred to in the following are relative to the examined point, and the positive x axis from the examined point.

Let the real function $f(\bar{r})$ express the gray values in Ω, with the center at the examined point, such that $\bar{r}$ is the local coordinate vector. Then one can expand f as

$$f(\bar{r}) = \sum_{m,n\in Z} c_{mn}\Psi_{mn}(\bar{r}) \tag{2}$$

with

$$\begin{aligned} c_{mn} &= (f,\Psi_{mn}) \\ &= \frac{1}{2\pi R}\int_\Omega \frac{1}{r} f(\bar{r})e^{i(mwr+n\varphi)}d\Omega \end{aligned} \tag{3}$$

because $\mathcal{C}(\Omega)$ is a Hilbert space and $\{\Psi_{mn}\}_{m,n\in Z}$ constitutes a complete orthonormal base:

$$(\Psi_{mn},\Psi_{m'n'}) = \delta_{mm'}\delta_{nn'} \tag{4}$$

with $\delta_{mm'}$ being the usual Kronecker delta.

POINT CONCENTRATION

DEFINITION: Let P be an operator from $C(\Omega)$ to the function set X, $X \subset C(\Omega)$. Then P is a projection from $C(\Omega)$ to X if

$$P^2 f = Pf$$

for all f in $C(\Omega)$.

Our goal is to find an algorithm based on operations done in the spatial domain which still gives some indication about whether the energy is concentated to a point in the frequency domain. The algorithm should posses the following properties:

1) Whenever the neighborhood, $f(\bar{r})$, is equivalent to one of the basis functions, Ψ_{mn}, except possibly for a scale factor B, the algorithm should detect this particular basis function save a sign change of it's index tuple, (n,m). That is $(f, \Psi_{m'n'}) = 0$ for all tuples (n',m') except for a tuple (n,m). In other cases it should give some sort of dominating tuple (n,m). It should be noted that Ψ_{mn} is complex valued. For real neighborhoods consisting of the real or imaginary part of a Ψ_{mn}, this condition will be enough to identify the neighborhood except possibly a phase factor. Given the tuple (n,m), Ψ_{mn} is unique. Call the operator of finding the tuple (n,m), and associating the function Ψ_{mn} to that, as P then:

$$P^2 f = Pf = \Psi_{mn}$$

for any $f \in C(\Omega)$. This is equivalent to saying that the sought algorithm is a projection to the countable set $\{\Psi_{mn}\}$, according to the projection definition above.

2) The projection value (or parameter) should be rotation and radial phase invariant for pure inputs of:

$$f(r,\varphi) = B\Psi_{mn}(r,\varphi)$$

with some scalar B. That is

$$Pf(r+r_0, \varphi+\varphi_0) = Pf(r,\varphi) = \Psi_{mn}$$

should be fullfilled.

3) Whenever the spectrum of the real valued local neighborhood differs from a pattern with a point concentrated spectrum, an uncertainty parameter should reflect that. By attaining low values, for example, this parameter could indicate the relevance of the projection parameter, and conversely to suppress it if the neighborhood differs from a central symmetric pattern.

The use of the uncertainty parameters is indicated in [4]. The uncertainty parameter and the projection parameter are combined in every point of the image to form a vector, in such a way that the magnitude of this vector becomes inverse proportional to uncertainty parameter, (the confidence in the projection parameter) and the argument of it becomes the projection parameter. This can be visualised by allowing the magnitude to modulate the intensity of a point in a color TV monitor and the argument of it, representing the projection parameter, to modulate the color of the same point. The result is a color image representing a decision in every neighborhood of the original image. The projection parameter and the confidence parameter values evaluated in every point in the image can be thought of as two separate images influencing each other. A point with a low confidence level looks dark in the resulting image, no matter what the color of the point is. A point with a high confidence level is emphasized by illumination, and it's color is revealed.

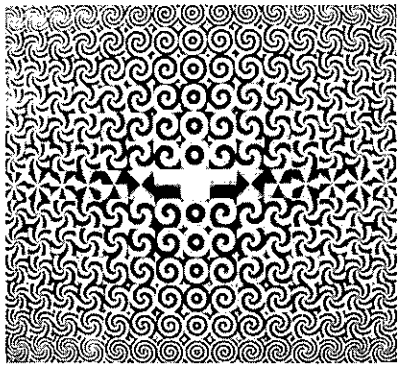


Figure 1) The image illustrates some of the basis functions Ψ_{mn}. The real parts of Ψ_{mn} are mapped linearly to the gray values of the monitor.

The algorithm we propose consists of the computations described by (5)-(9):

$$A \triangleq \|f\| \tag{5}$$

$$m_d \triangleq \frac{\|D_r f\|}{A\omega} \tag{6}$$

$$n_d \triangleq \frac{\|D_\varphi f\|}{A} \tag{7}$$

$$C^2_{\Omega m} \triangleq \frac{\|D_r^2 f\|^2}{A^2\omega^4} - m_d^4 \tag{8}$$

$$C^2_{\Omega n} \triangleq \frac{\|D_\varphi^2 f\|^2}{A^2} - n_d^4 \tag{9}$$

m_d and n_d are radial respectively angular frequency measures. $C_{\Omega m}$ and $C_{\Omega n}$ are the uncertainty measures associated with m_d respectively n_d. Denote the projection parameters by the tuple $(\hat{n}, \hat{m})$. That is $(\hat{n}, \hat{m})$ points out the location of an eventual point concentration in the spectrum. To produce $\hat{m}$ and $\hat{n}$ from m_d and n_d, we observe that $\hat{m}$ and $\hat{n}$ should be integers. Moreover they take positive as well as negative values. However since we assume real valued images, the requested point concentration will consist of two concentrations symmetrically located around the origin of the coordinates in the spectrum. This is due to the Hermitian property of the coefficient transformation. Hence we need only give the position of one of these concentrations. Thus we can assume that $\hat{m}$ is allways positive. We will simply assign to $\hat{m}$ and $\hat{n}$ the closest integers to m_d and n_d with proper sign:

$$\begin{aligned} \hat{m} &= round(m_d) \\ \hat{n} &= sign \times round(n_d) \qquad sign \in \{-1, 1\} \end{aligned} \tag{10}$$

sign is the sign of $\sum_{m,n\in Z} mn\frac{|c_{mn}|^2}{A^2}$, the calculation of which is given in next section. Let us see what (5)-(9) does for a neighborhood :

$$f = \sum_{m,n\in Z} c_{mn}\Psi_{mn} \tag{11}$$

We get through (5)

$$A^2 = \sum_{m,n\in Z} |c_{mn}|^2$$

This is the energy of the neighborhood in terms of the centrally symmetric function set $\{\Psi_{mn}\}$. (4) together with (6), (11) yields:

$$\begin{aligned} m_d &= \frac{\|D_r f\|}{A\omega} = \| \sum_{m,n\in Z} i\frac{mc_{mn}}{A}\Psi_{mn}\| \\ &= \Big(\sum_{m,n\in Z} \frac{c_{mn}^2}{A^2} m^2 \Big)^{\frac{1}{2}} \end{aligned} \tag{12}$$

Hence m_d is the weighted root mean square of all radial frequency measures, m. It should be observed that a particular radial frequency number, m, is weighted by the uniform sum of all angular frequency energies. The weights constitute energy distribution of the input function. The higher the energy share of Ψ_{mn} in the total energy, the more m_d will be close to m. (12) fulfills obviously the projection requirement after rounding m_d to the closest integer, $\hat{m}$. Similarly n_d will be the weighted mean square of all angular frequencies of different order:

$$n_d = \Big(\sum_{m,n\in Z} \frac{|c_{mn}|^2}{A^2} n^2 \Big)^{\frac{1}{2}} \tag{13}$$

The latter is insensitive to the sign changes in n. The consequence of this is a real neighborhood of

$$f = \Psi_{mn} + \Psi_{-m-n} + \Psi_{m-n} + \Psi_{-mn},$$

is projected to a $\Psi_{|m||n|}$ input. The decision is to the favor of one of the two equally strong candidates. When $f = \Psi_{mn} + \Psi_{-m-n}$ then $m_d = |m|$ and $n_d = |n|$ which in turn reflects the necessity of the variable *sign* referred to earlier (10). Uncertainty parameters $C_{\Omega m}$ and $C_{\Omega n}$ are proposed to be as in (8) and (9), and $C_{\Omega m}$ yield through (4), (8), (11), (12)

$$\begin{aligned} C_{\Omega m}^2 &= \frac{\|D_r^2 f\|^2}{A^2\omega^4} - m_d^4 = \sum_{m,n\in Z} \frac{|c_{mn}|^2}{A^2} m^4 \\ &\quad - 2\sum_{m,n\in Z} \frac{|c_{mn}|^2}{A^2} m^2 m_d^2 + m_d^4 \sum_{m,n\in Z} \frac{|c_{mn}|^2}{A^2} \\ &= \sum_{m,n\in Z} \frac{|c_{mn}|^2}{A^2} (m^2 - m_d^2)^2 \end{aligned} \tag{14}$$

which can be viewed as a weighted variance for the integers m^2. It attains it's minimum in the case when

$$\frac{|c_{mn}|^2}{A^2}(m^2 - m_d^2)^2 = 0$$

for all $n, m \in Z$. This occurs if and only if

$$\sum_n \frac{|c_{mn}|^2}{A^2} = 1$$

for some $m = m'$ since m_d^2 is constant. Thus if $C_{\Omega m}$ is zero then there exists one unique radial frequency in the neighborhood. And it is given by the estimation, m_d. When this is the case the energy is concentrated to a horizontal line through $m = m_d$. Since $C_{\Omega m}$ is a variance it also reveals some information about the shape of spectral density of the neighborhood. If $C_{\Omega m}$ is small then it is likely to think that the neighborhood is a degraded version of a wave with a well defined radial frequency, m_d. Conversely it is unlikely that the association of m_d to the neighborhood will be relevant, if $C_{\Omega m}$ is large. Interpretations of n_d and $C_{\Omega n}$ are similar to m_d and $C_{\Omega m}$'s. Given m_d, n_d, and the *sign* parameters the tuple $(\hat{n}, \hat{m})$ is computed according to (10). We adopt the uncertainty parameters $C_{\Omega n}$ and $C_{\Omega m}$ for $\hat{n}$ respectively $\hat{m}$. We propose $C_{\Omega\nu}$,

$$C_{\Omega\nu}^2 = C_{\Omega m}^2 + C_{\Omega n}^2 \tag{15}$$

to be the uncertainty parameter for the tuple $(\hat{n}, \hat{m})$. $C_{\Omega\nu} = 0$ if and only if $C_{\Omega m} = C_{\Omega n} = 0$. But $C_{\Omega m} = 0$ if and only if the total energy is concentrated on a horizontal line and $C_{\Omega n} = 0$ if and only if the total energy is concentrated on a vertical line. The only possibility for the neighborhood to fulfill these two requirements is being an input possessing total point concentration in it's spectrum, with location on the intersection of the lines $m = m_d$, $n = n_d$.

LINE CONCENTRATION

We will examine whether the energy spectrum has line concentration. Let the line we look for be

$$m = \tan(\theta)n \tag{16}$$

we assume that the line goes through the origin of the coordinates in the coefficient domain. Since the real functions coefficient transforms should be Hermitian, their energy spectra are even, forcing an eventuall line concentration to pass through the origin of the coordinates of the coefficient plane. A real neighborhood f can be expanded in the basis functions as before, yielding:

$$f = \sum_{m,n\in Z} c_{mn}\Psi_{mn}$$

with the Hermitian coefficients c_{mn}. The energy concentration of the neighborhoods in general degrades from a line through the origin of the coordinates. Let us measure this degradation by $C_{\Omega\theta}$, which is the average sum of the squares of the distances of the spectrum points to the line given by (16):

$$\begin{aligned} C_{\Omega\theta} &\triangleq \sum_{m,n\in Z} (m - \tan(\theta)n)^2 \cos^2(\theta) \frac{|c_{mn}|^2}{A^2} \\ &= \sin^2(\theta) \sum_{m,n\in Z} n^2 \frac{|c_{mn}|^2}{A^2} + \cos^2(\theta) \sum_{m,n\in Z} m^2 \frac{|c_{mn}|^2}{A^2} \\ &\quad - \sin(2\theta) \sum_{m,n\in Z} mn \frac{|c_{mn}|^2}{A^2} \end{aligned} \tag{17}$$

We want to find a θ which minimizes $C_{\Omega\theta}$. This is the least square estimation of θ and it is straight forward to find θ:

$$\frac{dC_{\Omega\theta}}{d\theta} = (n_d^2 - m_d^2)\sin(2\theta) - 2p\cos(2\theta) \qquad (18)$$

where

$$p = \sum_{m,n\in Z} mn\frac{|c_{mn}|^2}{A^2}$$

If $n_d^2 - m_d^2 \neq 0$ or $p \neq 0$ then choose the minimizing θ as:

$$\theta_d = \frac{1}{2}\tan^{-1}(n_d^2 - m_d^2, 2p). \qquad (19)$$

The degradation or uncertainty measure is given by substituting (19) in (18) and using the trigonometric half angle formulas:

$$C_{\Omega\theta} = \frac{1}{2}\left(n_d^2 + m_d^2 - \sqrt{(n_d^2 - m_d^2)^2 + 4p^2}\right). \qquad (20)$$

The angle given by (19) gives the axis around which the moment of inertia is minimum and the moment of inertia is given by (20) if $\frac{|c_{mn}|^2}{A^2}$ is seen as a point mass, [5]. The omitted case when both $p = 0$ and $n_d^2 - m_d^2 = 0$ corresponds to local neighborhoods with no specific orientation. Because $\frac{dC_{\Omega\theta}}{d\theta}$ vanishes according to (16), any θ would work as minimizing argument to (17). This case implies that

$$\sum_{m,n\in Z} m^2\frac{|c_{mn}|^2}{A^2} = \sum_{m,n\in Z} n^2\frac{|c_{mn}|^2}{A^2}$$

$$\sum_{m,n\in Z} mn\frac{|c_{mn}|^2}{A^2} = 0$$

The class of functions having this property in their spectra is the class of functions with coinciding principal axes in the coefficient domain. Neighborhoods of Ψ_{00}, $\Psi_{mn} + \Psi_{-m-n} + \Psi_{m-n} + \Psi_{-mn}$ are examples of such functions. we observe that $m_d^2 = 0$ and $n_d^2 = 0$ implies that the neghborhood is a constant function and consequently have no orientation. Thus to keep the consistency of the meaning of $C_{\Omega\theta}$ in the case when $n_d^2 - m_d^2 = p = 0$, we should define $C_{\Omega\theta} = \infty$ and leave θ undefined. p which is needed to calculate θ_d and $C_{\Omega\theta}$ according to (19) and (20), can be easily found in the spatial domain to be:

$$p = \sum_{m,n\in Z} mn\frac{|c_{mn}|^2}{A^2} = \frac{1}{A^2\omega}\left(\frac{\partial f}{\partial r}, \frac{\partial f}{\partial \varphi}\right)$$

Implementation of the scalar products given above for every neighborhood of a digitized image is straight forward after the usage of the chain rules:

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x}\cos(\varphi) + \frac{\partial f}{\partial y}\sin(\varphi)$$

$$\frac{\partial f}{\partial \varphi} = \frac{\partial f}{\partial y}r\cos(\varphi) - \frac{\partial f}{\partial x}r\sin(\varphi)$$

By that we can transfer the scalar products to be valid for functions defined in cartesian coordinates. At this point we can use either the band limited signal theory or some quadrature rule to evaluate the resulting integrals, given that we know $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ at a rectangular net of points. It can be shown that the scalar product evaluations at every point is obtained by convolutions with FIR-filters.

CONCLUSION

Both the point concentration parameters m_d, and n_d and the line concentration parameter θ_d are best fits of a point or a line respectively through the origin of coordinates of the coefficient domain. The best fit is in the sense that the two variance measures given, which are adopted as uncertainty measures, are minimized. It is interesting to note that the approach lends itself to linear symmetry parameter extraction as well, with a minor change. By linear symmetry we mean the neighborhoods with iso-gray values being straight lines in cartesian coordinates. Parallel lines belong to such neighborhoods. Hence it is possible to find the dominating frequency and the dominating orientation of a neighborhood, [6], with the least error variance in the fourier domain in a similar manner. The only difference is the scalar product and the shape of the neighborhood. The scalar product of the linear symmetry case becomes the usual $\mathcal{L}^2(\Omega)$ scalar product with Ω being a rectangle. The complete ON-basis set is of course $\{e^{i(m\omega_x x + n\omega_y y)}\}_{m,n\in Z}$.

REFERENCES

[1] Max Knoll, Johann Kugler, Joseph Eichmeier and Oskar Höfer: Note on the Spectroscopy of Subjective Light Patterns. Journal of Analytical psychology, No 7, 1962,

[2] Per Lövsund: Biological Effects of Alternating Magnetic Fields with Special Reference to the Visual System. Dissertation No 47, 1980, Linköping Studies in Science and Technology, Linköping University, Sweden.

[3] W. Rudin: Real and Complex Analysis. McGraw Hill, 1969.

[4] G.H. Granlund: Hierarchical Image Processing. Proceedings of SPIE Technical conference, Geneva, April 18-27, 1983.

[5] I.L. Meriam: Statics. John Wiley & Sons, New York, 1980.

[6] H. Knutsson: Filtering and reconstruction in image processing. Dissertation No. 88, 1982, Linköpings Studies in Science and Technology, Linköping University, Sweden.

[7] Robert A. Hummel: Feature Detection Using Basis Functions. Computer Graphics and Image Processing 9, 1979.

[8] P.E. Danielsson: Rotation invariant linear operators with directional response. Proceedings of 5'th international conference on pattern recognition, December 1980.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Edge Detection by Combining Directional Derivatives.

J. Besuijen

Control Systems and Computer Engineering Group
Department of Electrical Engineering
Twente University of Technology, P.O.Box 217
7500 AE Enschede, The Netherlands.

First and second order directional derivatives are combined to produce accurate and thin edges, even from noisy or blurred originals. Results show a superior overall performance to first order gradient edge detectors like the Sobel, Prewitt and Roberts edge detector, and similar or better results than the Marr-Hildreth operator.

1. Introduction

The detection of all and only the relevant edges in an image still is a complex task. Using a simple kind of gradient method directly on an image degraded by noise can cause discontinuities in the detected edges and the detection of many false edge points. Suppressing the noise by low-pass filtering will blur the edges and make it more difficult to detect their position and structure. The use of a first derivative kind of edge detector in a low-pass filtered image will lead to edges of several pixels wide: an unwanted feature for most applications of edge detectors. Recently developed edge detectors make use of zerocrossings of second derivatives of low-pass filtered images. This has the advantage that the edge width will be limited; false edge points of course can still be detected.

2. Edge detection using zerocrossings of second derivatives.

An elegant solution to the edge detection problem has been put forward by Marr and Hildreth [1-2]. They proposed to filter the image by circle-symmetric Gaussian filters of different widths and then compute the zerocrossings of the Laplacian:

$$\nabla = \frac{\delta^2}{\delta y^2} + \frac{\delta^2}{\delta x^2}$$

and combine the results to find, what they call: "the primal sketch". They used the Gaussian filter because it has no sidelobes in the frequency domain, and the Laplacian because it is the simplest non-directional linear differential operator. This method will generally result in a thin edge of reasonably accurate position, but it has some deficiencies.
Berzins [3] described and quantified the displacement of contours which can be introduced by the Marr-Hildreth edge detector, especially at curves or corners in the contour. Another problem is that the M-H-operator always yields closed contours. This can be very useful in certain types of images, but very disturbing in others. If contours adjoin or intersect, conflicts can arise, which can results in contours wandering off through the middle of areas of constant grey level. This is illustrated in figure 1(a-c).
Given an original image with three areas of constant grey level as in figure 1a, depending on the ratio of the different grey level steps, the M-H edge detector will give edges as depicted in figure 1b or 1c. These faults cause severe problems for the analysis of a scene.

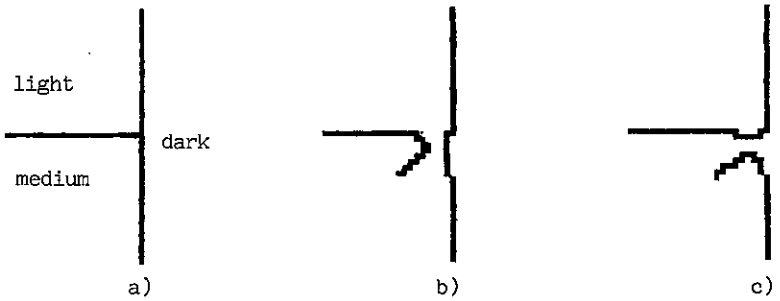


Figure 1.

Combining results from several widths of Gaussians will not help, and in many cases even intelligence can not cover up these gaps.

The displacement of contours is entirely due to errors introduced by the Laplacian, and not due to the Gaussian filtering [3]. This holds for the intersection problems as well: The zerocrossings of second order gradients perpendicular to the contour are always located in the right position. It seems to be preferable to compute these zerocrossings. The problem is of course that the direction of a contour-part is usually not known in advance. Haralick [4] developed a method based on his facet model which computes gradients perpendicular to the local contour direction. This method however does not seem to be computationally fast and the results still show edges of several pixels wide.
Given the results and conlusions above, it seems possible to develop an edge detection method based on a simple combination of second directional derivative zerocrossings, that would give slim and connected edges even in low-pass filtered noisy images.

3. A fast and robust edge-detection algorithm

The characterisation of intensity changes in a digital image has to consist firstly of a filtering step and secondly derivatives of several types have to be taken and combined, as shown recently by Torre and Poggio [5]. They further show that a Gaussian filtering is near optimal, and considerably faster than optimal solutions, because the 2D-filter can be decomposed in two 1D-filters. The use of directional derivatives seems to imply the use of filters that are longer in the direction perpendicular to that of the derivative. These directional filters however tend to stretch the edges at corners and intersections of contours. For the purpose of scene analysis an edge-detector has to produce an image that is as intelligible as possible. Misleading structural information is far more disturbing than random false detected edge pixels.
The Combination Of Directional Derivatives (CODD) algorithm here presented does not produce spurious edges and performs well under noisy conditions. With e(i,j) the edge values in pixel (i,j) and f(i,j) the grey level function in the Gaussian filtered original image:

$$e(i,j) = \sqrt{\left(\frac{\delta f(i,j)}{\delta x}\right)^2 + \left(\frac{\delta f(i,j)}{\delta y}\right)^2},$$

$$\text{if } \frac{\delta f(i,j)^2}{\delta x^2} >\!< 0 \text{ or } \frac{\delta f(i,j)^2}{\delta y^2} >\!< 0$$

$$= 0, \text{ otherwise}$$

with >< denoting the crossing of the level to the right of the operator. So the second derivative is used to find the location of the edge points and the first derivative is used to compute a slope value. This slope value of course is well known from other edge detectors and is very appropriate for square pixel grids. A problem arises when the image is heavily low-pass filtered before edge detection, because the slope values decrease with the width of the filter used. A scaling constant should be applied to correct the attenuation introduced by the filtering. The scaling is also necessary for the second derivative, because thresholds are needed in the detection of the zerocrossings. The scaling factor is also dependant of the differential operator used, and the curvature of the edge. For the [-1,0,1] operator and a straight edge the scale factor s for a Gaussian filter with standard deviation σ will be:

$$s = \mathrm{erf}(1.5/\sigma) - \mathrm{erf}(-.5/\sigma)$$

Another advantage of this method is that all processing can be done in parallel although the detection of zerocrossing is a bit too tricky for most parallel image processors. For reasonable quality images a 3*3 convolution will suffice for the Gaussian filtering. The differentiations are done with simple [-1,0,1] and [-1,2,-1] operators for the first and second directional derivative resp. The detection of zerocrossings requires a rather complex logical expression and will therefore take several image processing steps on current parallel image processors.

4. Results.

A problem in the comparison of edge detection algorithms is the lack of good quality criteria and standard test images, that cover all the aspects of edge detection. The best known quality criterion is the Figure Of Merit, developed by Pratt and Abdou [6]. This FOM only works for images with a single edge, and is very dependent on what the user defines as an edge pixel. Due to these inconsistencies it seems best to let the results illustrate the qualities of the CODD-operator, and not add definition dependent percentages.

Figure 2a.

Figure 2b.

Figure 2 shows the edge-map for the CODD-operator for some typical test images from [6]. The edge maps are shown for a vertical and a diagonal edge of width w = 1, step h = 24, and a signal to noise ratio SNR = 1.
For all the results in figures 2-4 an 11*11 Gaussian filtering mask was used with standard deviation σ = 2. The threshold values used, were defined by user interaction.
Synthetic test images are usually created with the same philosophy in mind as used for the development of the edge detector. The checkerboard test images used by Haralick [4] for his second directional derivative edge detector is well suited for the facet model underlying this edge detector. (A facet is a K*K pixel block, which grey level function can be modelled by at most a second order polynome). A coincidence is that the Marr-Hildreth edge detection method tested in the same article is also very well suited for the checkerboard test image. At this symmetric type of intersections the zerocrossings of the Laplacian do not shift.

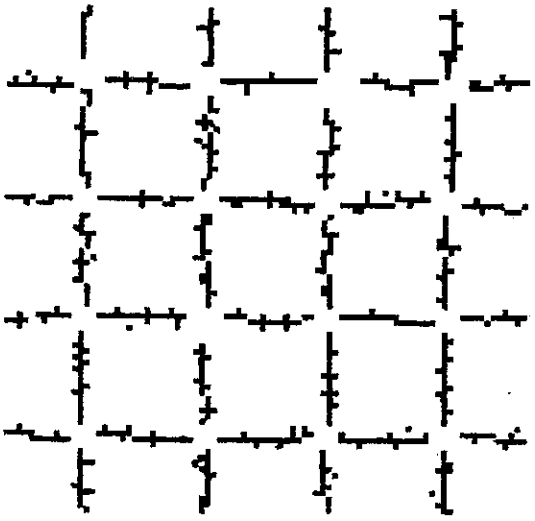


Figure 3a.

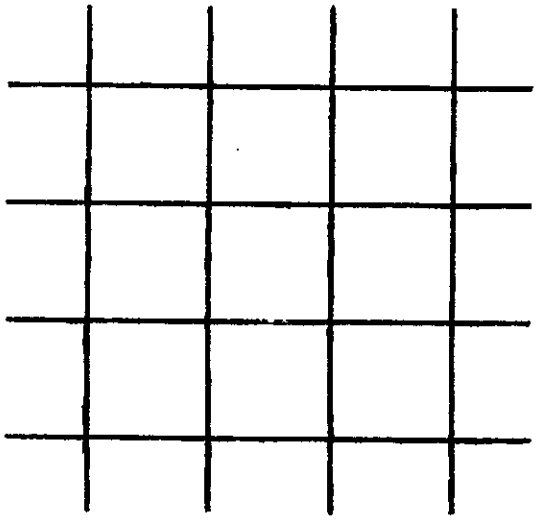


Figure 3b.

Reactions to Haralick's article [7-8] further show that it is nearly impossible to objectively compare edge detection algorithms, especially when they work best with specially developed hardware.
Figure 3 shows the response of the CODD-operator to the checkerboard test image, with a) a signal to noise ratio of 3 dB, and b) no noise.

The detector performs relatively poor on these crossings of edges. Due to the heavy filtering the output at the crossings is weak. In noisy conditions the crossings cannot be detected. The CODD-operator however does not introduce spurious edges as can be seen in figures 3b and 4b. Figure 4 shows the response of the CODD-operator to the test-image of figure 1a. The grey levels used are 159, 127 and 95. The original image of figure 4a was degraded with noise of SNR = 1 for the smallest grey level steps.

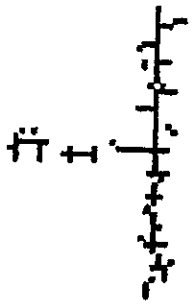


Figure 4a.

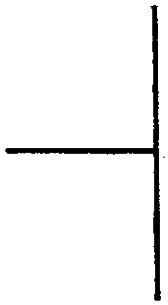


Figure 4b.

5. Conclusions.

The CODD-operator here presented seems to be fast, accurate and robust. Comparison with the Marr-Hildreth-operator shows an opposite behaviour on symmetric and asymmetric edge crossings. It would be interesting to know how human perception copes with these different kinds of crossings.

[1] Marr, D. and Hildreth, E., Theory of edge detection, Proc. R. Soc. Lond. B 207 (1980) 187-217.

[2] Hildreth, E. C., The Detection of Intensity Changes by Computer and Biological Vision Systems, CVGIP 22 (1983) 1-27.
[3] Berzins, V., Accuracy of Laplacian Edge Detectors, CVGIP 27 (1984) 195-210.
[4] Haralick, R.M., Digital Step Edges from Zero Crossing of Second Directional Derivatives, PAMI 6 (1984) 58-68.
[5] Torre, V. and Poggio, T.A., On Edge Detection, PAMI 8 (1986) 147-163.
[6] Abdou, I.E. and Pratt, W.K., Quantitative Design and Evaluation of Enhancement / Thresholding Edge Detectors, Proc. IEEE 67 (1979) 753-763.
[7] Grimson, W.E.L. and Hildreth, E.C., Comments on [4], PAMI 7 (1985) 121-127.
[8] Haralick, R.M., Author's Reply on [7], PAMI 7 (1985) 127-129.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

M.14

AUTOMATIC COUNTING OF ASBESTOS FIBRES

G. van Antwerpen, P.W. Verbeek and F.C.A. Groen*

Pattern Recognition Group, Dept. of Applied Physics
Delft University of Technology, Lorentzweg 1
NL-2628 CJ DELFT, The Netherlands

This project was sponsored by the European Community and the Dutch Government under Grant ENV-694-N (B)

Keywords:
Cellular logic, shape analysis, industrial inspection, size measurement, image analysis, asbestos, environment, labour hazard.

A method is presented for automatic analysis of Transmission Electron Microscope images of samples of fibres and other objects. The analysis is to be applied to samples from environments with asbestos hazard, where an objective measure of fibre frequency is wanted. An image is automatically corrected for shading and thresholded, the binary image is distance transformed and skeletonized.
Skeletons are chain coded while width at each skeleton location is stored. Crossings are removed and resulting segments recombined. Total length and typical width of (overlapping, not a priori known) objects are estimated. From these features objects are designated as fibres. Fibres are then counted and statistics is performed on the image. Locations on the fibre are calculated and sent to the electron microscope for X-ray chemical identification.

1. INTRODUCTION

One of the problems in the analysis of asbestos images in electron microscopy is the discrepancy between fibre counting results of different human counters as well as between results of the same counter at different times. Moreover, fibre counting is a very time consuming and extremely tedious job. Early 1983 a project was started to develop a method for fully automatic analysis of Transmission Electon Microscopic (TEM) images of asbestos or other mineral fibres. This paper describes the computer program for automatic fibre recognition which is part of an integrated effort towards automation of asbestos counting with transmission electron microscopy, comprising the following items:

- selection and development of a TEM preparation technique and TEM-system automation to produce TEM pictures of constant quality (S/N ratio, contrast, shading etc.) for the full size range of asbestos fibres.
- development of a microcomputer application program for fibre recognition, following the counting rules for asbestos fibres.
- development of hardware and software for interfacing the microcomputer with the TEM and an EDXA+ system for chemical analyses.

Existing solutions for analysis of (microscopic) asbestos images, like those of Magiscan [1] have been studied. Some ideas, such as the method for separation of overlapping objects have been adopted. The Magiscan method does not perform a width measurement or analysis (as it has been developed for light microscopy where fibres are typically a few pixels wide). Indeed it showed very poor performance for wide fibres.

* This work was part of a research project of the Department of Indoor Environment of the TNO division for Society, Delft in cooperation with the Technical University Delft and the Centre for Analytical Electron Microscopy, Leiden.

+ EDXA = energy dispersive X-ray analyses.

2. PROBLEM DEFINITION

The following problem had to be solved. As an input the computer gets an image with - in general - uneven background grey value, in which fibres and other objects are present. These fibres and objects may overlap (conglomerates). First of all a solution must be found for the reduction of the input image to an image in which objects and background are separated. Next the objects must be analysed, which should result in a description that indicates whether or not an object is an (asbestos) fibre. Rules for counting and measuring fibres and conglomerates have been internationally accepted and the solution pursued here should be in accordance with them. The method should also allow indicating a number of positions on a detected fibre, where energy dispersive X-ray analysis (EDXA) will be performed to determine the type of asbestos.

3. COUNTING RULES

The counting rules can be roughly described as follows. Count all objects with length/width ratio over three. For conglomerates separate constituents must be measured as far as possible. For fibres partly covered by a non-fibre particles the following rules hold: fibres that are visible at both sides of a particle are counted as one; fibres of which an end is covered by a particle are considered to run half way under the particle, as long as this is less than the visible fibre length. More detailed information is given in the remainder of the paper.

4. IMAGES

The images to be analysed presently consist of 256x256 pixels, each with grey value between 0 and 255. Image size in pixels determines the accuracy of size measurement. In this project it was limited by the available systems. When hardware for larger images becomes available image size can be easily adapted. It is an advantage of larger images (with higher resolution) that both small and very large objects can be processed in one pass. The solution proposed here asks for separate analyses for large and small objects, at different magnifications. Of course the full benefit of image analyses for automation of electron microscopy is not yet reached.

5. IMAGE PROCESSING, DEFINITIONS, OPERATIONS

Input images have 8 bit grey values. Objects are separated from background by thresholding. The threshold is calculated from the image grey value histogram.
If necessary a shading correction is performed first. The binary image (object pixels 1, background pixels 0) are processed by cellular logic operations (erosion, dilation, skeletonization, Hilditch [2]). Skeletons - thin lines roughly in the middle of objects and reflecting their topology - are efficiently stored and analysed in Freeman chain code representation.

6. METHODS

For recognition of fibres in a TEM image the following operations must be performed: image segmentation, binary image analysis and object analysis.

6.1. Image segmentation

First the image must be reduced from a grey value image to a useful binary image. This is accomplished in the following steps.

6.1.1. Background correction

For background correction so-called minimum and maximum filters have been used. These filters assign to any pixel the minimum or maximum value in its neighbourhood as its new value. Applying a minimum filter followed by a maximum filter of the same size one finds the lower hull of the grey value plot. Objects being lighter than their background disappear. The lower hull is somewhat smoothed by a linear low pass filter and used as background estimation to be subtracted before thresholding. A one-dimensional illustration has been sketched in figure 1.

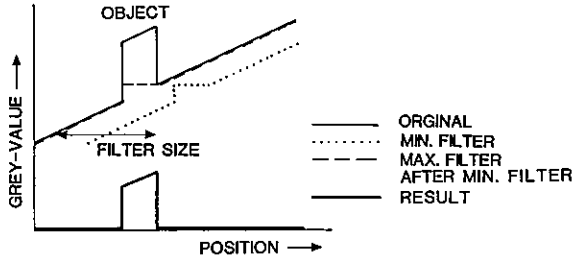


Fig. 1. One-dimensional example of the correction for a changing background value.

6.1.2. Thresholding

Now that the background correction has been applied, segmentation can be achieved by simple thresholding. The threshold value is found from the image histogram.
The histogram of the image after shading correction is found to be roughly a normal distribution with its maximum at the background

value P. Its tail at higher grey values contains the objects wanted. As thin poorly visible fibres should be detected as well, a threshold definition is chosen that results in threshold values near the background peak in the histogram. Smeulders rule: "measure the full width of the background peak at half height and add this to the peak position" (cf. fig.2) is very useful. Lower thresholds yield many false objects, higher ones miss essential objects.

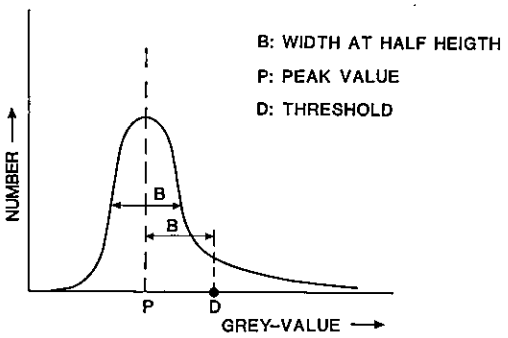


Fig. 2. Threshold determination.

6.1.3. Small objects
Thresholding even followed by the elimination of single pixel noise results in an image with still numerous very small isolated objects (mostly noise generated artifacts), certainly no fibres. In order to free the remaining analysis of this burden a method has been devised to remove these objects. Objects that become a single pixel by two skeletonizing erosion passes (without endpixel condition) are removed. This causes objects shorter than six pixels to disappear.

6.2 Analysis of the binary image

Between the creation of the binary image and the start of the actual analysis the objects must be reduced to lines with known width. This representation has been chosen as it allows conglomerates to be considered as graphs (sets of touching and crossing lines) the structure of which is easy to describe. For each position on such a line the width of the original object must be stored in order to allow later length-width analysis. This result can be achieved through the following steps.

6.2.1. Distance transform and skeleton
From a binary image one may get a "distance"-image through the "distance transform", as described by Borgefors [4], where each pixel of an object gets a value equal (or proportional) to the shortest distance to the object boundary. One may define a skeleton that contains the maxima of this distance image [5]

Such a skeleton can be written as a graph (nodes connected by edges) where at each location on the edges the distance to the border of the original object is known. It has been found essential to calculate this distance in a more sophisticated way as the distance of the skeleton to the object boundary gives insufficient width information. Instead the distance transform is calculated on a blown up version of the image where each pixel has been replaced by 2x2 pixels of the same value. For a skeleton pixel thus replaced the 2x2 pixels get 2x2 distance values; the maximum of these is adopted as local width at the skeleton pixel. In an object of width 3 the skeleton is as far from the boundary as in an object of width 4.

6.2.2. Shaving the skeleton
Skeletons usually have ravels emerging from rough object boundaries. Ravels hamper analysis and should be removed if possible. A procedure has been developed that removes skeleton branches if they protrude no more than two pixels outside the object. This proves sufficient to remove the worst ravels without damaging interesting protrusions from larger objects, protrusions that might be fibres.

6.2.3. Object labeling
The remaining skeletons are labeled. Each object skeleton gets a number for later sequential analysis.

6.3. Object analysis

Starting from the image with numbered skeletons and the width value image an analysis program is run. Analyzing an object in each cycle this program goes through the following steps.

6.3.1. Removing crossings
Fibres can cross. Where objects cross at least one skeleton point will have more than the usual two neighbours: a "branch point". Such branch points are removed, the skeleton breaks up and the object dissolves in separate segments of known skeleton and width.

6.3.2. Chain code generation
In this step the skeleton segments, so far in pixel representation in the image, are stored as chain code in a file of the program. This amounts to a huge data reduction and corresponding speeding up of the subsequent analysis.

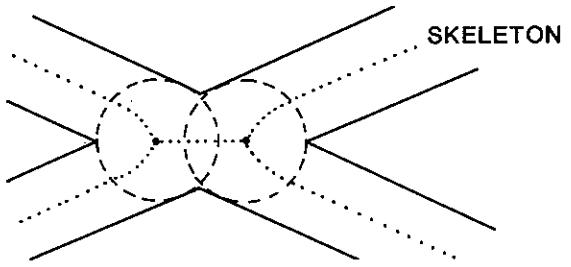


Fig. 3. Skeleton of crossing objects.

6.3.3. Removing data near crossings
Where fibres cross the skeleton usually has two branch points (cf. figure 3). The segment ends that remain after removal of branch points bend towards each other in a confusing way instead of being each other's extrapolation. The bending takes place in a cicular area around the branch point with a diameter roughly equal to that of the local object diameter. Segment ends within these areas are easily removed. This facilitates linking of related segments at a later stage of analysis.

6.3.4. Breaking skeletons at kinks
As fibres may lie end to end causing a sharp bend in the common skeleton, skeletons must be broken at such kink positions. In order to achieve this, at each location of the skeleton forward and backward direction are estimated. If the difference exceeds a given threshold (e.g. 45 degrees), then the skeleton is broken there.

6.3.5. Endpoint description
Starting from the chain code represented skeletons a table is generated that contains each skeleton end point. The table stores position, direction of departure, skeleton length and segment number. Skeleton length is calculated according to Vossepoel and Smeulders [6]

$$LS = .948 * NG + 1.40 * ND$$

where LS = skeleton length, NG = number of skeleton steps along the pixel grid and ND = number of diagonal skeleton steps.
The direction is estimated by fitting a straight line along a piece of line as long as the local object width, but not smaller than eight pixels.

6.3.6. End point linking
In order to appropriately count crossing fibres opposing segments must be counted as one. Hence each pair of ends is tested for linkability. The link is made if the following criteria are met:

6.3.6.1 Angle between ends less than a given threshold (22.5 degrees).

6.3.6.2 Angle between ends and their straight connection less than a given threshold (22.5 degrees for close ends, the threshold decreases with increasing distance).

6.3.6.3 Straight connection shorter than a given factor (3.) times the shortest of the segments to be connected.

6.3.6.4 Straight connection is within the original object.

The magic numbers in these criteria have been chosen so as to mimic the human counting behaviour (under the counting rules) as closely as possible.

6.3.7. The analysis of objects found
Now that the objects have been reduced to skeleton segments with connected or non-connected ends, the analysis of separate fibres can be performed. The length of a single fibre is determined as the sum of known segment lengths plus the sum of link distances. To this result, equal to the path length between the graph end points, is added the average width at both end points. It is assumed that the skeleton ends at a point equally far from the fibre sides and the end. Objects with one or more branches get a length equal to the maximum path length through the skeleton (cf. figure 4). The fibre width is determined from the histogram of the widths along the skeleton. Presently the mean of the lower 70% is calculated as object width. Thus the frequent upper outlyers are excluded, but the method wants some refining.

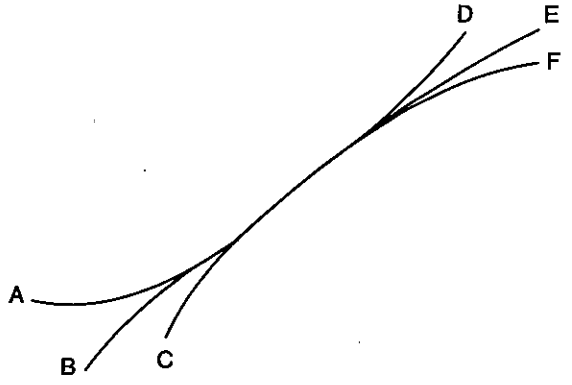


Fig. 4. Length of a split fibre is the distance, along the skeleton, from A to E.

6.3.8. Is the object a fibre?
In order to determine if the object found is a fibre the counting rule is applied that the length/width ratio should exceed the value 3. In very small objects the ratio measured is less dependable and thus for objects shorter than 0.5 microns the ratio should exceed 7.
When an object has been acknowledged as fibre a set of equidistant positions on the skeleton is generated and transmitted to the Transmission Electron Microscope, which initiates an EDXA analysis of chemical composition at the corresponding sample locations. The positions are selected away from crossings in order to

prevent analysis where objects overlap. The X-ray results decide if the fibre consists of asbestos and if so, the possible kind of asbestos.

6.3.9. Problems

Problems are caused by small objects. When they are isotropic with a rough boundary they usually have a skeleton not at all representative for the shape. As the width is not measured perpendicularly to the fibre axis but simply as twice the shortest distance from skeleton to boundary no dependable width measurement is made. Nevertheless the distances measured could well yield shape information. This is subject of further study.

7. Software and hardware

The software package has been written in Fortran-77, a number of time-critical routines (filters and distance transfom) have been written in C. The binary operations are performed by a Cellular Logic Processor, special hardware for VME-bus systems designed and developed in a cooperation project of Delft University of Technology and a department of the Dutch Organization for Applied Research T.N.O. (not the asbestos project). The system has been implemented on a UNIX-M68000 TNO-IBBC Geminix computer, to be coupled to the Transmission Electron Microscope for full automation.

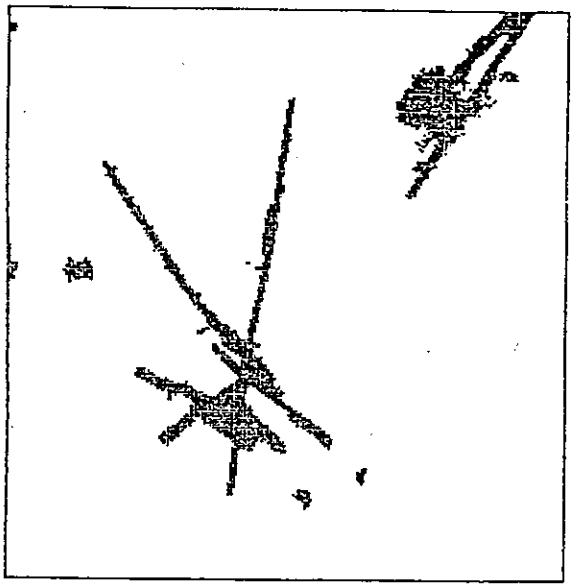


Fig. 5. Result of the processing of an asbestos image. Thresholded original.

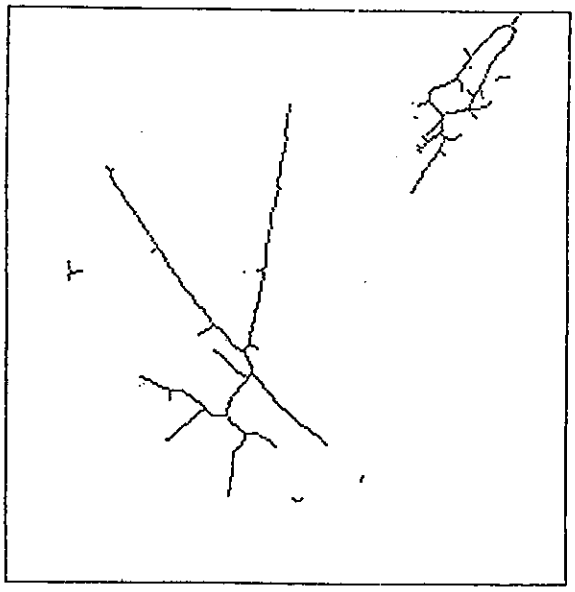


Fig. 6. Result of the processing of an asbestos image. Skeleton of fig. 5.

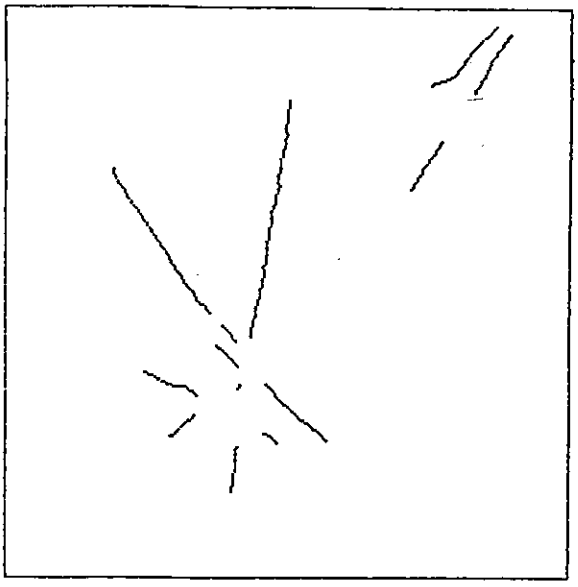


Fig. 7. Result of the processing of an asbestos image. Removal of small objects and small strokes. Removal of a small region around junctions.

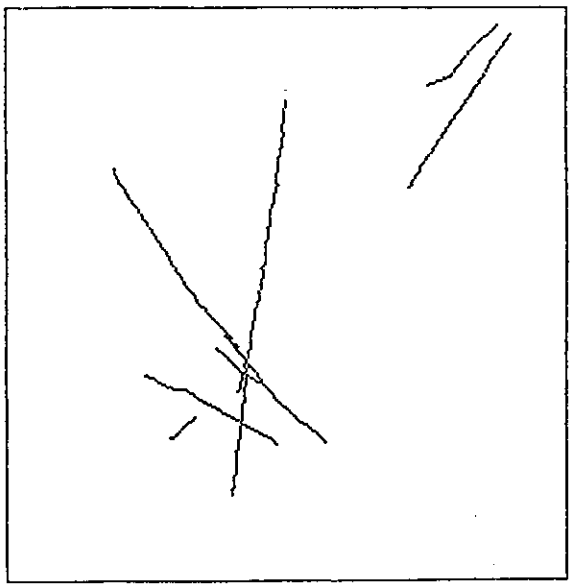


Fig. 8. Result of the processing of an asbestos image. Skeleton with the connections proposed by the program.

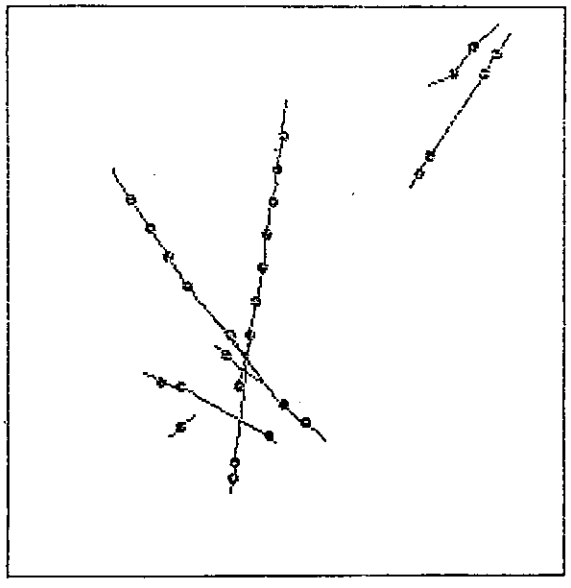


Fig. 9. Result of the processing of an asbestos image. Points for analysis added.

8. RESULTS AND CONCLUSIONS

- Comparitive counts have shown that the difference between automatic TEM counting and manual TEM counting is of the same order of magnitude as the difference between manual counters. Analysis of the differences shows that the main problem of the detection of very fine asbestos fibres between large fibres has been solved satisfactorily.
- A typical series of processing results is given in figures 5 to 9.
- This software package has been implemented for on-line fibre analysis (EDXA) of asbestos fibres by Transmission Electron Microscope.

ACKNOWLEDGEMENTS

The close and inspiring cooperation with Ir. W. ter Kuile, L. Drenth, P. Zandveld and the stimulating discussions with M. van Noord, Ir. P. Meyer, Prof.dr. I.T. Young, Ir. L. Dorst, Dr.ir. A.W.M. Smeulders, Drs. H. Meppelder, Dr. L. Strackee, Ir. W. Sprong are gratefully acknowledged.

REFERENCES

[1] R.N. Dixon and C.J. Taylor, Automated asbestos fibre counting, Inst. Phys. Conf. Ser. 44 Chapter 4.

[2] C.J. Hilditch, Linear skeletons from square cupboards, Machine Intelligence IV, B. Meltzer and D. Michie, Eds. Edinburgh. Edinburgh Univ. Press, 1969, pp. 403-420.

[3] A.W.M. Smeulders, Pattern analysis of cervical specimens. PhD Thesis, Kanters BV, Alblasserdam, 1985.

[4] G. Borgefors, Distance transformations in Arbitrary Dimensions. Computer Vision, Graphics and Image Processing, 27, 1984, pp. 321-345.

[5] L. Dorst and G. van Antwerpen, Pseudo Euclidean Skeletons, Proceedings of the Eighth Int. Conf. on Pattern Recognition, Paris, 1986.

[6] A.M. Vossepoel and A.W.M. Smeulders, Vector code probability and metrication errors in the representation of straight lines of finite length. Comp. Graphics and Image Processing, 20, pp. 347-364, 1982.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MEASURING ROTATIONS AND TRANSLATIONS OF DIGITIZED IMAGES

S. Alliney (*) and C. Morandi (**)

(*) Dipartimento di Matematica, Università di Bologna, via Vallescura, 2 - I-40136 Bologna
(**) DEIS, Università di Bologna, viale Risorgimento, 2 - I-40136 Bologna
and DEA, Università di Ancona, via Brecce Bianche, I-60100 Ancona.

Abstract: A novel algorithm for the registration of rotated and translated images is presented. Numerical procedures and experimental results are reported.

1. INTRODUCTION

Image registration is a relevant problem in many fields. As an example, in satellite image processing [1] it is necessary to align earth images obtained in different radiation bands, or images of the same region taken at different times. Similarly, in guidance systems, an image of the ground must be compared with a stored reference to check the flight course [2].

The same algorithms used for image registration can be used for the stabilization of image sequences. In fact, it frequently occurs that the observation on the TV monitor of a sequence of images is made difficult by unpredictable relative movements of the camera and the scene. Image registration algorithms yield the displacement of each new image with respect to a reference one, and this displacement may be subtracted from the coordinates of each pixel in the new image to obtain a stabilized version for display. The problem was first encountered in mobile video tracking systems, where vibrations in the camera mount cause severe problems. More recently image stabilization was recommended in connection with a video-assisted system for aircraft guidance [3].

Also in the biomedical field stabilization would be frequently desired since often involuntary movements of the subject cannot be avoided. The problem was considered in [4] in connection with a television ophthalmoscope.

There are several image registration algorithms known [1, 5-7], mostly dealing with translational movements only. Among them, the outstanding performance of the phase correlation algorithm [8-9] should be pointed out.

This work deals with an original method for the registration of rotated and translated images, based on the use of 2D Fourier transforms (FT) of the images. As known, the magnitude of the FT of an image is shift-infariant, whereas it rotates by the same angle as the image itself. Therefore it is possible to recover the rotation angle by comparing the magnitudes of the FT_S of the two images. Once the rotation angle is known, displacement components are determined either using the phase information, as in [8-9], or by simply computing a cross-correlation function.

2. ABOUT THE ESTIMATION OF RIGID ROTATIONAL MOVEMENTS

Let $g_1(x, y)$ denote a two-dimensional "density function" rapidly decreasing at infinity. If we establish a polar reference system over the plane (x, y), we have, for any pair $(x, y) \neq (0, 0)$,

$$x = \rho \cos \vartheta$$
$$y = \rho \sin \vartheta$$

Thus the density function can be rewritten as $g(\rho \cos \vartheta, \rho \sin \vartheta) = f(\rho, \vartheta)$, and it is evident that the function f is periodic (with period 2π) with respect to its second argument. It follows that there exists the Fourier series representation

$$f_1(\rho, \vartheta) = \sum_n a_{1n}(\rho)\, e^{in\vartheta} \tag{2}$$

$$a_{1n}(\rho) = (1/2\pi) \int_0^{2\pi} f_1(\rho, \vartheta)\, e^{-in\vartheta}\, d\vartheta \tag{3}$$

where the coefficients $a_{1n}(\rho)$ are themselves rapidly decreasing functions at infinity. If we consider another image g_2, identical to the previous one but a for a rigid rotation around the origin by an angle $\Delta\vartheta$, the new density function in polar coordinates is obviously

$$f_2(\rho, \vartheta) = f_2(\rho, \vartheta + \Delta\vartheta) = \sum_n a_{2n}(\rho)\, e^{in\vartheta} \tag{4}$$

with

$$a_{2n}(\rho) = (1/2\pi) \int_0^{2\pi} f_2(\rho, \vartheta + \Delta\vartheta)\, e^{-in\vartheta}\, d\vartheta = a_{1n}(\rho)\, e^{in\Delta\vartheta} \tag{5}$$

Equation (5) suggests a straightforward technique for evaluating the angular shift $\Delta\vartheta$. Let us consider the integrals

$$A_{1n} = \int_0^{\infty} a_{1n}(\rho)\, \rho\, d\rho \tag{6}$$

and

$$A_{2n} = \int_0^\infty a_{2n}(\rho)\,\rho\,d\rho = e^{jn\Delta\vartheta} \int_0^\infty a_{1n}(\rho)\,\rho\,d\rho \tag{7}$$

It is easy to see that

$$|B_n| = |A_{2n}/A_{1n}| = 1 \quad ; \qquad \arg(B_n) = \arg(A_{2n}/A_{1n}) = n\Delta\vartheta \tag{8}$$

In order to evaluate A_{1n} and A_{2n}, however, we do not need explicitely the Fourier coefficients; using definitions (3), we obtain directly

$$A_{1n} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^\infty f_1(\rho, \vartheta)\, e^{-jn\vartheta}\, \rho\, d\rho\, d\vartheta \tag{9}$$

$$A_{2n} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^\infty f_2(\rho, \vartheta)\, e^{-jn\vartheta}\, \rho\, d\rho\, d\vartheta \tag{10}$$

In cartesian coordinates, equation (9) becomes

$$A_{1n} = \frac{1}{2\pi} \iint_{IR2} g_1(x, y) \cos[n \cos^{-1}(x/\sqrt{x^2+y^2})]\, dx\, dy - \frac{1}{2\pi} \iint_{IR2} g_1(x, y) \sin[n \sin^{-1}(y/\sqrt{x^2+y^2})]\, dx\, dy \tag{11}$$

A similar expression holds for A_{2n}.

It is worth noting that equation (11) allows a direct evaluation of coefficients A_{1n} and A_{2n} without any intermediate transformation from cartesian to polar coordinates. We will give later on details about the numerical evaluation of integrals (11), but since now we point out that we will deal with picture elements defined over a rectangular rnesh.

Equation (8) provides in principle, a complete resolution of the problem. In practice, the complex numbers B_n are affected by unpredictable phase errors (we can avoid amplitude errors by a normalization), and it is convenient to determine the unknown $\Delta\vartheta$ by a least square fit. If the B_n's have been evaluated for $n = 0, 1, \ldots, N-1$, we look for the real number $\Delta\vartheta$ which minimizes

$$e(\Delta\vartheta) = \sum_{n=-(N-1)}^{N-1} |B_n - e^{jn\Delta\vartheta}|^2 \tag{12}$$

where $B_{-n} = B_n^*$.

Now, problem (12) is nonlinear and its solution may be somewhat difficult (the idea of considering the arguments of the complex numbers involved in the sum and trying a linear regression is misleading, because of the periodicity of the complex exponential). Alternatively, we can consider the problem in the domain of the variable ϑ. If we define the functions

$$G(\vartheta) = (2N-1)^{-1} \sum_{n=-(N-1)}^{N-1} B_n e^{jn\vartheta} \tag{13}$$

and

$$F(\vartheta) = (2N-1)^{-1} \sum_{n=-(N-1)}^{N-1} e^{jn\vartheta} \tag{14}$$

It is easy to see that problem (12) is completely equivalent to minimizing

$$E(\Delta\vartheta) = \int_0^{2\pi} |G(\vartheta) - F(\vartheta + \Delta\vartheta)|^2 d\vartheta \tag{15}$$

being

$$E(\Delta\vartheta) = (2N-1)^{-2}\, e(\Delta\vartheta) \tag{16}$$

In the formulation (15), the minimization problem can be given an approximate (though very accurate) solution at an acceptable computational cost. Let us recall that

$$F(\vartheta) = (2N-1)^{-1} \frac{\sin[(2N-1)\vartheta/2]}{\sin(\vartheta/2)}, \quad \vartheta \neq 0 \; ; \qquad F(0) = 1 . \tag{17}$$

According to (17), $F(\vartheta)$ is a continuous function with an absolute maximum, equal to 1, at $\vartheta = 0$. Furthermore, if $N \gg 1$ at any other extremum point $\vartheta' \neq 0$, we have

$$|F(\vartheta')| \leqslant C \simeq 1/\pi \tag{18}$$

and therefore the detection of the absolute maximum of $F(\vartheta + \Delta\vartheta)$ is not difficult. Now, if we assume that $G(\vartheta)$ presents a similar shape, an approximate value of $\Delta\vartheta$ can by found out by searching for the absolute maximum of $G(\vartheta)$ itself. It is worth noting, however, that $G(\vartheta)$ would be a shifted version of $F(\vartheta)$ only in the ideal case, where the coefficients B_n have been exactly evaluated. In practice, any B_n - except, of course, B_0 - will be affected by unpredictable phase errors, so that we could expect that even the shape of $G(\vartheta)$ will be distorded. A straightfourward, but cumbersome, analysis shows that - at the first degree of approximation - such disturbances do not affect the position of the maximum point of $G(\vartheta)$. The detailed computations will not be reported; we only recall that such statistical analysis can be carried out along the guidelines established in [9]. On the other hand, numerical experiments confirm the remarkable robustness of the present algorithm, as we will discuss in section 5.

3. THE NUMERICAL PROCEDURE

According to eq.n (13), we have only to compute the coefficients B_n of the Fourier representation of $G(\vartheta)$. These may be determined using a recursive procedure to evaluate the integrals (11). Let us define:

$$I_n = \iint_D g(x, y) \cos[n \cos^{-1}(x/\sqrt{x^2+y^2})]\, dx\, dy \tag{19}$$

$$J_n = \iint_D g(x, y) \sin[n \sin^{-1}(y/\sqrt{x^2+y^2})]\, dx\, dy \tag{20}$$

where D denotes the (finite) domain of integration corresponding to the actual image. In order to evaluate I_n and J_n, $n = 0, 1, \ldots$ we will use certain properties of the weight functions T_n and V_n:

$$T_n(x/\sqrt{x^2+y^2}) = \cos[n \cos^{-1}(x/\sqrt{x^2+y^2})] \tag{21}$$

$$V_n(y/\sqrt{x^2+y^2}) = \sin[n \sin^{-1}(y/\sqrt{x^2+y^2})] \tag{22}$$

The first one, $T_n(\cdot)$, is simply the n-th order Chebyshev polynomial of the first kind, whilst $V_n(\cdot)$ is related to the second kind Chebyshev polynomial.

It is easy to show, using elementary formulae for trigonometric functions, that the following recurrence relations hold:

$$T_n\left(\frac{x}{\sqrt{x^2+y^2}}\right) = \frac{x}{x^2+y^2}\, T_{n-1}\left(\frac{x}{\sqrt{x^2+y^2}}\right) - \frac{y}{x^2+y^2}\, V_{n-1}\left(\frac{y}{\sqrt{x^2+y^2}}\right) \tag{23}$$

$$V_n\left(\frac{y}{\sqrt{x^2+y^2}}\right) = \frac{y}{x^2+y^2}\, T_{n-1}\left(\frac{x}{\sqrt{x^2+y^2}}\right) + \frac{x}{x^2+y^2}\, V_{n-1}\left(\frac{y}{\sqrt{x^2+y^2}}\right) \tag{24}$$

Furthermore, we have

$$T_0(x/\sqrt{x^2+y^2}) = 1 \quad ; \quad V_0(y/\sqrt{x^2+y^2}) = 0\,, \tag{25}$$

by which we can start the recursive evalutation of $T_n(\cdot)$ and $V_n(\cdot)$ for any n. Now, we consider the discrete counterparts of eq.n (19) and (20); assuming a rectangular mesh over the domain D, I_n and J_n can be estimated (apart from a non relevant proportionality constant) as:

$$I_n \simeq \sum_{h,k} g_{hk}\, T_n(a_{hk}) \tag{26}$$

$$J_n \simeq \sum_{h,k} g_{hk}\, V_n(b_{hk}) \tag{27}$$

where

$$g_{hk} = g(x_h, y_k) \quad ; \quad a_{hk} = x_h/\sqrt{x_h^2+y_k^2} \quad ; \quad b_{hk} = y_k/\sqrt{x_h^2+y_k^2}$$

Using (23) and (24), (26) and (27) become

$$I_n \simeq \sum_{h,k} g_{hk}\,(a_{hk}\, T_{n-1}(a_{hk}) - b_{hk}\, V_{n-1}(b_{hk})) \tag{28}$$

$$J_n \simeq \sum_{h,k} g_{hk}\,(b_{hk}\, T_{n-1}(a_{hk}) + a_{hk}\, V_{n-1}(b_{hk})) \tag{29}$$

Hence we can deduce an efficient computational scheme, formally, we rewrite

$$I_n \simeq \sum_{h,k} a_{hk}\{g_{hk}\, T_{n-1}(a_{hk})\} - \sum_{h,k} b_{hk}\{g_{hk}\, V_{n-1}(b_{hk})\} \tag{30}$$

and

$$J_n \simeq \sum_{h,k} b_{hk}\{g_{hk}\, T_{n-1}(a_{hk})\} + \sum_{h,k} a_{hk}\{g_{hk}\, V_{n-1}(b_{hk})\} \tag{31}$$

The terms in braces are exactly the components which should be added to evalutate I_{n-1} and J_{n-1}. Thus we need only two-dimensional arrays, say $\mathscr{A}$ and $\mathscr{B}$, with initial values

$$\mathscr{A}^{(0)}_{hk} = g_{hk} \quad \text{and} \quad \mathscr{B}^{(0)}_{hk} = 0\,.$$

At any step, we have to perform the following operations:
(i) Evaluate:

$$I_n = \sum_{h,k} \mathscr{A}^{(n)}_{hk} \quad \text{and} \quad J_n = \sum_{h,k} \mathscr{B}^{(n)}_{hk} \tag{32}$$

(ii) Update:

$$\begin{bmatrix} \mathscr{A}^{(n+1)}_{hk} \\ \mathscr{B}^{(n+1)}_{hk} \end{bmatrix} = \begin{bmatrix} a_{hk} & -b_{hk} \\ b_{hk} & a_{hk} \end{bmatrix} \begin{bmatrix} \mathscr{A}^{(n)}_{hk} \\ \mathscr{B}^{(n)}_{hk} \end{bmatrix} \tag{33}$$

The matrix involved in the updating scheme is a "rotation matrix", and its norm is equal to 1. This may produce an unwanted propagation of rounding errors if the computation is to be repeated many times. Such a difficulty, however, may be avoided by a periodic renormalization of the computed elements of matrices $\mathscr{A}$ and $\mathscr{B}$.

The whole computational procedure can be summarized as follows:

1) For a fixed N, compute Bn, $n = 0, 1, \ldots N-1$. Recall that $B_{-n} = B_n^*$.
2) Once $B_n = u_n + jv_n$ is known, $G(\vartheta)$ is immediately evaluated as

$$G(\vartheta) = \sum_{n=0}^{N-1} [u_n \cos(n\vartheta) - v_n \sin(n\vartheta) \tag{34}$$

and its maximum point can be found out e.g. by a simple bisection algorithm, preceeded by a coarse scanning of the ϑ-axis.

4. REGISTRATION OF ROTATED AND TRANSLATED IMAGES

The algorithm described in the previous section provides an efficient tool for the registration of rotated and translated images. Let $i_1(\underline{P})$, $\underline{P} = [x, y]^T$ be the reference image, and $i_2(\underline{P})$ a replica translated by $\underline{P}_0 = [x_0, y_0]^T$ and rotated by $\Delta\vartheta$, so that

$$i_2(\underline{P}) = i_1(\underline{\underline{R}} \cdot \underline{P} - \underline{P}_0) \tag{35}$$

where $\underline{\underline{R}}$ is the rotation matrix

$$\underline{\underline{R}} = \begin{bmatrix} \cos\Delta\vartheta & \sin\Delta\vartheta \\ -\sin\Delta\vartheta & \cos\Delta\vartheta \end{bmatrix} \tag{36}$$

Taking the Fourier transform $I_1(\underline{\xi}) = \mathscr{F}\{i_1(\underline{P})\}$ and $I_2(\underline{\xi}) = \mathscr{F}\{i_2(\underline{P})\}$, according to the shift and rotation theorems [10] it turns out that

$$I_2(\underline{\xi}) = e^{-i\underline{P}_0 \cdot (\underline{\underline{R}}\,\underline{\xi})} \quad I_1(\underline{\underline{R}}\,\underline{\xi}) \tag{37}$$

Therefore

$$|I_2(\underline{\xi})| = |I_1(\underline{\underline{R}}\,\underline{\xi})| \tag{38}$$

i.e., the magnitude of the second image transform is but the magnitude of the first image transform, referred to a new system of frequency coordinates rotated by the same angle by which i_2 is rotated with respect to i_1. Therefore, $\Delta\vartheta$ may be determined by simply comparing the magnitudes of I_2 and I_1 according to the algorithm discussed in sections 2 and 3.

Once ϑ is known the displacement P_0 may be determined by comparing the phases of I_2 and I_1 [8, 9]. According to eq. (37),

$$e^{j\angle I_2(\xi)} = e^{-i\underline{P}_0 \cdot (\underline{\underline{R}}\,\underline{\xi})}\ e^{j\angle I_1(\underline{\underline{R}}\,\underline{\xi})} \tag{39}$$

From the values of $I_1(\underline{\xi})$, known at the nodes of a rectangular mesh in the frequency plane, it is possible to derive, by suitable interpolation, the values corresponding to $I_1(\underline{\underline{R}}\,\underline{\xi})$, i.e. the values of I_1 at the nodes of a mesh rotated by $\Delta\vartheta$. It is then easy to evaluate

$$e^{j\{\angle I_2(\xi) - \angle I_1(\underline{\underline{R}}\,\underline{\xi})\}} = e^{-i\underline{P}_0 \cdot (\underline{\underline{R}}\,\underline{\xi})} = e^{-i(\underline{\underline{R}}^{-1}\underline{P}_0)\cdot\underline{\xi}} \tag{40}$$

the inverse transform of which yields a δ-distribution centered at $\underline{Q}_0 = \underline{\underline{R}}^{-1}\,\underline{P}_0$ in the continuous case, a less than unity peak in the discrete case, if noise and other disturbances an taken into account. Once, by looking for the maximum of the IFT of (40), $\underline{Q}_0$ is found, $\underline{P}_0$ is easily determined as $\underline{P}_0 = \underline{\underline{R}}\,\underline{Q}_0$.

5. EXPERIMENTAL RESULTS

The algorithm was tested on simple 128 x 128 synthetic images of the type shown in fig.s 1a, b. Proper sampling of the images is fairly important in order to avoid aliasing effects, which degrade the magnitude spectrum. The images

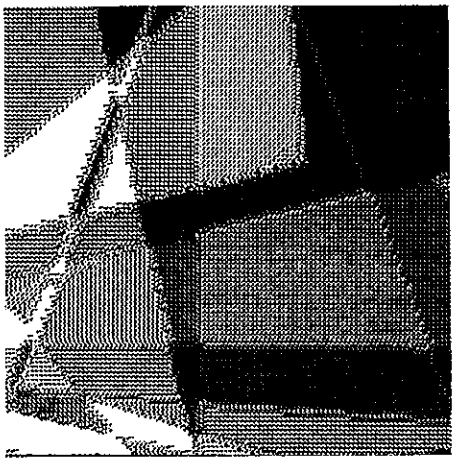


Fig. 1a — Reference image.

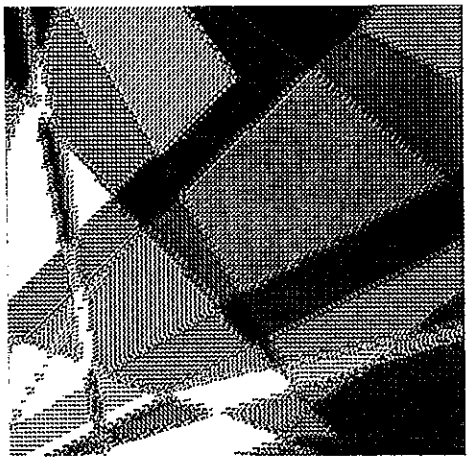


Fig. 1b — Rotated (30°) and translated (8, 15) image.

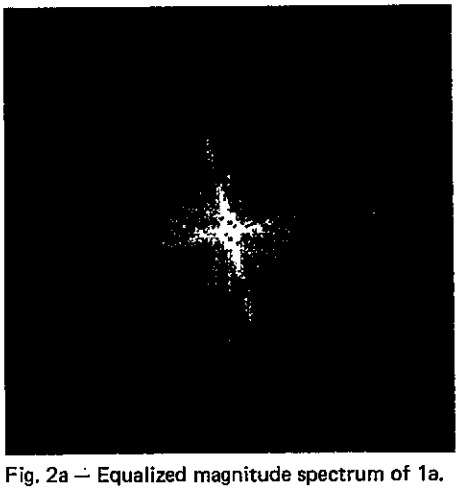


Fig. 2a — Equalized magnitude spectrum of 1a.

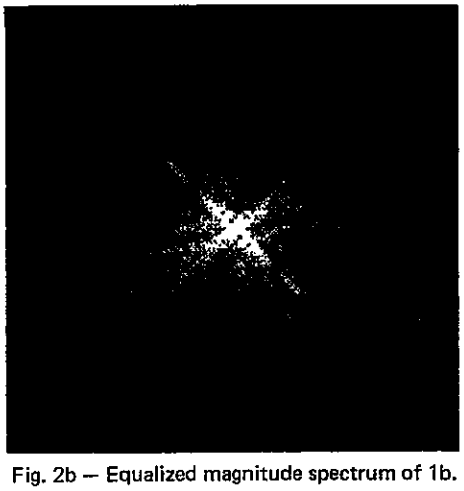


Fig. 2b — Equalized magnitude spectrum of 1b.

were windowed with a gaussian window with $\sigma = 30$ to reduce discontinuities at the frame boundary and thus the related orthogonal cross centered at the origin of the magntude spectrum. Once the magnitude spectrum is computed, it is very important to discord noise components, while enhancing the information useful for detecting the rotation. In practice, we found convenient to perform coarse (17 levels) histogram equalization of the magnitude spectrum, the result of which is shown in fig.s 2a, b. Then, the algorithm described in sections 3 and 4 allowed a conect estimation of both rotation and shift. Work is now in progress to compare various interpolation procedures for the evalutation of $I_1(R\,\xi)$.

REFERENCES

[1] P.E. Anuta, *Spatial registration of multispectral and multitemporal digital imagery using FF techniques,* IEEE Tr. Geosc. Electron., GE-8 (1980) 353 - 368.

[2] D. Casasent, *Pattern recognition: a review,* IEEE Spectrum, March 1981, 28 - 33.

[3] G. Hofele, K. Luetjen, E. Reposi: *Electronic stabilization of images within image sequences,* Proc. 6 Int. Conf. on Pattern Recognition, Munich, Oct. 19 - 22, IEEE Computer Society Press, New York, (1980) 313 - 315.

[4] G. Cristini, E. De Castro, A. Martelli, C. Morandi, M. Vascotto: *Feasibility of an electronic ophthalmoscope with compensation of random eye motion,* Proc. Int. Conf. on *Adavance in Image Processing and Pattern Recognition,* Pisa (I), Dec. 10 - 12, North Holland, (1985).

[5] R.A. Emmert, C.D. McGillem: *Multitemporal Geometric Distortion Correction Utilizing the Affine Transformation,* Proc. IEEE Conf. on Machine Processing of Remotely Sensed Data, Oct. 16 - 18 (1973) 1B-24-1B-32.

[6] D.I. Barnea, H.F. Silverman: *A class of algorithms for fast digital image registration,* IEEE Tr. Computers, C-21 (1982) 179-186.

[7] C. Cafforio, F. Rocca: *Tracking moving objects in television images,* Signal Processing, (1979), 133-140.

[8] C.D. Kuglin, D.C. Hines: *The phase correlation image alignment method,* Proc. of the IEEE 1975 Int. Conf. on Cybernetics and Society, (1975), 163 - 165.

[9] S. Alliney, C. Morandi: *Digital image registration using projections,* IEEE Tr. PAMI, March 1986.

[10] A. Papoulis: *Systems and Transforms with Application in Optics,* McGraw-Hill, New York, (1968), 90.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

M.17

DESIGN OF A TEXTURE FEATURES EXTRACTOR DEDICATED PROCESSOR

G. Ouvradou, D. Barba (§)

E.N.S.T.b.r. boite postale 832 29285 BREST cedex (FRANCE)

This paper describs a current study dealing with the hardware implementation of a new and original method of texture characterization. A list of specific features of this method is followed by a short presentation of the reasons for undertaking this study and the goals we set ourself. Before dealing with the algorithm aspects of the method, the paper gives an informal description of the latter. The problems are expressed in terms of choosing an architecture which is adapted to the algorithm and which corresponds to the internal criteria of the study. This is followed by a presentation of the structure and the control scheduling of the processing unit which shows how a problem of data dependence is solved. The conclusion briefly sums up the work to date and defines future lines of study.

1. INTRODUCTION

The algorithm we intend to implement was named " the curvilinear method " by its designers Barba and Ronsin, and was presented at EUSIPCO 83 [1].

Many methods have already been tested in the field of image segmentation by texture characterization; this new one offers some very interesting features:

- a segmentation efficiency similar to that of approved methods [2];
- it is not specifically related to a particular texture type, and takes into account the notion of contour;
- moderate computation expense as compared with classical methods.

Thanks to these remarkable features, we are considering the extension of this method into areas of image analysis requiring short response times, such as robotics or coding.

We therefore investigated the implementation of the algorithm with two main objectives in mind: high processing speed and low implementation costs. Obviously, the latter criterion precludes the use of a powerful image processor as well as the expensive computer configuration it needs. We therefore decided to design a special purpose processor. The characterisation of a TV picture (512 X 512 X 8) has to be performed in a few hundred milliseconds, which is close to a human observer's response time.

In comparison with already completed implementations on 16-bit-minicomputers, the speed would be improved by a factor at least one thousand.

We should like to point out that we have chosen to direct this study towards the methodological problems raised by the development and implemen-tation of parallelization techniques. Consequently, our first step will not be to search for a circuit or a family of dedicated circuits to solve our problem, but for an original architecture.

2. REVIEW OF THE ALGORITHMIC ASPECTS

Let us recall the principle of the texture characterization method (a formal description is given in [1].

Let us consider a continuous space composed of the picture plane and its spatial distribution of luminance $z(x,y)$. Texture is characterized by associating a vector to each point in the picture. The components of this vector represent a measurement of the luminance profile in the neighborhood of this point, for various scan directions.

This measurement is computed as the luminance profile length starting from the point. This involves a sequence of elementary displacements dx' on the picture plane in the selected directions. When the current profile length reaches a predetermined value μ, the measurement is defined as the distance covered in the picture during scanning. Computing ds as a length element of the luminance profile, introduces a scaling factor λ. The length element may then be computed as

$$ds^2 = \lambda^2 dx'^2 + dz^2 \qquad (1)$$

The method makes use of three parameters μ, λ

§ I.R.E.S.T.E. 3 rue du Maréchal Joffre 44041 NANTES cedex (FRANCE).

and angular resolution $\Delta\theta=2\pi/n$. The first two parameters allow the range of the characterization to be centered according to the contrast and scale of the texture to be characterized. With regard to the angular resolution, experimentations showed [3] that results are satisfying and stable for n=8 ($\Delta\theta$= 45°), as far as the quality of the segmentation is concerned.
The following paragraph describes the transposition of the method in order to implement it upon digital pictures.

2.1. Adaptation to discretized pictures

A square sampling structure was selected. Since the method proved to be robust during experimentations, we avoided the interpolation problem due to the diagonal directions by selecting an apprpriate topology.
Let P1 and P2 be two pixel of the image with rectangular coordinates (i1,j1) and (i2,j2) respectively ; the distance between P1 and P2 is defined as

$$d(P1,P2)= MAX(|i2-i1|,|j2-j1|)$$

We can thus transpose the expression (1) into a dicret space context, given an angular resolution of $\pi/4$

$$\Delta s^2 = \lambda^2+\Delta z^2 \qquad (2)$$

2.2. Search for an efficient algorithm

Two main aspects were taken into account in order to minimize the potential complexity of implementation.
The first one is related to the basic principle of the method. It became clear that the scan distance around a point increases with the reduction in texture of the area to which the point belongs. This type of area greatly increases calculating time but the characterization offers little information in return. A new parameter kmx was therefore introduced. Its function is to anticipate the end of the component computation when the distance covered in the image plane reaches the value kmx. The effectiveness of this test can be inhibited by choosing kmx >μ/λ (scanning distance if there is no signal).
We then investigated the execution scheme of the algorithm. Experimentations on natural pictures [3] supply an estimation of the typical average value of the texture components ktyp =15 (in so called standard conditions: μ=200, λ=4). This lead to the conclusion that, if the components are independently computed, each one need an average of fifteen iterations. In fact, it is possible to take advantage of the correlation between colinear components of pixel that are located on the same scan path. In this context the elements Δs involved in the component computation of two adjacent pixel are in common for the greatest part.
The algorithm which we apply (lack of space accounts for our not presenting it here), enables us to produce p colinear components by having access to q luminance data, so that $p+1\leqslant q\leqslant p+kmx$ on average $q \simeq p + ktyp$. Independent computation of the components would result in $q \simeq p.ktyp$.

3. PROCESSOR ARCHITECTURE

Inspite of the extended literature concerning the field of parallel computers [4,5,6,7,8], it is difficult to discern any coherent lines of research as far as a design methodology is concerned. The principal reason for this lies in the great variety of techniques of parallelism implementation that have been developped. Pratically all of these techniques however are based on the following three fundamental principles:

i) pipelining
ii) processor replication
iii) functional parallelism.

Therefore, two questions would appear to establish a link between these fundamental principles and major parallel processing schemes used by computer designers:
Would the solution of the problem be facilitated by

i) a partition into sequential subproblems or a data partition for parallel processing ?
ii) a strictly synchronous execution of parallel tasks or a free running one with interlocked cycles time to time ?

The answers are neither binary nor independent, but they might serve as general guidelines. Thus, if the response to the first question favours data partition, the dedicated processing scheme will be of SIMD or MIMD type. A division into sequential subproblems, on the other hand, would require a PIPELINE type solution. Moreover a SYSTOLIC approach can be applied to both answers or neither and will be preferred for low-level algorithm implementation.
The answer to the second question is related to the different level of parallelism implementation (logical circuit level or phase instruction execution level, for example).
An asynchronous solution necessitates a more elaborate signalization than a synchronous one. In the present case, our answers to the two questions are as follows.
One of the characteristics of our algorithm is that it treats data in vector form. The efficiency of the algorithm increases with the length of the vector. This is a major advantage of pipeline type architecture [9]. The algorithm is however data dependent (the number of iterations necessary to compute a component is not known before hand). This could create serious problems in a pipeline implementation, particulary in the presence of recurrent forms (current curvilinear integral). This problem can only be avoided by applying a procedure that does not exploit the correlation among components. A systolic approach might be considered because this would conserve memory activity.

These architectural approaches restrict our possibilities of answering the second question for indeed the pipeline or systolic concepts are related to a synchronous timing. Further-

more the latter reduces implementation costs and does not create any particular problems within a small system, an aspect which is important given the goals of this study.

4. PROCESSING UNIT WITH PIPELINE STRUCTURE

4.1 Solution of the algorithm data dependence problem

The procedure followed is to divide the execution scheme of the algorithm at the point where the dependence phenomenon arises.
The latter is due to the overflow of the current curvilinear integral (test $s<\mu$) and its iteration index (test $k<kmx$). If during an iteration, both tests are positive, computation of the integral is continued (process a new Δs element). Inversely, it is necessary to output the component the calculation of which has just been terminated (equal k). Before initiating a new computation step, the context of the following pixel at level (k-1) has to be set up.

It will be noticed that this dependence has no effect on the computation procedure. Only the initiation rate of the latter would be affected. Consequently, a static pipeline structure was chosen for the Δs calculus implementation.

The part of the algorithm which is under data dependence requires a different strategy.
The dependence creates a feedback effect in a pipeline execution scheme. However, this must not abort a previously initiated computation as the pipeline efficenty would significantly decrease. For the same reason, it is not useful to limit the initiation rate of components computation steps. Therefore, it is better to minimize the execution time of the loop with the feedback. As a result we will use an extensive functional parallelism.
Given the above, the data-dependent process is managed as follows:
Two processes are maintained during a computing step. Each process takes into account a different hypothesis in the result of the test which terminates the step. When the result is available, the context of the successful process is duplicated and the next step is initiated. This strategy favours the throughput but limits the stages utilisation to 50%. Nonetheless it allows a very regular control scheduling.

The accorded implementation structure can be defined as a mixed type which applies both the functional parallelism principle and the dynamic pipeline principle.
The first principle is intensified by use of parallel type arithmetic components. The second materializes in synchronous circuits interconnected in a dynamically configurable way.

4.2 Design and timing of the processing unit

This unit is made up of two subunits A and B that are connected in pipelined fashion. Their role is to undertake the functional separation described in the previous paragraph.

SUBUNIT A:
It computes $\Delta s = (\lambda^2+\Delta z^2)^{1/2}$
Three pipelined stages with the following functions:
- input latching (z(i),z(i-1))
- computation of $|\Delta z|$ (arithmetic)
- computation of Δs (look-up RAM 256X8)

It allows an initiation rate of one cycle.

SUBUNIT B:
It is equiped with a FIFO memory. This queue makes it possible to hold back the Δs used for the component being computed.
Arithmetic circuits and interconnection paths allow the following sequence:
(notation:
Δsnw = current sample
Δsnw(-) = preceding sample
Δsold = initial sample
T(u,v) boolean function of integers /
T(u,v)=(u AND v kmx)
<..action i,..> = parallel execution)

```
IF, at previous cycle, T(u,v)=TRUE
     DO in present cycle
          <input Δs,load queue,s:=s+Δsnw,
           sdw:=s-Δsold,k:=k+1,kdw:=k-1,T(s,k)>
     ELSE DO in present cycle
          <output(k-1),unload queue,
           s:=sdw+Δsnw(-),sdw:=sdw-Δsold,k:=kdw+1
           kdw:=kdw-1,T(sdw,kdw)>
ENDIF
```

4.3 Input/Output characteristics

A detailed examination offers the following conclusions:
After the start-up sequence, the processing unit either requests a new data item or produces a new texture component.
If no dead cycle in the data stream occurs during the vector switching phase, this rule is upheld.
A simple control stategy manages the start-up and flush sequences of the processing unit.

These characteristics make it possible to connect the processor memory and its processing unit through a synchronous half-duplex channel. Furthermore, the Proc. unit can use the whole channel thoughput in a continuous way during the entire processing of an image. This will allow an optimum use of the processor memory bandwith.

5. CONCLUSION OF THE STUDY AND PROSPECTS FOR THE FUTURE

The algorithmic analysis of this method allowed the definition of a dedicated architecture, particulary because it was possible to define at which level and in which way the parallelism could be introduced. Thus the processing unit structure and timing were specified.
At this point, two main factors determine future trends:

The first is the potential troughput of the processing unit.
The second is the specification of system aspects required to achieve this theoretical power.

The timing characteristics of the processing unit described in the previous section allows us to consider that this goal lies in the not too distant future. The memory and channel controller are presently easy to specify. Before proceeding on to further steps of this study let us sum up by comparing the goals we set with the present state of the study. In order to evaluate the potential troughput of the processing unit, realistic conditions were defined:
in the context of a design with TTL standard components, a cycle time of 100 nanoseconds appeared reasonable. Under these conditions, the processing unit is able to lead an eight directions texture characterization of a 512X512 picture in 360 milliseconds. These results are very encouraging, particularly since a parallel implementation of several processing units appears possible.

REFERENCES

[1] D. Barba, J.Ronsin, "New method in texture analysis in the context of image segmentation" EUSIPCO 83, pp283-286 (Erlangen)

[2] J.Ronsin, D.Barba, S.Raboisson, "Comparison between coocurrence matrices, local histograms and curvilinear integration for texture characterization" Second International Symposium on Optical and Electro-optical applied Sciences and Engineering, Dec.85 (pp 596-15).

[3] D.Barba, J.Ronsin, "Image segmentation using new measure of the texture features", Digital Image Processing 84,pp749-753; (Florence)

[4] J.Kittler , J.B.Duff (Eds), "Image processing system architectures", 1985, RSP.

[5] R.W.Hockney, C.R.Jesshope, "Parallel computers" , 1981,AH-LTD.

[6] A.P.Reeves, "Parallel computer architectures for image processing", Computer Vision Graphics and Image Processing , n°25, pp68-88, 1984.

[7] S.Castan, "Architectures adaptées au traitement d'images", T.S.I., pp431-445, Mai 85.

[8] S.Yalamanchili and al. "Image processing architectures: a taxonomy and survey",in Progress in Pattern Recognition 2, North Holland, 1985.

[9] P.M.Kogge, "The architecture of pipelined computers", Mc Graw-Hill Book, 1981.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Synthesis of a Representation of Local Visual Signals Using a Scale-Invariance Approach

Ireneusz Defée

Department of Electrical Engineering
Tampere University of Technology
P. O. Box 527, SF–33101 Tampere, Finland

A model for the derivation of a representation of local visual signals is presented. After defining local signals, we study the effects of scale changes on their representation. This enables us for the introduction of a filter-bank model for the scale-invariant representation. The structure of a filter bank optimal with respect to visual detection criteria is derived using discrete prolate spheroidal sequences. Using the presented approach the structure of low-level visual tract of both biological and computer vision systems is described in uniform way.

1. INTRODUCTION

The problem of representation of local signal features has received recently considerable attention mainly in the context of short-time Fourier analysis [1, 2, 3]. Representation of local signal features is especially important in vision for explanation of the structure of low-level biological vision and for the synthesis of artificial vision systems. The structure of low-level vision has been attempted for explanation mainly from the point of edge detection and providing optimal resolution in position and frequency domain [4, 5, 6, 7]. It has been however established that low-level biological vision systems have characteristics of spatial frequency analysers [8]. This feature asks for more extensive analysis of these systems in terms of their goals and structure.

In this paper we aim for the derivation of the structure of low-level vision using a scale-invariance approach. We construct first a model for local bandlimited signals and next consider low-level vision as a system for their invariant representation under local scale changes. Analysis of local scale changes enables for the derivation of a bank of filters by which such an invariance may be realized. Specific form of these filters in the discrete case is obtained from the condition of optimal position-frequency resolution expressed by maximal energy concentration in the frequency domain and is given by the discrete prolate spheroidal sequences.

2. BASIC MODEL

In this section we present basic model for the derivation of a structure of low-level vision system. Such a derivation requires first formulation of goals which the system is implementing. In our formulation the goal of visual system is to synthesize invariant representation of local aspects of signals. Visual signals are discrete and bandlimited in nature since they are projected by an optical system, serving as an analog lowpass antialiasing filter, on a photoreceptor matrix.

2.1. Local signals

We define first local signals which will form certain primitives from which global signals will be synthesized. Let $f(x)$ will be a 1-D bandlimited signal and let $f_s(x), f_a(x)$ be its respective symmetric and antysymmetric components. We shall call a signal local if its symmetric component will have only one extremum and antysymmetric component will be monotonic. This definition of local signals is not related to any particular "short" intervals but instead it captures those features which are primitve and correspond to point-like structures in case of symmetric component and edge-like structures in case of antysymmetric component. In practice these features are defined for short observation windows and it is important to observe that each signal may be represented as a running sum of local signals.

2.2. Invariance

We shall turn now to the problem of invariant representation of local signals. Of interest here is invariance with respect to scale changes, analyzed from signal theory point of view. Assume first that the scale of a local signal projected by an optical system on a photoreceptor matrix has been decreased. Then the size of the projected local signal is decreased, but also there is an apparent shift of its spatial frequency content towards higher frequencies. This may lead to the loss of information about the signal when certain frequency band becomes higher than cutoff frequency of optical lowpass filter. Quite similarly when the scale of a signal is increased there is an apparent shift of its spatial frequency content towards lower frequencies. This may lead to the increase of information about the signal. The increase is described by additional frequency band which is now lower than cutoff frequency of the lowpass filter.

Suppose now that certain frequency band ΔF is shifted shifted by certain factor c up or down the frequency scale. Then it is easy to see that the ratio of band width to the center frequency is constant - in other words this is

constant Q frequency band shift.

We see form this analysis that to describe the effects of scale changes properly we need a system which operates both in position and spatial frequency domain, detecting changes in spatial frequency content and physical size of signals.

3. FILTERING CASCADE

In order to built a system for invariant representation of local signals we shall divide spatial frequency spectrum into a set of frequency bands. The number of these bands will be depending on scale resolution i.e. sensitivity of the system for the detection of smallest scale changes. Another prerequisite is the realization of optimal position resolution of local signals. In the discrete case both these conditions are best satisfied by filters based on the discrete prolate spheroidal sequences [9]. The discrete prolate spheroidal filter [9] serves as a prototype filter in the design of filtering cascade for the invariant representation. Let $\phi_0(c,\omega)$ denotes such a filter for certain value of position-bandwidth product c. Now we select a set of center frequencies $\omega_1,\ldots,\omega_k$ and place at each center frequency a prototype filter with band width rescaled to preserve constant Q. This procedure is similar to the critical band synthesis performed in [10] and shown there are possibilities for center frequency filter spacings. The sum of subsequent filter outputs in the cascade constitute made low pass filters, outputs from which constitute optimal filtering cascade for symmetric local bandlimited signals. Outputs from bandpass filters made optimal filtering cascade for antysymmetric components of local signals.

The last stage which is needed for building invariant representation of local signals is network of local detectors. Within the resolution limits of the filtering cascade, a set of level detectors is connected to filter outputs. Next, outputs from those detectors which correspond to shifted filter ouputs are connected together. In this way, scale-invariant representation is realized.

4. CONCLUSION

In this paper, we have addressed the problem of an invariant representation of local signals in low-level vision. It was shown that such a representation requires a filtering cascade, optimized also for the position resolution. Our approach gives explanation for the filtering structures found in biological vision systems [8, 11] and serves as a basis for the synthesis of high-quality computer vision. At the same time the usual approach from the point of view of edge detection is extended to the more general local signal representation problem. Of further interest is extension to the representation of explicit 2-D local signals, especially local orientations. This may be done by suitable rearranging of filters in 2-D position plane, covering the same spectrum, using logarithmic transformation and converting the orientation detection problem into a linear shift invariant filtering problem. The result of this operation should be a structure algorithmically. similar to the retina - lateral geniculate nuclei - striate cortex in the visual tract.

REFERENCES

[1] Allen, J.B. and Rabiner, L.R., "A unified approach to short-time Fourier analysis and synthesis," *Proc. IEEE*, vol. 65, pp. 1558-1564 Nov. 1977.

[2] Portnoff, M.B. "Representation of digital signals and systems based on short-time Fourier analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, no.3, pp. 55-69, Feb. 1980.

[3] Allen J.B., "Short-term spectral analysis and synthesis and modification by discrete Fourier transform," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, no.3, pp. 235-238, June 1977.

[4] Shanmugam, K.S. et al., "An optimal frequency domain filter for edge detection detection in digital pictures," *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. PAMI-1, no.1, pp. 37-49, Jan. 1979.

[5] Jernigan, M.E. and Wardell, R.W., "Does the eye contain optimal edge detection mechanisms?," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-11, no.2, pp. 441-444, Feb. 1981.

[6] Marr, D. and Hidreth, E., "Theory of edge detection," *Proc. Royal Soc. London B*, vol. 207, pp. 187-217, 1980.

[7] Marčelja, S., "Mathematical description of the responses of simple cortical cells," *J. Opt. Soc. Am.*, vol. 70, pp. 1297-1300, 1980.

[8] Maffei, L. and Fiorentini A., "The visual cortex as a spatial frequency analyser," *Vision Res.*, vol. 13, pp. 1255-1267, 1973.

[9] Mattehws, J.D. et al., "The discrete prolate spheroidal filter as a digital signal processing tool," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1471-1478, no. 6, Dec. 1985.

[10] Petersen, T.L, and Boll, S.F., "Critical band analysis - synthesis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 656-663, no. 3, June 1983.

[11] Marčelja, S., "Initial Processing of visual information within the retina and the LGN," *Biol. Cybernetics*, vol. 32, pp. 217-226, 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A RANDOM FIELD MODEL BASED ALGORITHM FOR TEXTURED IMAGE SEGMENTATION*

J. E. Bevington and R. M. Mersereau

School of Electrical Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

A split/merge type algorithm is being developed for the task of textured image segmentation. The algorithm is based on the use of Gaussian random fields to model the image surface within disjoint regions. A priori knowledge of model parameters is not assumed. A Maximum-Likelihood estimation procedure is used to define boundaries for region splitting, and probability of error based distance measures are used for merge decisions. Some preliminary results are presented.

1. INTRODUCTION

Image segmentation is considered by many to be the first stage in a general image understanding system. The goal of segmentation is to group pixels into regions in such a way that the regions correspond to visually distinct entities, such as objects or surfaces in the original image. Region information is then sent to "higher level" processes which use scene-specific knowledge to perform object identification.

Over the past 10 to 15 years a great many segmentation algorithms have been proposed (see [1] for an overview), but the problem is still far from being solved. One of the problems in segmentation is in dealing with texture. Roughly speaking, a textured region is one in which various properties of individual pixels (e.g. intensity, hue, etc.) vary widely, but in which relationships of these properties among neighboring pixels are such that humans perceive the region as a whole. Examples include a patch of grass, a brick wall, a pile of pebbles, a patch of sand, etc. The fundamental problem in segmentation, however, is that we do not quantitatively understand the rules or mechanisms by which human beings perform the low level grouping which we are trying to emulate. Until these phenomena are understood, there will be limits on what segmentation algorithms can accomplish.

In this work we are concerned with the problem of segmenting textured monochrome images about which we have no a priori knowledge. In some sense this is an unsupervised learning problem in pattern recognition to which some of the standard clustering techniques might apply. It differs from the standard problem, though, in that spatial relationships among data to be classified must be taken into account. In the segmentation algorithm described below, statistical models are used to characterize the image surface within each of the "regions" (or, in a sense, clusters). The statistical models are homogeneous Gaussian random fields and provide a means of dealing with texture. A decision as to whether or not two smaller regions are similar enough to be merged to form a larger region is made strictly according to a distance measure defined in terms of the respective statistical model parameters. Individual pixels are never considered as separate entities; initial regions are formed by splitting arbitrarily selected square frames using an approximate maximum likelihood (ML) estimation procedure. In the current scheme the number of final regions is preset.

We note here that the more successful segmentation algorithms (e.g. [2]) make use of cues other than average region gray level statistics to make splitting and merging decisions. Such heuristics include region size, region shape, and contrast along boundaries. While it is appropriate to consider such factors (since humans probably use them), for the present we restrict ourselves to the more tractable distance-based decisions.

* This work was supported by the Joint Services Electronics Program under Contract DAAG29-84-K-0024.

2. THE SEGMENTATION ALGORITHM

2.1. *General Description*

The segmentation algorithm is a relatively straightforward split/merge procedure. The image is first divided into square frames (typically of size 32×32 or 16×16 pixels), which serve as the initial regions. Next, each region is split into two or more regions according to the results of applying the ML boundary detection algorithm described below. Adjacent pairs of regions are then merged until the total region count drops to a prespecified number.

The order in which regions are merged is determined by the statistical distance measures for the various region-neighbor pairs. The pair of regions merged at each step is the pair for which this distance is smallest (i.e., the pair for which the statistical models are most similar). Throughout the course of the segmentation procedure, a list is maintained for each region giving the identities of the neighboring regions along with the associated statistical distances. The list is ordered according to the distance, and the list is updated whenever necessary due to a merging or splitting operation. Locating the most similar neighboring region pair at any step in the process is then simply a matter of examining the first record on the neighbor list of each of the current regions.

2.2. *ML Boundary Finder*

The Maximum-Likelihood boundary estimation algorithm used in the region splitting procedure is an outgrowth of an algorithm first described in [3]. For completeness we will first describe the original algorithm (which we will call the *basic procedure*), then describe the extensions currently employed.

Basic Procedure

Given an $M \times N$ array of samples, we assume that each sample is taken from one of two known Gaussian random fields. We assume that the shapes of the subregions corresponding to the two random fields are such that 1) the subregions are simply connected and 2) a left-right partition of the array exists. More precisely, let $x[i,j]$, $1 \le i \le M$, $1 \le j \le N$, be an array of image samples, and let $Z_1[i,j]$ and $Z_2[i,j]$ be discrete, homogeneous, Gaussian random fields with known means and covariances. Let $C[i,j]$ be a binary valued indicator function, defined for $1 \le i \le M$ and $1 \le j \le N$, such that

$$C[i,j] = \begin{cases} 1, & X[i,j] = Z_1[i,j] \\ 2, & X[i,j] = Z_2[i,j]. \end{cases} \tag{1}$$

We impose the following constraint on $C[i,j]$: For each i, $1 \le i \le M$, there exists a number $b[i]$ such that

$$0 \le b[i] \le N,$$

$$C[i,j] = 1 \quad \text{for} \quad j \le b[i],$$

and

$$C[i,j] = 2 \quad \text{for} \quad j > b[i] .$$

This constraint essentially requires that the boundary between region 1 (process Z_1) and region 2 (process Z_2) run vertically through the sample array such that it is a function of the first array index. The problem is now to estimate $b[i]$. For notational convenience below we will sometimes refer to the function $b[i]$ as the vector $\underline{b}$.

The maximum likelihood approach to the problem is to find that $\underline{b}$ which, for an observed set of samples $\underline{x}$, maximizes $p(\underline{x} \mid \underline{b})$, or, equivalently, minimizes

$$\Lambda(\underline{x} \mid \underline{b}) = -c_1 \log p(\underline{x} \mid \underline{b}) + c_2 , \tag{2}$$

where $p(\underline{x} \mid \underline{b})$ is the probability density of $\underline{x}$ conditioned on $\underline{b}$, and c_1 and c_2 are arbitrary constants with $c_1 > 0$. It was shown in [3] that an approximation for $\Lambda(\underline{x} \mid \underline{b})$ can be computed through the use of 2-D linear prediction. The result is given by

$$\Lambda'(\underline{x} \mid \underline{b}) = \sum_{i=1}^{M} \Lambda'_i(\underline{x} \mid \underline{b})$$

$$= \sum_{i=1}^{M} \left\{ \sum_{j=1}^{b[i]} \{e_1^2[i,j]/\sigma_1^2 + \log \sigma_1^2\} + \sum_{j=b[i]+1}^{N} \{e_2^2[i,j]/\sigma_2^2 + \log \sigma_2^2\} \right\} \tag{3}$$

where

$$e_k[i,j] = x[i,j] - u_k - \sum_{(m,n) \in Q_k} a_k[m,n](x[i-m,j-n] - u_k)$$

$$u_k = E\{Z_k[i,j]\},$$

$$\sigma_k^2 = E\{e_k^2[i,j]\},$$

$$E\{e_k[i,j]\} = 0 , \quad k = 1,2.$$

The $a_k[m,n]$ are the 2-D linear prediction coefficients for process Z_k, and Q_k is the pre-determined prediction mask coordinate set. Note that the algorithm becomes quite simple if white noise processes are used as models, in which case the Q_k become empty sets.

It is evident from equation (3) that minimization of Λ' could be carried out on a line by line basis; that is, for each i, choose $b[i]$ which minimizes $\Lambda'_i(\underline{x} \mid \underline{b})$. It has been found necessary in practice, however, to impose smoothness constraints on $\underline{b}$. A constraint of the form

$$|b[i] - b[i+1]| \le k \tag{4}$$

with $k=2$ is being used currently and is easily incorporated into a dynamic programming procedure for minimization of Λ'.

Extensions to the Basic Procedure

Two limitations of the basic boundary estimation procedure must be overcome to make the procedure useful as a general region splitting device:

1) the need for a priori knowledge of model parameters and

2) the restrictions on allowable boundary shape.

The first limitation is addressed with by making the basic procedure part of an iterative scheme in which region statistics and boundary locations are estimated alternately. If no a priori information is available, we start with an arbitrary initial guess for the boundary (usually a straight line through the middle of the initial region), then estimate the statistical model parame-

ters for the resulting sub-regions. Using these model parameters and the basic procedure we estimate a new boundary location, then re-estimate the model parameters using the newly defined sub-regions. The process continues until one of the following occurs:

1) The boundary changes less than a pre-specified amount from the previous iteration,

2) $A'(\underline{x} \mid \underline{b})$ increases,

3) The size of one of the subregions becomes smaller than a pre-determined threshold,

4) A maximum iteration count is exceeded.

If condition 3) occurs, no boundary is output and the original region definition is retained. Conditions 1), 2) or 3) are typically met after 3 or 4 iterations. If the procedure converges, it converges to a local maximum of the joint likelihood function for model parameters and boundary location. Sensitivity of the solution to initial boundary estimate has not yet been investigated.

To overcome the restrictions on boundary shape and direction we run the iterative algorithm in both the horizontal and vertical directions, then combine the results using one of the following two methods. In the first method, the statistical distance between the pair of subregions determined by each of the two boundary finder trials is evaluated, and the partition yielding the greater distance is kept while the other is discarded. In the second method, both computed boundaries are kept, and every closed subregion defined by these boundaries is considered to be a new region. Which of the two methods to use depends on the type of image and the size and shape of regions being sought. The first method may miss sharp corners but will not be sensitive to small scale textural variations. The second method is less sensitive to changes in boundary direction, but may result in many small regions which later must be merged. Small regions may lead to segmentation errors in the presence of coarse texture.

2.3. *Statistical Distance Measures*

The results of the region merging phase of the segmentation procedure are critically dependent on the measure of region similarity or distance employed. Currently, regions are characterized by the parameters of the preselected statistical models, so the distance measure must be defined in terms of these models.

One approach to defining such a measure is to attempt to evaluate the performance (probability of error) of an optimum receiver which decides which model produced a set of observed data, given that the data was in fact produced by one of them. A high probability of error indicates that the models are similar, hence the distance between them should be low, whereas a low probability of error indicates dissimilar regions for which the distance should be high. Probability of error usually cannot be evaluated directly, so some other quantity, hopefully indicative of error probability, is evaluated instead. This approach leads to measures such as the divergence and the Bhattacharyya distance, both discussed in [4]. These measures are general enough that they can be adapted to a wide range of specific models and have the desirable property of being symmetric. The form of these measures for the Gaussian random field models used in our experiments are described below. Other measures, such as those described in [5] and [6], were intended specifically for autoregressive models, but would have to be modified to enable discrimination on the basis of gray level mean and variance, which can be important cues in vision.

The divergence [4] is defined as the difference in the mean values of the log likelihood ratio under the two hypotheses (data from process 1 vs. data from process 2):

$$J = E_1\{\log L(\underline{x})\} - E_2\{\log L(\underline{x})\} \tag{5}$$

where $E_k\{\ \}$ is expected value given hypothesis k and $L(\underline{x})$ is likelihood ratio, given by

$$L(\underline{x}) = \frac{p_1(\underline{x})}{p_2(\underline{x})} \quad . \tag{6}$$

The functions $p_1(\underline{x})$ and $p_2(\underline{x})$ are the conditional densities of the data vector $\underline{x}$ under hypotheses 1 and 2, respectively. For the case where the hypothesized models are white Gaussian random fields with parameters m_1, σ_1, m_2, σ_2 the divergence is given by

$$J = \frac{1}{2}\left[\frac{(m_1-m_2)^2+\sigma_1^2}{\sigma_2^2} + \frac{(m_1-m_2)^2+\sigma_1^2}{\sigma_1^2} - 2\right]. \tag{7}$$

Following the development in [3], a more general (approximate) expression can be derived for the case of colored Gaussian random fields.

The Bhattacharyya distance measure is given by [4]

$$B = -\log \int [p_1(x)p_2(x)]^{1/2}dx \tag{8}$$

where $p_1(x)$ and $p_2(x)$ are the conditional densities of the data under the two hypotheses. For the special case of white noise processes with parameters m_1, σ_1, m_2, and σ_2 the expression becomes [4]

$$B = \frac{1}{4}\frac{(m_1-m_2)^2}{\sigma_1^2+\sigma_2^2} + \frac{1}{2}\log\left[\frac{\sigma_1^2+\sigma_2^2}{2\sigma_1\sigma_2}\right]. \tag{9}$$

Finally, a simple ad hoc measure was tried in some of the segmentation experiments, and often gave results which were better in the sense of agreement with human perception than the probability of error based measures:

$$D = |m_1 - m_2| \tag{10}$$

where m_1 and m_2 are the mean values of regions 1 and 2.

3. RESULTS AND CONCLUSIONS

The segmentation algorithm described above has been implemented for white noise statistical region models and some preliminary results have been obtained. Figure 1 shows a SAR image segmented into 5 regions using the divergence distance measure, while figure 3 shows the "girl" image from the USC data base divided into 25 regions using the ad hoc difference-of-means measure. The results for the SAR image are quite good, while those for the girl image show the limitations of the current algorithm. For complicated images such as the girl image it is clear that our rather simple models and distance measures do not come close to embodying the mechanisms used by human beings.

A bright spot in the results examined thus far is the performance of the region splitting algorithm. When a perceptible binary division of a region does exist, the algorithm is usually able to find it. We are currently examining ways to use edge (discontinuity) information to determine when the splitting algorithm should be applied recursively, so as to further split subregions resulting from the initial split. Edge information can also be used to confirm or refute the presence of a *noticeable* contour along the boundary given by the algorithm.
We believe that the performance of our model based approach could be improved by adding trend information to the models. An obvious difficulty which exists when using a homogenious model for a region which has a significant slope (linear trend) is that the slope component will cause a high estimated variance for the model. The model in this case would produce a region which did not *look* anything like the original, a situation which leads to merging errors no matter how the model-based distance is defined.

It seems likely that the probability of error based distance measures used in this work are not appropriate if the goal is agreement with human perception. This was suggested by superior performance of the ad hoc difference-of-means measure in some of our experiments and is supported by the following argument. Suppose we have two regions such that each has constant gray level (zero variance) and such that the difference in gray levels is some small number ϵ. Any probability of error based distance in this case will be extremely large, whereas it is possible to make ϵ small enough that the difference in the regions is not perceptible.

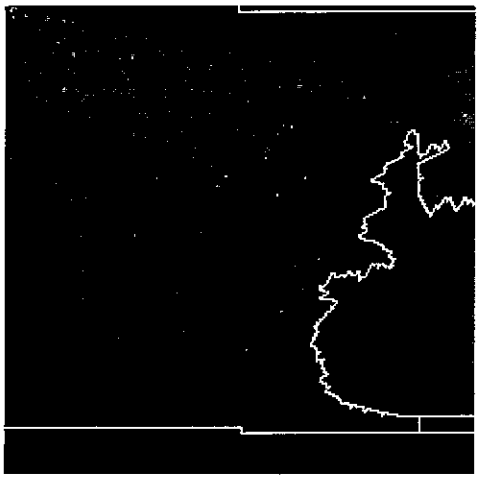


Figure 1

Figure 2

REFERENCES

[1] Haralick, R. M. and Shapiro, L. G., "Image Segmentation Techniques," *Computer Vision, Graphics, and Image Processing*, vol. 29, 1985, pp. 100-132.

[2] Nazif, A. M. and Levine, M. D., "Low-Level Image Segmentation: An Expert System," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. PAMI-6, No. 5, Sept. 1984, pp. 555-557.

[3] Bevington, J. E. and Mersereau, R. M., "A Maximum-Likelihood Approach to Image Segmentation by Texture," *Proc. Int. Conf. Acoust. Speech Sig. Process.*, San Diego, California, March 1984.

[4] Kailath, T., "The Divergence and Bhattacharyya Distance Measures in Signal Selection," *IEEE Trans. Commun. Tech.*, Vol. COM-15, No. 1, Feb. 1967, pp. 52-60.

[5] Itakura, F., "Minimum Prediction Residual Principle Applied to Speech Recognition," *IEEE Trans. Acoust. Speech Sig. Proc.*, Vol. ASSP-23, No. 1, Feb. 1975, pp. 67-72.

[6] DeSouza, P. and Thomson, P. J., "LPC Distance Measures and Statistical Tests with Particular Reference to the Likelihood Ratio," *IEEE Trans. Acoust. Speech Sig. Proc.*, Vol. ASSP-30, No. 2, April 1982, pp. 304-315.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SYNTHESIS OF NATURAL STRUCTURED TEXTURES

P. Volet and M. Kunt

Signal Processing Laboratory, Swiss Federal Institute of Technology,
16, ch. de Bellerive, CH-1007 Lausanne, Switzerland.

This paper presents some new results dealing with a new method for the synthesis of natural structured textures [1] and includes sigificative improvements. This method applies to textures which can be modeled as a combination of a primitive and a placement rule. The improvements concern mainly two steps : the study of the primitive and the reconstitution of the placement rule.

1. INTRODUCTION

Each picture taken in the real world may be segmented into regions separated by contours. These regions correspond to some groups of objects and represent their surface. Generally speaking, the visual appearance of these regions is called texture. The goal of texture synthesis is to reproduce artificially this appearance from a minimum set of parameters. The analysis is performed to extract these parameters from natural textures. Picture coding (for transmission and/or storage) and image synthesis (for flight simulators, TV, movies...) are typical application fields of texture synthesis.

Textures make up a wide class of images. It is necessary to classify them in many categories in order to design some models [2]. This paper is confined to structured textures which can be modeled by a primitive (i.e. a basic shape) repeatedly positioned in the picture plane according to some placement rule. In addition, it is assumed that the primitive does not vary locally and that the placement rule is locally regular. In other words, such a texture is locally periodic and thus the primitive occurs along an hexagonal (eventually a square) grid [3]. However the periodicity and the primitive may present slow variations globally.

2. SUMMARY OF THE METHOD

2.1. Analysis

The analysis of a natural texture is done locally on sub-images. The placement rule is extracted first and is characterized by a basis of two linear independant vectors. This basis fully describes the grid along which the primitive occurs. This extraction is done by detecting the maxima of the autocorrelation function of the sub-image. A preprocessing (median filtering) is done before the estimation of the autocorrelation function to make the detection of the maxima easier. Figure 1 shows on real data an original (a) and preprocessed (b) texture sub-image, the corresponding autocorrelation function (c), the detected maxima and the chosen basis (d).

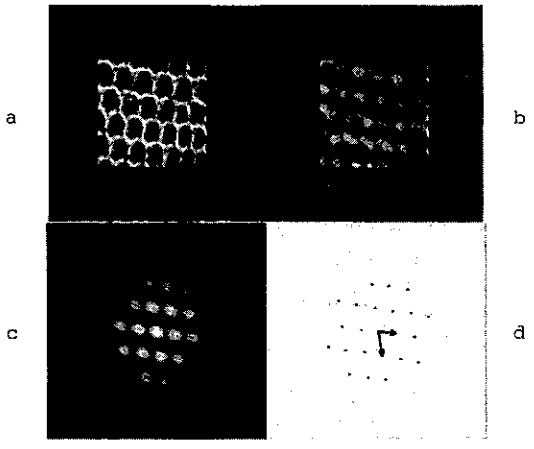


Figure 1 : original (a) and preprocessed (b) texture sub-image, autocorrelation function (c), detected maxima and chosen basis (d)

The extraction of the primitive is based on the knowledge of the placement rule. The idea is to represent the texture in another coordinate space, where the basis characterizing the placement rule is orthonormal. This space is called the normalized coordinate space (NCS) of the texture, in contrast with the original coordinate space (OCS). This operation is performed by an affine transform. The extraction of the primitive is straightforward in the normalized coordinate space. Indeed, any square of unit area of texture in this space includes one normalized primitive. Nevertheless, as it is desirable to compare the primitive extracted in one sub-image with the other ones, it is not possible to select such a

square anywhere. Section 3 presents a technique solving this problem. Figure 2 shows first an original texture sub-images in the OCS (a) and in the NCS (b) and then the repetition of the extracted primitive in the OCS (c) and in the NCS (d).

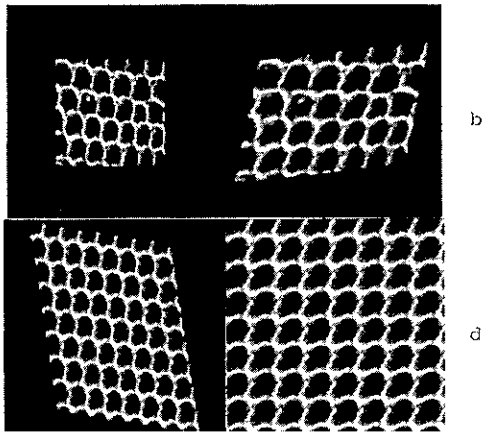


Figure 2 : original texture sub-image in the OCS (a) and in the NCS (b), repetition of the extracted primitive in the OCS (c) and in the NCS (d)

2.2. Synthesis

The synthesis is done by scanning the OCS, pixel by pixel, and assigning a grey level at each one of them. The only information available is a set of basis and normalized primitives. Conceptually it is done at each pixel location in two steps. The first one is an interpolation carried out by mean of classical methods a normalized primitive. It is assumed that the extracted primitives are assigned to the center locations of the analysis sub-images. The second step is to compute the relative position within this primitive in order to get the grey level.

In practice, only the primitive pixels needed are interpolated and the relative position is evaluated first. This is done by considering a global normalized texture which maps the entire picture plane. The transform which links the OCS and the NCS is no longer affine but more complex. Section 4 explains how to calculate it. Since the position information needed is included in this transform, it is not necessary to explicitely compute the global normalized texture.

3. STUDY OF THE PRIMITIVE

The purpose of this section is to explain how to extract one primitive in each normalized sub-image (like the one shown in fig. 2b) in order to compare them together. This comparison is necessary to analyze the variations of the primitive from one position to another. This problem is equivalent to the search a reference point in each primitive.

To solve this problem, the first idea is to intercorrelate normalized sub-images with each other, to find the maximum locations and to determine the position shift of their primitives. But the results given by this method are generally of poor quality. This is due to the fact that the assumption of regularity of the placement rule is only approximately verified. Natural textures have variations in their periodicity or even defects.

Consequently, this first idea must be modified to get a more robust method. The principle used is to compute a mean normalized primitive over the entire picture (global mean primitive) and to intercorrelate it with each normalized sub-image. This allows to extract in each one the best primitive candidate, starting for each of them at a same reference point.

The computation of this global mean primitive is first done by calculating in each sub-image a local mean primitive. This is performed by splitting the sub-image in adjacent unit area squares and summing them. Then these local mean primitives are compensated on position shifts (by intercorrelation and row and/or column cyclic permutation) and finaly summed. If the regularity assumption is violated in one sub-image, the corresponding local mean primitive tends to become a constant and does not influence noticeably the shape of the global mean.

If ever the primitive varies too much along the entire picture to allow the computation of a good global mean primitive, it is always possible to replace it by a set of regional mean primitives calculated on a fewer number of adjacent sub-images.

4. RECONSTITUTION OF THE PLACEMENT RULE

The first need of the synthesis method is the knowledge of a transform which normalizes the entire picture plane. Because of the assumption of slow variations, the transform used is locally affine. This corresponds to a locally useful first order approximation of the exact transform but this is no longer valid on the entire picture and a more complex transform has to be found. Mathematically, it can be represented by a pair of two-dimensional functions $a(k,l)$ and $b(k,l)$ such that $(a(k,l),b(k,l))$ is the projection in the normalized space of the point (k,l). The same method is used for estimating both $a(k,l)$ and $b(k,l)$, so only the computation of the estimate $\hat{a}(k,l)$ will be presented.

Let $\hat{a}(k,l)$ be a 2-D discrete function with sampling periods Δx and Δy. Its partial derivatives may be interpreted as being the horizontal component of the elementary displacements in the normalized space corresponding to infinitesimal displacements along the rows, respectively the columns, in the original one. These displacements (and thus the partial derivatives) are easily computed if at each location (k,l) the coefficients of the affine transform that locally normalize the texture are known. This is done by attributing the coefficients estimated during the analysis at the center of each sub-image and then by interpolating them at each location. Classical interpolation method may be used.

The integration of these partial derivatives will give the function $\hat{a}(k,l)$. The use of its Fourier transform A(m,n), which is sampled with periods Δf and Δg, is proposed to get the solution. It is related to the partial derivatives by :

$$\partial\hat{a}(k,l)/\partial x = j\ 2\pi\ F^{-1}\{m\ \Delta f\ \hat{A}(m,n)\} \qquad (1)$$

$$\partial\hat{a}(k,l)/\partial y = j\ 2\pi\ F^{-1}\{n\ \Delta g\ \hat{A}(m,n)\} \qquad (2)$$

In practice, equations (1) and (2) must be rewritten as :

$$\partial\hat{a}_x(k,l)/\partial x = j\ 2\pi\ F^{-1}\{m\ \Delta f\ \hat{A}_x(m,n)\} \qquad (3)$$

$$\partial\hat{a}_y(k,l)/\partial y = j\ 2\pi\ F^{-1}\{n\ \Delta g\ \hat{A}_y(m,n)\} \qquad (4)$$

where $\hat{A}_x(m,n)$ and $\hat{A}_y(m,n)$ are the estimates of A(m,n) given by each partial derivative. In most cases these estimates are not equal, which implies that $\hat{a}(k,l)$ depends on the integration path. The proposed strategy to get a result independant from this path is to modify the estimates of the partial derivatives. This may be done by adding some quantities $e_x(k,l)$ and $e_y(k,l)$ so that :

$$\hat{A}_x(m,n)+E_x(m,n) = \hat{A}_y(m,n)+E_y(m,n) = \hat{A}(m,n) \qquad (5)$$

where $\hat{A}(m,n)$ is the estimated Fourier transform of the result. It is straightforward to see that $E_x(m,n)$ and $E_y(m,n)$ have to be minimized. Choosing the mean square criteria and doing some algebra it is found that :

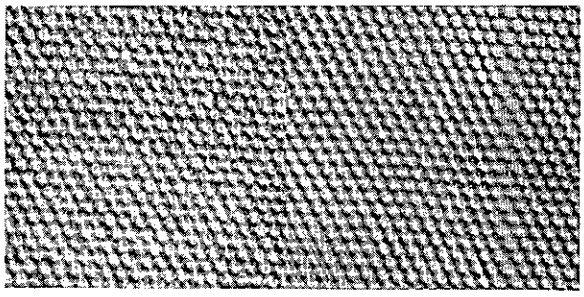
$$\hat{A}(m,n) = \frac{m^2\ \Delta f^2\ \hat{A}_x(m,n) + n^2\ \Delta g^2\ \hat{A}_y(m,n)}{m^2\ \Delta f^2 + n^2\ \Delta g^2} \qquad (6)$$

Equation (6) give the value of A(m,n) at each location except at (0,0). This means that the origin of the normalized space may be arbitrarily positionned.

Although the goal seems to be reached, a practical problem remains. It is due to the sampling of the Fourier transform, which implies the periodicity of $\hat{a}(k,l)$. Because of this, the projection of the starting point of each row (or each column) is very close to the projection of the corresponding ending point. In other words, each row or column is projected on a closed curve in the normalized space. The idea proposed to cancel this effect is to compute $\hat{a}(k,l)$ on a double sized texture where each row and each column of the original texture is repeated once but in the reverse order of the samples. In this case, the first half of a row or a column (which is the interesting part) is projected on a non-closed curve. The second half is projected on the same curve but with the starting and ending points being permuted.

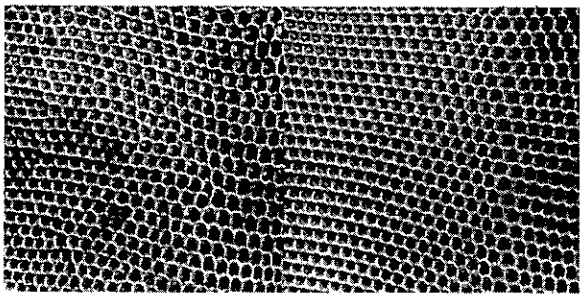
5. EXPERIMENTAL RESULTS

Figures 3 to 6 show some experimental results. All the pictures are digitized with 256x256 pixels and quantified at 256 grey levels. It can be verified that in each synthesis some variations of the primitive have been reproduced. The number of sub-images used for the analysis is a very important parameter because it sets the resolution of the analysis. The experimental values are sixteen (4x4) adjacent sub-images of size 64x64 in the case of the three first synthesis (figures 3 to 5). For figure 6 this number is increased to demonstrate its importance. The new values are thirty-six (6x6) and sixty-four (8x8) partially overlapped



a b

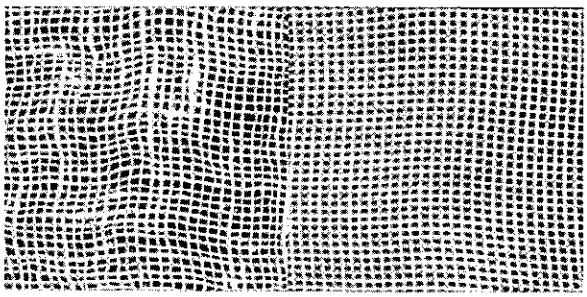
Figure 3 : original texture (a) and synthesis (b) from 16 analysis sub-images



a b

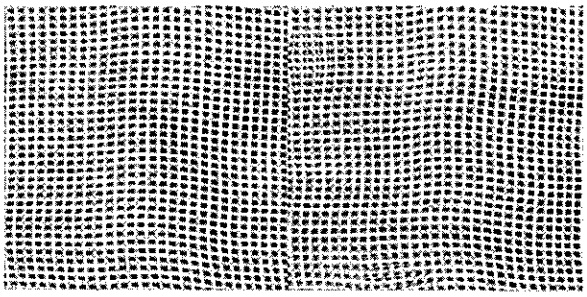
Figure 4 : original texture (a) and synthesis (b) from 16 analysis sub-images

sub-images of size 64x64 and 48x48 respectively. These two last synthesis correspond to the original texture shown in figure 5.



a b

Figure 5 : original texture (a) and synthesis (b) from 16 analysis sub-images



a b

Figure 6 : synthesis from 36 analysis sub-images (a) and 64 analysis sub-images (b)

6. CONCLUSION

In this paper, two main improvements to a new method for the synthesis of a family of structured textures have been presented after a brief summary of the method as a whole. By allowing on one hand the reproduction of some variations of the primitive and on the other hand the choice of the analysis resolution, the quality of the synthesized pictures may significantly be increased. This is confirmed by the experimental results. The quality of the synthesis depends now only on the degree of validity of the assumptions of the model. Further research will investigate a new modelisation and will allow to consider a wider class of natural structured textures.

REFERENCES

[1] P.Volet and M.Kunt, "A new method for the synthesis and efficient coding of natural structured textures", Proc. of the Picture Coding Conference in the 2nd International Symposium on Optical and Electro-Optical Science and Engineering, Dec. 2-6, 1985, Cannes, France.

[2] R.M. Haralick, "Statistical and structural approaches to texture", Proceedings IEEE, vol.67, pp. 786-804, may 1979.

[3] S.W. Zucker, "Toward a model of texture", Computer Graphics and Image Processing, vol.5, pp. 190-202, 1976

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

THE CONSTRAINED DISTANCE TRANSFORMATION: A PSEUDO-EUCLIDEAN, RECURSIVE IMPLEMENTATION OF THE LEE-ALGORITHM

Leo Dorst and Piet W. Verbeek

Delft University of Technology, Dept. of Applied Physics
Lorentzweg 1, 2628 CJ Delft, The Netherlands

An algorithm is introduced that performs tasks similar to those of the Lee-algorithm, but in a way that is faster, more accurate, and more easily implementable in hardware. It can, among other things, be used to find the shortest route through a binary landscape.

1. INTRODUCTION: DISTANCE TRANSFORMATIONS AND THE LEE-ALGORITHM

The distance transform has become a practical tool after Borgefors [1][2] gave a quick and accurate pseudo-euclidean distance transform algorithm. Normally, it is used to measure the size of objects, but one could also use it to measure the distance of an arbitrary image point to a given goal point (Fig. 1). By making a distance transform of the whole image (except the goal point), at each point the distance to the goal point is found. We can then find the shortest path to the goal point P, starting from a point A, by simply following the distance gradient downward to its source. Seen in this way, the distance transform is capable of transforming a global optimization problem (find the shortest path to the goal) to a local problem (follow the gradient).

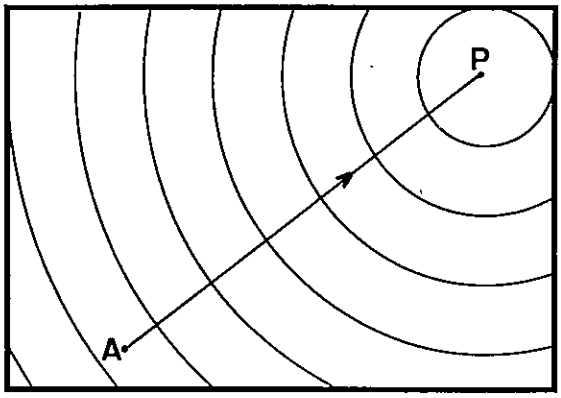


Figure 1

The constrained distance transformation presented here is an extension of the normal distance transformation.
Consider a situation as in figure 2, where we have several impenetrable objects, and would like to move from any point to the goal point P, along the shortest path. This problem could be solved easily if there were a means of propagating 'distance waves' from P, taking into account the impenetrability of the objects. If that is done, space is filled with distances, and by following the steepest

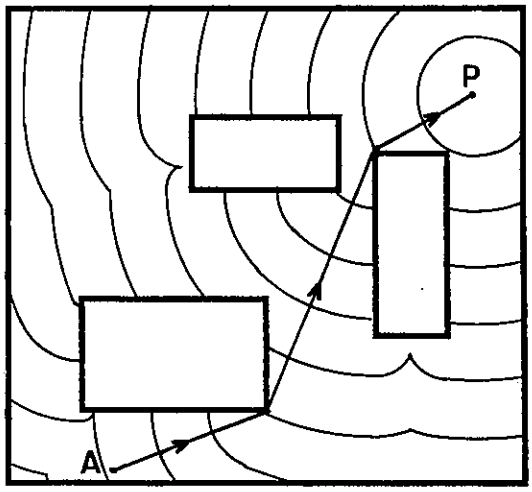


Figure 2

gradient all the way down, starting from a given point A, one eventually ends up at P. Note that at every point it can be determined locally what step should be taken. An algorithm producing this distance image is the famous Lee algorithm [3]. However, this algorithm has the disadvantage that it is rather slow, that it produces distances according to the cityblock or chessboard distance, and that it is difficult to implement in hardware.
In this paper, we will present an alternative algorithm, called the constrained distance transform (CDT), which has none of these disadvantages. It is an extension of the Borgefors distance transformation. The algorithm works in arbitrary dimensions; here, we will treat the 2-dimensional case.

2. THE BORGEFORS ALGORITHM

Recently, a new algorithm by Borgefors [1] and an improvement thereof [2] have made distance transforms a practical tool, for the following reasons:
1) It produces the distance transform by means of a 2-pass, recursive filteroperation, with a filter of size 3*2 or 5*3 (depending

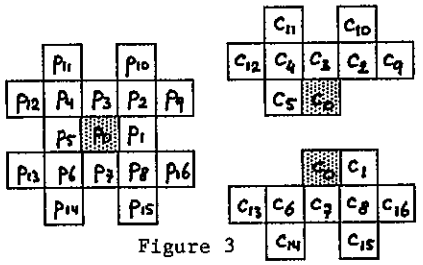


Figure 3

on the accuracy needed), using very simple integer operations.

2) The Borgefors Distance Transformation (BDT) has 3 coefficients which allow very close approximation of the Euclidean distances, both in value and in isotropy.

The Borgefors distance transform algorithm works in two passes. In the first pass the recursive filter of figure 3a is placed at each object point p_0. The points in this window are those with labels from $F_1 = \{0,2,3,4,5,9,10,11,12\}$ (figure 3b). The new value for p_0 is computed as:

$$p_0 := \min_{i \in F_1} \{dist(p_i)\} \qquad (1)$$

with

$$dist(p_i) = p_i + c_i \qquad (2)$$

Here p_i indicates a pixel from the result image (this is the recursiveness), and c_i are the filter coefficients of figure 3a:

$$\begin{aligned} c_0 &= 0 \\ c_3 = c_5 &= d_1 \\ c_2 = c_4 &= d_2 \\ c_9 = c_{10} = c_{11} = c_{12} &= d_3 \end{aligned} \qquad (3)$$

Initially, the background points have a value 0, and object points have a value ∞.
In the second pass the filter of figure 3c is used, where the points are those with labels from $F_2 = \{0,1,6,7,8,13,14,15,16\}$. Now, the new p_0 is:

$$p_0 := \min_{i \in F_2} \{dist(p_i)\} \qquad (4)$$

with

$$\begin{aligned} c_0 &= 0 \\ c_1 = c_7 &= d_1 \\ c_8 = c_6 &= d_2 \\ c_{13} = c_{14} = c_{15} = c_{16} &= d_3 \end{aligned} \qquad (5)$$

The result is an image where each point contains the minimum 'chamfer' distance to a background point, where a grid step is measured as d_1, a diagonal step as d_2 and a knight's move as d_3. Let us indicate this Borgefors distance transform with BDT(d_1,d_2,d_3). It follows that a city-block distance transformation is BDT(1,2,3), or BDT(1,∞,∞), a chess-board distance is BDT(1,1,∞), and a locally correct Euclidean distance BDT(1,$\sqrt{2}$,$\sqrt{5}$) , or, as a good rational approximation of the ratio $1:\sqrt{2}:\sqrt{5}$, such as d_1=5, d_2=7, d_3=11. 'Circles' (points with equal distance to a given point) corresponding to the coefficient values are given in fig.4. One may view the working of the Borgefors algorithm as that of a recursive distance propagator. The region of influence of a point on the result in one pass of the algorithm is as sketched in figure 5.

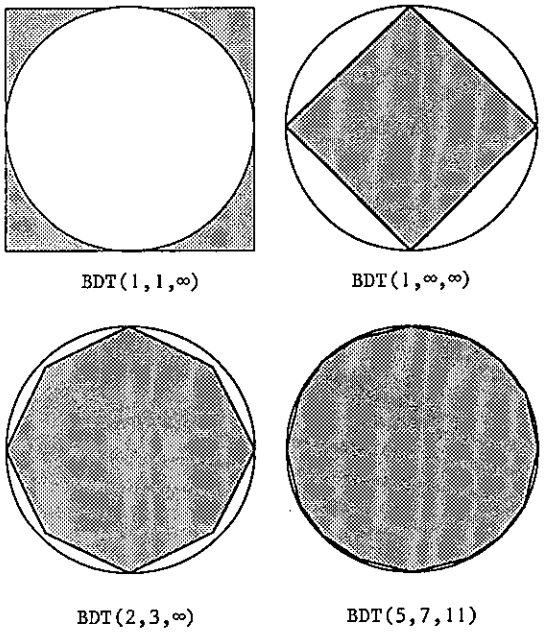


Figure 4

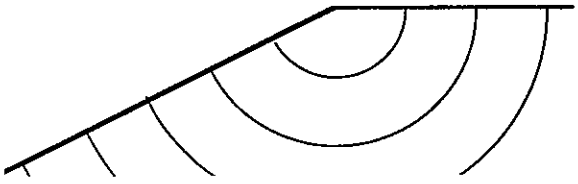


Figure 5

3. THE CONSTRAINED DISTANCE TRANSFORMATION ALGORITHM

In the Constrained Distance Transformation (CDT) in its most general form, two input images M and B are used. Image M is a binary mask, and indicates objects and background pixels, just as in the Borgefors algorithm. B is a grey value image, and contains the boundary conditions: pixels of which the distance is already known. The images are now scanned by the same recursive filters as in the Borgefors algorithm, but taking into

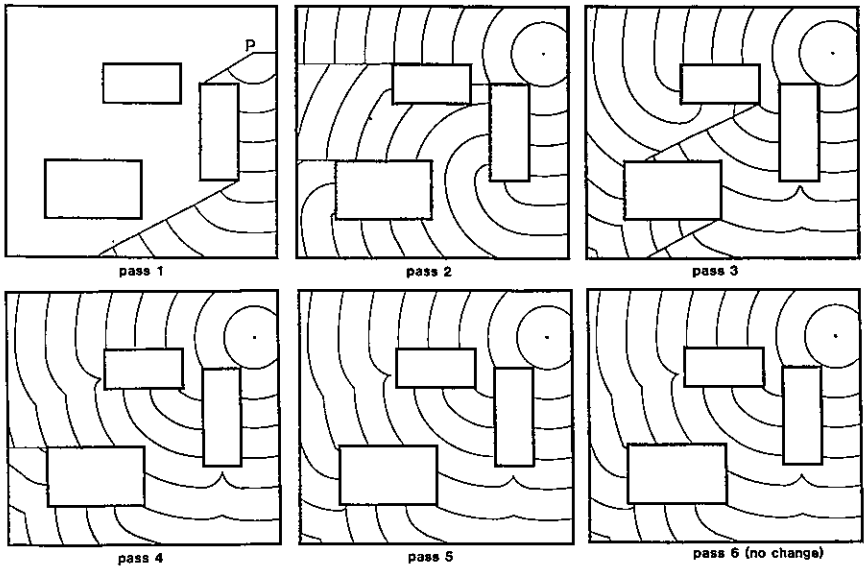


Figure 6

account the values in B in the following manner.
One still has, at a position p_0, that the new value for p_0 is computed as:

$$p_0 := \min\{dist(p_i)\}$$

but now the function $dist(p_i)$ is redefined as

$$dist(p_i) = \begin{cases} p_i + c_i & \text{if } p_{i,B} = 0 \\ p_{i,B} + c_i & \text{if } p_{i,B} \neq 0 \end{cases} \qquad (6)$$

where p_i indicates a pixel from the result image (this is the recursiveness), and $p_{i,B}$ the corresponding pixel from the image B (this incorporates the constraints).
The image is scanned till there is a scan in which no change occurs. The number of scans is dependent on the number of secondary wave sources in the image (see figure 6).
From the formula one can see that $p_{i,B}$ will be a source of distance waves if it is smaller than p_i, and will absorb waves if it is larger than p_i.
In this algorithm, one can make impenetrable boundaries by pixels with a value of ∞ in image B: never will such a pixel function act as a source for new distance waves. In that case, the B image can be taken binary, and (6) redefined to

$$dist(p_i) = \begin{cases} p_i + c_i & \text{if } p_{i,B} = 0 \\ \infty & \text{if } p_{i,B} = 1 \end{cases} \qquad (7)$$

(If one uses the 5*3 window, these impenetrable arcs should be at least 2 pixels thick, otherwise 'leaking' will occur!).
Throughout the image, the speed of propagation of the distance waves is the same.

4. THE SHORTEST PATH ALGORITHM AND VARIATIONS

4.1 The Basic Algorithm

The constrained distance transformation was developed to solve the old optimization problem of finding the shortest path to a goal point avoiding obstacles, using the basic idea behind the Lee algorithm. The way to proceed is indicated schematically in figure 6.
The image B with the boundary conditions contains the obstacles, with a distance value of ∞. Image M, with the mask indicating where the distances should be computed, is everywhere 'set' (value ∞) except at the goal point, which is set to 0. The CDT algorithm described then produces a distance landscape caused by waves bending around the obstacles. The shortest path from a given point to the goal point is found by following the steepest gradient downward.
For the gradient in a distance image we can be somewhat more precise: what we are looking for is the distance source. That is, for a point p_0 we should see if there is a point p_i in the neighborhood of figure 3, which has exactly the right difference:

$$p_0 - p_i = \begin{cases} d_1 & \text{if } i=1,3,5,7 \\ d_2 & \text{if } i=2,4,6,8 \\ d_3 & \text{if } i=9,10,12,12,13,14,15,16 \end{cases} \qquad (8)$$

If no such point can be found we have reached the source of the distance waves, which is the goal point.
There are points in the image where two or more paths tie. These points are local maxima, saddle points, or promontories in the distance

landscape, and they denote the position of medical axis points. They can be found from the distance landscape by a fast pseudo-euclidean skeletonizing algorithm [4].

4.2 A Modification: Fat Robots

In making the distance transform of the obstacles we have not taken into account that some of the paths between them could be too narrow to move through.
Supposing the object one wants to move is a circle (or, in D dimensions, a D-sphere) of radius r, this can easily be mended by first dilating the B-image by this circle. (Note that dilation can be done circularly by application of the Borgefors distance transform, and thresholding). This closes off narrow passages, and if we now perform the basic algorithm, paths are guaranteed to run at a distance of at least r away from the obstacles (figure 7).

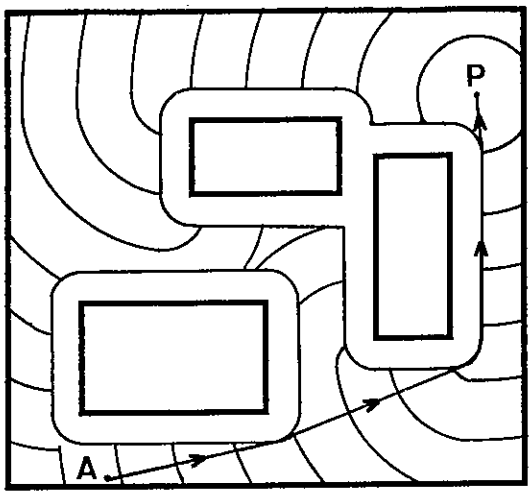


Figure 7

4.3 Comparison with the Lee algorithm

The above applications can also be solved by the classical algorithm of Lee [3]. However, there are reasons why the distance transform approach is to be preferred.
In the Lee-algorithm, a 'cost' transform is made of the image; these 'costs' can be quite general functions. A comparison with the CDT-algorithm is possible if the 'cost' is 'distance to the source'. In the Lee-algorithm, the goal point radiates distance waves, and these are expanded by adding layer after layer of the distance wavefronts. This requires an expensive administration of the points to be treated, or as many scans of the image as the value of the maximum distance. The CDT-algorithm requires a number of passes corresponding to the number of object layers seen from the goal point (usually small, see fig.6).
The fact that we only need a simple recursive filter makes the algorithm very well suited for hardware implementation. Already in software, it is fast (in C, on a MC68000 8MHz system, data fetched over VME bus, for a 256*256 image, 3 seconds for each image pass). Another disadvantage of the Lee-algorithm is that the only distances that can be propagated are the city-block or chessboard distance. As was seen (fig. 4), these differ substantially from the Euclidean measures, especially in their directional behaviour. The CDT-algorithm, propagating the hexadecagonal Borgefors distances, approximates the Euclidean case much more closely.

5. CONCLUSION

An algorithm was presented that can generate a distance landscape taking into account boundary conditions.
Among other things, the algorithm can be used as a fast, more accurate, and hardware-implementable means to obtain the 'cost'-landscapes the Lee algorithm requires. It might make Lee-algorithm-like procedures feasable for many applications where they could not be used before due to lack of speed or Euclidean behaviour.
The algorithm is a modification of the well-known Borgefors distance transform algorithm. It uses 5*3 recursive filter, with simple integer operations, and requires few passes through the image.
Five applications will be illustrated on the actual poster:
- optimal path between obstacles
- perimeter measurement
- convex hull determination
- reconstruction from medial axis
- curvature filtering

Further research is being performed, aiming at the use of the CDT-algorithm in robot state space to solve the collision avoidance problem. This will be reported elsewhere.

REFERENCES

[1] G. Borgefors, Distance Transformations in Arbitrary Dimensions, Computer Vision, Graphics and Image Processing, 27, 1984, pp. 321-345.
[2] G. Borgefors, Distance Transformations in Digital Images, accepted for CVGIP, 1986.
[3] C.Y. Lee, An Algorithm for Path Connections and Its Applications, IRE Transactions on Electronic Computers, September, 1961, pp. 346-365.
[4] L. Dorst, G. van Antwerpen, Pseudo-Euclidean Skeletons, accepted for CVGIP, 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IDENTIFICATION OF 2D OBJECTS IN 3D SPACE

Andrzej Śluzek

Warsaw University of Technology
Institute of Automatic Control
ul. Nowowiejska 15/19
00-665 Warszawa, Poland

In many applications of computer vision devices, the planar objects must not be considered in 2D space. Such a situation arises if the axis of a camera is non-orthogonal to the plane of observed objects. Then, because of the perspective distortion, the shapes received from the camera do not correspond to the original shapes of objects. The paper presents two algorithms which are not sensitive to these deformations. The first algorithm is the simplified version of the second one. The algorithms can be applied not only to the identification but also to the localization of 2D objects.

1. INTRODUCTION

The real-time applications of computer vision devices need reliable algorithms of identification of objects. It is also obvious that these algorithms should have certain computational properties (i.e., small complexity, possible hard-ware implementation etc.). Moreover, to successfully apply such algorithms, the environment of a vision system should fulfil several conditions. The satisfactory reliability of identification can be achieved only for single or non-overlapped objects.

If object are two-dimensional or can be regarded as such, the application of moments of the intensity function yields the best results. Moment-based algorithms are invariant under rotation, translation and change of scale of objects (Hu /1/). They also fulfil the above-mentioned computational requirements.

The paper deals with the problem of adaptation of moment-based algorithms to the identification (and localization) of 2D objects observed by a tilted camera. In this situation, the received shape of an object is determined not only by 2D transformations (i.e., translation and rotation), but also by spatial relations (i.e., inclination of a camera). As a result, the camera "sees" the deformed object (Fig.1).

If objects are in a good distance off a camera, the deformations can be approximately described by linear transformations. Otherwise, the non-linear perspective transformations should be applied. The two presented algorithms are not sensitive to these deformations. The first algorithm deals with the simplified problem (linear transformation) and the second one is based on the perspective transformation.

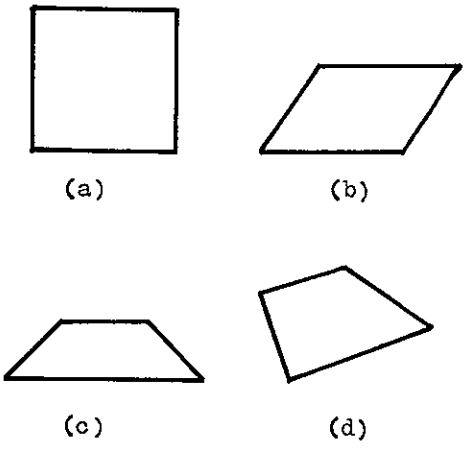


Fig.1 Original shape (a) and deformed shapes (b,c,d) of a square

2. MATHEMATICAL FOUNDATIONS

2.1. Moments of intensity functions

Let $f(x,y)$ be an intensity function which represents the observed object. A moment of order $p+q$ of the function $f(x,y)$ is defined by the following expression:

$$m_{pq} = \iint x^p y^q f(x,y)\,dx\,dy. \qquad (2.1)$$

A related set of central moments is defined by

$$c_{pq} = \iint (x-x')^p (y-y')^q f(x,y)\,dxdy, \qquad (2.2)$$

where

$$x' = m_{10}/m_{00} \qquad y' = m_{01}/m_{00}.$$

Several useful equations for moments of order 1 and 2 should be mentioned:

$$c_{11} = m_{11} - m_{10}m_{01}/m_{00} \qquad (2.3)$$

$$c_{20} = m_{20} - m_{10}^2/m_{00} \qquad (2.4)$$

$$c_{02} = m_{02} - m_{01}^2/m_{00}. \qquad (2.5)$$

let the domain of the function f(x,y) (i.e., the shape of the observed object) be transformed by a differentiable mapping A

$$u = a_1(x,y) \qquad (2.6)$$

$$v = a_2(x,y).$$

Then the moments will be transformed in accordance with the following expression (n denotes transformed moments and d - transformed central moments):

$$n_{pq} = \iint u^p v^q f(x,y) J\,dxdy, \qquad (2.7)$$

where

J - Jacobian of the mapping A.

Similarly,

$$m_{pq} = \iint x^p y^q f(u,v) K\,dudv, \qquad (2.8)$$

where

K - Jacobian of the inverse mapping.

2.2. Transformations of objects

The observed shape and position of a 2D object are determined by two transformations B and C. B-transformation changes the planar position of the object in accordance with Eq.(2.9) (Fig.2).

$$u = K(x\cos Z - y\sin Z) + G \qquad (2.9)$$

$$v = K(x\sin Z + y\cos Z) + H$$

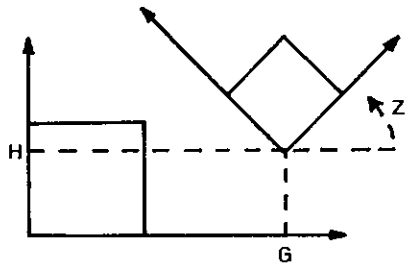


Figure 2

C-transformation results from the inclination and rotation of a camera (Fig.3).

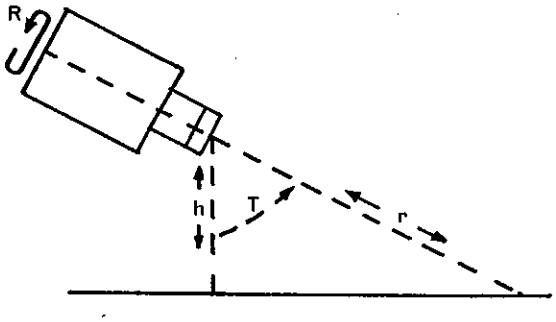


Figure 3

If the object is in a good distance off a camera and/or the angle of inclination T is small, C-transformation can be regarded as the linear one and is described by the following formula (rotation of a camera is ignored):

$$u = x \qquad (2.10)$$

$$v = y\cos T.$$

If the perspective distortion cannot be ignored, C-transformation is defined by

$$u = Px/(P + y) \qquad (2.11)$$

$$v = Py/(P + y),$$

where

$$P = h/\sin T \text{ or } P = hr/\sqrt{r^2 - h^2}.$$

2.3. Moment invariants

A cerain set of expressions, which are invariant under B-transformation, have been derived (/1/, /2/) from the moments of order 2 and 3. The simpliest two are

$$\frac{c_{20} + c_{02}}{m_{00}^2} \qquad (2.12)$$

$$\frac{(c_{20} - c_{02})^2 + 4c_{11}}{m_{00}^4} \cdot \qquad (2.13)$$

From the formulae 2.12 and 2.13 the following expression can be obtained:

$$\frac{c_{20}c_{02} - c_{11}^2}{m_{00}^4} , \qquad (2.14)$$

which appears to be invariant under every linear transformation.

3. DESCRIPTION OF ALGORITHMS

3.1. General remarks

The received shape of an object is determined by the superposition of B and C-transformations. The parameters of B-transformation (Eq.(2.9)) describe the localization of the object and they are unknown, while in the majority of applications of vision systems (e.g. airplane cameras, cameras mounted on robot arms, etc.) the parameters of C-transformation (Eqs.(2.10), (2.11)) can be regarded as known. Thus, the algorithm of recognition of 2D objects consists in two steps:

(i) identification of the object (we assume, if needed, that C-transformation is known),
(ii) localization of the object.

3.2. Identification of objects

The identification of objects can be simplified by applying Eq.(2.10); otherwise, Eq.(2.11) should be used.

In the first case the identification is possible without any information about C-transformation. Since the superposition BC is the linear transformation, the formula (2.14) can be applied to verifying whether the observed object fits the template object.

If the perspective distortion is not ignored, the identification of objects is slightly more complex. It is easy to show that the initial (before applying C-transformation) moments of the object are expressed as follows:

$$n_{00} = P^3 \iint \frac{f(u,v)}{(P-v)^3}\, dudv \qquad (3.1)$$

$$n_{10} = P^4 \iint \frac{uf(u,v)}{(P-v)^4}\, dudv \qquad (3.2)$$

$$n_{02} = P^5 \iint \frac{v^2 f(u,v)}{(P-v)^5}\, dudv \text{ , etc.} \qquad (3.3)$$

The object can be identified by applying these equations, then Eqs.(2.3)-(2.5) and, finally, the invariant (2.12) or (2.13).

3.3. Localization of objects

Localization of objects is performed by finding parameters of B-transformation. To accomplish this task, the template moments of the object (m_{pq}, c_{pq}) are needed as well as the n_{pq}, d_{pq} moments obtained following the application of B-transformation.

If the second algorithm of the identification is applied, the moments n_{pq} will be known (Eqs.(3.1)-(3.3)). Otherwise, they can be computed from the formulae

$$n_{00} = n'_{00}/\cos T \qquad (3.4)$$

$$n_{10} = n'_{10}/\cos T$$

$$n_{01} = n'_{01}/\cos^2 T$$

$$d_{11} = d'_{11}/\cos^2 T \text{ , etc}$$

(n'_{00}, d'_{11}, etc denote the moments after the application of C).

The angle T (Fig.3) is specified or it can be easily obtained from the equation

$$\frac{d_{20} + d_{02}}{n_{00}^2} = \frac{d'_{20}\cos T + d'_{02}/\cos T}{(n'_{00})^2} \cdot \qquad (3.5)$$

The left side of Eq.(3.5) is known (invariant (2.12)).

Then the parameters of B-transformation can be computed from the set of simple formulae:

$$n_{00} = K^2 m_{00} \qquad (3.6)$$

$$\cos^2 Z = w(K, c_{20}, c_{02}, c_{11}, d_{20}, d_{11}) \qquad (3.7)$$

(w is a polynomial function)

$$G = n_{10}/n_{00} - K(x'\cos Z - y'\sin Z) \qquad (3.8)$$

$$H = n_{01}/n_{00} - K(x'\sin Z + y'\cos Z). \quad (3.9)$$

4. CONCLUSIONS

The presented algorithms have been tested for binary images about the size of 50x50 pixels. The experiments have shown that the simplified version of the identification can be more widely applied than it was expected. The perspective distortion must not be ignored only if the diameter of objects is maximum 4 - 5 times less than their distance off a camera and the angle T exceeds forty degrees.

The algorithms can process binary images and they require no sophisticated methods of enhacement of images (shapes of objects are not very sensitive to noises caused by external factors). The only necessary condition is that the background of objects should be regarded as equal to zero.

The computational structure of the algorithms is very regular (the implementation does not depend on a type of recognized 2D objects). Thus it is possible to apply array processors or to design a specialized processor which would execute the algorithms.

NOTES

If the rotation of a camera (the angle R in Fig.3) is considered, the simplified version of the identification will not be changed (the superposition of transformations is still linear). However, the formulae (3.1)-(3.3) should be modified, but this would not influence the idea of the algorithm. The method of the localization is also almost unchanged.

REFERENCES

/1/ Hu,M.K., Pattern Recognition by Moment Invariants. IRE Trans.Inf. Theory vol.IT-8 (1962) pp.179-187.
/2/ Gonzales,R.C. and Wintz,P., Digital Image Processing (Addison-Wesley, 1977).
/3/ Bolc,L. and Kulpa,Z.(eds), Digital Image Processing Systems (Springer Verlag, 1981).
/4/ Śluzek,A., Identification of Planar Objects in Visual Field of Robots (in Polish), in: Proc.of 1st Nat. Conf.on Robotics (Wrocław,1985) pp. 199-205.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW EDGE-DETECTION SCHEME BASED ON LOCAL CORRELATION FUNCTION

Giovanni GARIBOTTO

ELETTRONICA SAN GIORGIO - ELSAG S.p.A.
Via Puccini, n. 2
16154 GENOVA - ITALY

A new edge-detection technique is proposed, to overcome the usual problems of gradient-based methods, such as noise sensitivity and the strong dependence on the local contrast.
The basic idea is to use an adaptive estimate of the local correlation activity, in order to normalize the edge map and select significant oriented structures from uncorrelated noise patterns.
A suitable threshold on the local variance prevents the emphasis of flat uniform regions, which otherwise would exhibit large correlation values.
The example referred in the paper show improved performance over conventional operators, in the detection of low-contrast details.

1. INTRODUCTION

Edge detection represents a fundamental operation in most image processing tasks, to produce a simplified sketch of the scene. Such preprocessing is required to achieve the enhancement of oriented structures (such as lines, curves, contours) to be better perceived by the human eye (as in medical application) or to be used by a machine vision system (industrial robotics).
Robustness with respect to noise is particularly important in practical applications where the detection of low-contrast patterns is often affected by errors and uncertainty. [1]
Most commonly used techniques for edge-detection are gradient-like operators, essentially based on local derivative estimates, and their performance is heavily dependent on the amount of noise in the picture. Band-pass filtering is often used (as in the zero-crossing implementation) to minimize noise effects.
Anyway, poor performance is found at low contrast levels where some edges can be lost, since they are usually masked by the strong intensity of sharp discontinuities within the scene.
Henceforth, adaptive techniques are particularly attractive to achieve a normalized detection of contours and details, as well as to minimize the influence of noise.
The proposed approach in the paper is based on local correlation-like estimates as a measure of edge and structure activity which is insensitive to image contrast.
At first the non stationary statistical model of the image is briefly discussed, to introduce the adaptive correlation function and its estimation procedure.
A good deal of approximation is then carried out on this model to account for implementation and application constraints.
Finally, the proposed edge detection scheme is described in more details. It is shown that the full process can be reduced to selecting the most appropriate convolution on a matrix of local products, as a compromise between resolution and noise reduction.

2. CORRELATION MAP

The solution proposed here is based on the computation of local correlations in order to detect the presence of oriented structures with respect to uncorrelated noise. As such, it should be better defined as a pattern detection scheme, being well suited for the estimation of edges, lines, contours, etc.
As a matter of fact, a common property of such patterns is a large amount of local correlation along with the edge direction, with respect to unstructured noisy background.
The ambiguity determined by flat areas, which would have a maximum correlation factor, can easily be removed by a suitable threshold on the local variance estimate. As a consequence, the obtained measure bears just a poor relationship

to the ideal correlation function, and the corresponding edge pattern has to be evaluated through severe approximation steps.
The underlying model of the picture, in this adaptive edge detection scheme, is a non-stationary process, and the space-variant autocovariance function turns out to be

$$C_{i,j}(k,\ell) = \sigma^2(i,j)\,\rho(i,j)^{\sqrt{k^2+\ell^2}} \qquad (1)$$

for a non separable first-order Markov random field.
In the assumption that the process is wide-sense stationary within a moving window S of MxM pixels, a local mean $\bar{f}_{ij}$ is computed for each image sample f(i,j) to obtain:

$$\bar{f}_{i,j} = \sum_{(m,n)\in S} c(m,n)\, f(i-m,j-n) \qquad (2)$$

using low-pass filtering weights c(k,l).
The local difference component is then computed as:

$$d(i,j) = f(i,j) - \bar{f}_{i,j} \qquad (3)$$

The correlation factor $\rho(i,j)$ in (1) can be used as an activity measure of the presence of locally oriented patterns. This function is evaluated by local analysis within a suitable neighbourhood S_1 of NxN samples. The local variance is obtained as:

$$\sigma^2(i,j) = \frac{1}{N^2} \sum_{(m,n)\in S_1} d^2(m,n) \qquad (4)$$

The autocovariance function $C_{i,j}(k,l)$ is obtained as the usual biased estimate [2] :

$$C_{i,j}(k,\ell) = \frac{1}{N^2} \sum_{(m,n)\in S_1} d(m,n)\,d(m+k,n+\ell) \qquad (5)$$

and the correlation coefficient $\rho(i,j)$ is approximated by:

$$\rho(i,j) = 0{,}5\,\frac{|C_{i,j}(1,0)| + |C_{i,j}(0,1)|}{\sigma^2(i,j)} \qquad (6)$$

using lower order lags and absolute values to compensate for negative terms in the local autocovariance function. Moreover, to avoid high correlation values within almost uniform areas a thresholded estimate is used for the local variance

$$\sigma_T^2(i,j) = \begin{cases} \sigma^2(i,j) & \text{if } \sigma^2(i,j) > T \\ T & \text{otherwise} \end{cases} \qquad (7)$$

By substituting $\sigma_T(i,j)$ in (6) we obtain a coefficient $\rho_T(i,j)$ which represents a measure of the edge map of the image. Due to the normalization of the variance $\sigma_T(i,j)$, it is possible to detect significant oriented structures all over the image, irrespective of the non uniform contrast effects.

Fig. 1 refers an example of edge detection for a digitized X-Ray image, by comparing the results against a classical gradient operator.
A selection criterion for the processing parameters is briefly discussed in the following.
In principle the window size M for the estimation of the local mean should be different from size N (for local correlation estimate). As a general rule, using small values M (large filter bandwidth) we obtain an increasing enhancement of low-contrast details.
A different reasoning holds in the evaluation of $\sigma(i,j)$ and $\rho(i,j)$. Being statistical measures, they would require a fairly large support N, at the price of a reduced resolving power.
In the example of fig. 1c) the same window size is used M=N=5, to minimize the computational problems. This represents a reasonable compromise between the two opposite constraints of noise smoothing and resolution, even if such truncated estimates are poorly related to the previously discussed statistical parameters.

3. COMPUTER IMPLEMENTATION

A direct implementation of this edge detection method is computationally expensive since most intermediate products in the evaluation of (5) are often repeated for neighbouring pixels.
To minimize this redundancy, as well as to improve flexibility in the estimation, a slightly different approach is proposed, by storing an intermediate product function

$$P(i,j) = 0.5\,[d(i,j)\,d(i,j+1) + d(i,j)\,d(i+i,j)] \qquad (8)$$

The estimated edge function is further approximated by

$$\hat{\rho}_T(i,j) = \frac{\sum_{(m,n)\in S_1} W_{m,n} |P(i-m,\ j-n)|}{\sigma_T^2\ (i,j)} \quad (9)$$

where a suitable low-pass convolution mask $\{W_{k,\ell}\}$ allows to achieve a selective weighting of local products (8), to increase resolution in the edge estimate. To improve the system sensitivity to arbitrarily oriented patterns it is possible to introduce diagonal products in (8) to obtain:

$$P(i,j) = 0.25\ \{d(i,j)[d(i,j+1)+d(i+i,j)+ \quad (10)$$
$$+\ d(i+i,j+i)] + d(i,j+i)+d(i+i,j)\}$$

Moreover, the variance component $\sigma_T(i,j)$ in (9) is supposed to be a slowly varying function and it is mainly used for normalization purposes. As such, it is sufficient to compute a rough estimate within non overlapped windows, through the image, followed by bilinear interpolation.
Further examples of applications on different images will be presented at the Conference, to demonstrate the efficiency of the proposed method.

4. CONCLUSION

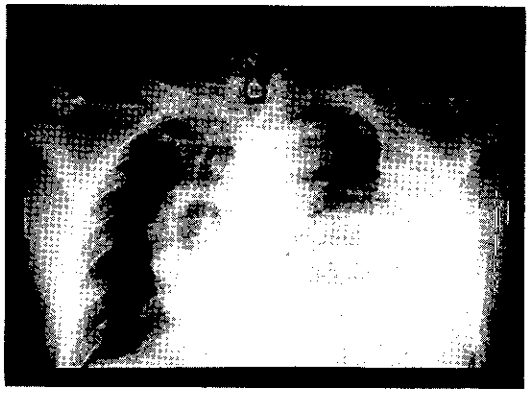
A new edge detection scheme is proposed, based on the assumption that oriented patterns to be detected should exhibit relevant local correlation, against uncorrelated non-edge noisy pixels. Moreover, this kind of measure is quite insensitive to contrast variations and allows to obtain adaptive edge detection. In fact it is possible to enhance low-contrast details with respect to sharp discontinuities, provided sufficiently structured pattern be present. The actual implementation of this method is quite far from the underlying statistical model, not only for practical reasons. At first the high correlation values which correspond to flat uniform regions have to be discarded, since they do not contribute to the edge map.
Moreover, the non-stationary model would require to perform statistical estimates within fairly large windows, with critical loss in resolution. Henceforth, the proposed approximate solution represents a compromise between flexibility and efficiency in the implementation. The edge map is obtained by computing local products of the difference component of the image, with further simplified evaluation of the local variance function. The obtained results are quite satisfactory to achieve adaptive edge detection in general conditions of variable contrast.

REFERENCES

[1] D.T.Kuan, A.A.Sawchuck, T.C.Strand, P.Chavel, "Adaptive noise smoothing Filters for Images with Signal dependent noise", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-7, n. 2 pp.165-177, Mar. 1985.

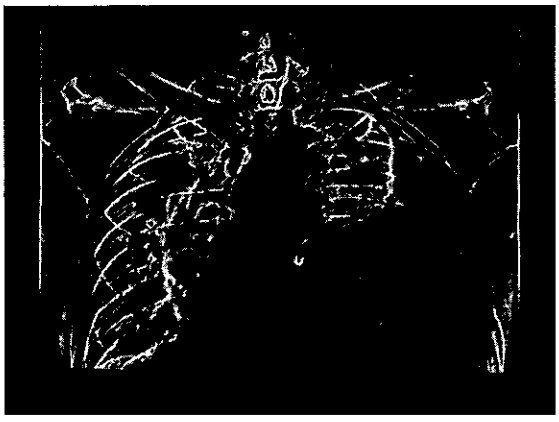
[2] A.V.Oppenheim, R.W.Schafer, "Digital Signal Processing", Englewood Cliffs, N.J., Prentice-Hall, 1975.

[3] G.Garibotto, "Digital X-Ray Enhancement through local difference equalization", Proceedings of SPIE Conference, Cannes, Dec. 1985.

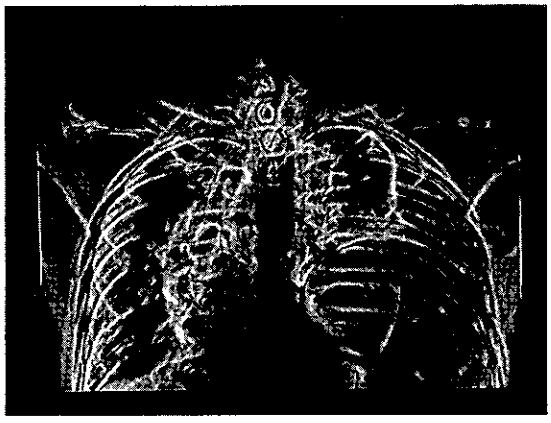


a

Fig.1. Example of edge detection.
a) Original digitized X-ray image.
b) Result of Sobel operator using a convolution mask of size (3x3).
c) Proposed pseudo-correlation map. using an estimation window (5x5)



b



c

Fig.1. cont.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

RECOGNITION OF PARTIALLY OVERLAPPED WORKPIECES BY CONTOUR SHAPE MATCHING

M.T.Pareschi, C.Raspollini

IBM Pisa Scientific Center, Via S.Maria 67, 56100 Pisa, Italy

An experimental 2-D vision system which can locate and identify partially occluded objects in gray-level noisy images is presented. The technique approximates the external object contours by straight line segments and circular arcs. This representation is used for locating and recognizing the visible parts of the workpieces.

1. INTRODUCTION

The problem of recognizing partially occluded objects is of considerable interest in industrial automation. Although it is possible to use shakers to separate overlapped objects located on a conveyor belt, a visual system able to recognize occluded parts is much more flexible.
In a typical industrial scene, the objects may be intermixed, partially occluded, and of unknown pose; however, the type of parts that will be present in each scene belongs to a known set. In other words, a description of the object to be recognized, constructed during a "learning" phase, is available.
Attempts to match the boundary or other significative features of an object with those of prestored models have been suggested in the literature.
Perkins [1] tries to approximate object contours by circular arcs and straight line segments (contiguous successions of these elements are called "concurves"). The recognition is performed by superimposing a prestored model to the object in a certain point, called center (the center is a meaningful object feature as, for example, a hole). The method then tries to align object and model concurves by rotating the model and the object with respect to each other. If a good alignment is reached, the object is recognized. Obviously, the recognition capability of this procedure strictly depends on the visibility of the center: it must exist and must not be hidden by an overlap.

In a recent work, Turney et al. [2] present a method of template matching which is based on matching subtemplates, each of them characterized by a parameter called "saliency". The saliency measures the extent to which the boundary segment distinguishes the object to which it belongs from other objects which might be present.

Bolles and Cain [3] present a method of locating partially visible two-dimensional objects, called the "Local-Feature-Focus-Method". This technique seems to be fast and reliable but the objects which can be recognized are only those having several local features, such as sharp corners and holes. Objects characterized by large continuous arcs cannot be recognized.
This paper presents a technique to recognize 2-D objects in digital gray images. The technique uses the contour profile of the objects allowing the recognition of partially occluded parts if a sufficient portion of the contour is visible in the scene.
The main steps of the technique are:
- edge finding by means of the Hueckel operator;
- contour gap filling and connected region detection;
- "contour" function computing and polygonal approximation;
- object recognition by the comparison between the object and the model polygonal approximation.

Preliminary versions of some of the concepts used in this approach have been reported in [4,5]. In the present work the recognition technique has been improved by using, to characterize the objects, additional parameters which take into account the spatial positions of the contour elements.

2. THE HUECKEL OPERATOR

The Hueckel operator [6,7,8,9] is one of the most widely-used operators described in the literature and it is actually both an edge and a line finder operator. In the present work, it has been used only as an edge operator.
The Hueckel operator gives two important information: the former is the position of the edges and the latter the angle of the tangent to the contour. These parameters will be used to create the contour function.

The Hueckel operator works very well because it is quite insensitive to random noise, but it needs high computational time (about 1 minute for images of 256×256 pixel on an IBM 4341-2 computer). The required computational time has been reduced by a pre-operator which detects the significant edge points; in this way the Hueckel operator can be used only to compute the exact gradient direction with a CPU time reduction of about 75%. Moreover, a further reduction could be obtained by using special parallel hardware.

3. CONTOUR GAP FILLING

Often, the Hueckel operator is not able to detect sharp corners and so the contours can present some discontinuities.
The implemented contour gap filling algorithm consists of two steps. The former fills one-pixel gaps using a 3×3 search window (figure 1), the latter fills M-pixel gaps, by a contour following algorithm which uses two search passes, a short and a long one. In the analyzed images, the biggest gap was five pixel wide (M=5).

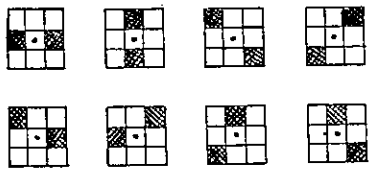


Figure 1

The first step of the procedure has been implemented in order to reach a shorter computational time in the second one (M-pixel gap filling), which performs the true contour connection.
The filling program is here briefly described: starting from the current pixel P1, a long search pass (M+1 pixels) is made (figure 2) looking for a new contour pixel P2. If it is found, the program tries to reach it by the short pass (1 pixel) in a prefixed number of steps. If it is impossible to reach P2 from P1 by the short pass, a contour discontinuity is supposed to exist and the filling is made.
The connection is not generally made between P1 and P2, but between the pixel P1V and P2V at the minimum euclidean distance. To detect P1V and P2V the program moves from P1 and P2 some pixels in clockwise and counterclockwise direction storing the couple of pixels at the minimum distance.

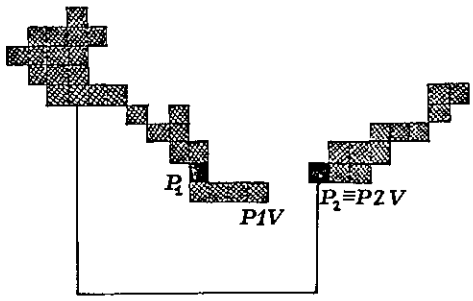


Figure 2

The pixel adjacent to P1 in a counterclockwise direction becomes the new current pixel and the procedure is repeated until the first starting pixel is reached again.
Whenever a contour is closed, the enclosed region is filled and labelled with a different value.

4. THE POLYGONAL APPROXIMATION

For each connected region corresponding to one or more overlapped objects, the contour direction is analyzed. More precisely, the "contour function" is computed. This function is the variation of the direction of the tangent to the contour (obtained by the Hueckel operator) versus the curvilinear coordinate along the object contour. For example, for a perfect square, the contour function is a step function with four horizontal steps, each of them 1/4 of the perimeter long, and $\pi/2$ radiants apart from each other.
The contour function is approximated by straight line segments [9]. This approach means an approximation of the object contour by straight line segments and circular arcs: the angle between the segment of the polygonal and the abscissa axis is related to the curvature radius of the corresponding arc in the perimeter.

5. THE RECOGNITION ALGORITHM

The segments found by the polygonal approximation are used for object recognition. The recognition algorithm compares the object with every stored model and looks for a "similarity", computed on the basis of the match between the object and the model segment sequences.
For the object under investigation and every stored model, the following steps are performed:

1) Comparison between the object and the generic model and computation of:
   - SCORE = true score
   - SCMP = maximum possible score
2) For the same object and model, associations between the corresponding segment sequences are possible. The association with the

maximum score (designed as SCMAX) is selected.

3) If:
SCMAX > KA and SCMAX/SCMP > KR
(KA,KR assigned threshold values)
the object is recognized.

The model related to SCMAX is chosen as that most similar to the object under investigation. Let us now explain in detail what SCORE and SCMP are and how the recognition algorithm works.
The model is a sequence of nsm segments characterized by five numbers:

$$\text{model segment} = (Lm_i, ALPHAm_i, BETAm_i, SCOREm_i, DISTm_{i,j}) \quad i,j=1,nsm;\ i \neq j$$

where:

- Lm_i = segment length;
- $ALPHAm_i$ = angle between the segment and the abscissa axis;
- $BETAm_i$ = difference between the ordinate of the middle point of the current and of the following segment.
- $SCOREm_i = 100*Lm_i*\cos(ALPHAm_i) / P$ (P is the total perimeter)
- $DISTm_{i,j}$ = distance between the middle points of the contour elements corresponding respectively to segment i and j.

The object is a sequence of nso segments characterized by four numbers:

$$\text{object segment} = (Lo_k, ALPHAo_k, BETAo_k, DISTo_{k,h}) \quad k,h=1,nso;\ k \neq h$$

An object segment "similar" to the model segment under consideration is looked for. If one of the conditions:

$$|Lm_i - Lo_k| / \max(Lm_i, Lo_k) < KL$$
$$|ALPHAm_i - ALPHAo_k| < KALPHA$$
$$|SBETAm_i - SBETAo_k| < KBETA$$
$$|DISTm_{i,i1} - DISTo_{k,k1}| / \max(DISTm_{i,i1}, DISTo_{k,k1}) < KDIST$$

KL,KALPHA,KBETA,KDIST: given thresholds values

is not verified, the correspondence between object segment k and model segment i is not considered. $SBETAo_k$ is the sum of all $BETAo_j$, j=k1,...,k-1 and $SBETAm_i$ is the sum of all $BETAm_l$, l=i1,...,i-1, where i1 and k1 are the indexes of the model and of the object segments corresponding to the previous match (figure 3). Alternatively, if all the conditions are verified, the quantity M_j, corresponding to the match (object segment k)-(model segment i), is computed:

$$M_j = PL_{ik}\ PA_{ik}\ SCOREm_i^j$$

where:

$$PL_{ik} = 1 - |Lm_i - Lo_k| / \max(Lm_i, Lo_k)$$
$$PA_{ik} = \cos(ALPHAm_i - ALPHAo_k)$$

The SCORE and SCMP parameters above mentioned are:

$$SCORE = \Sigma_j\ M_j$$
$$SCMP = \Sigma_j\ SCOREm_i^j$$

The comparison is performed in an ordered way to guarantee the correct succession:

- an object segment k, k=1,nso, similar to model segment 1, is looked for;
- an object segment j, j=k+1,...,nso,...,k-1, similar to model segment 2, is looked for;
- and so on

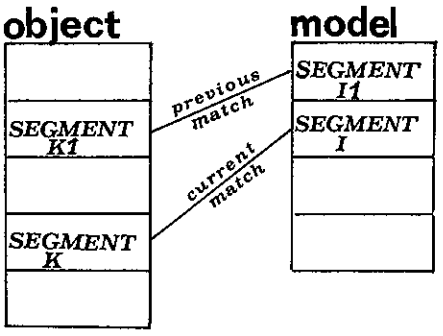


Figure 3

For the same model alternative match flows are analyzed. As soon as a sequence of object segments has been recognized, this sequence is "subtracted" from the entire polygonal, and the algorithm tries to recognize, in the remaining sequence, other objects.

6. RESULTS AND CONCLUSIONS

In this paper a vision system to locate and recognize partially visible two dimensional objects has been presented. Like any other method, it is based on a set of assumptions which implicitly define the class of tasks it can perform. The basic assumptions of the current version of the system are the following:

1) The objects rest on a plane in one of a few stable states.
2) The image plane of the TV-camera, which is the sensor used, is parallel to the plane supporting the objects.
3) The objects must be at a known distance from the TV-camera (no-size invariance).
4) A sufficient portion of the external contour (more than 30%) must be visible.

Typical applications of this procedure are the location and recognition of objects in an industrial environment, where the constraints introduced are not very limitative (for example, industrial parts on a conveyor belt). Figure 4 shows an example of the recognition step. The threshold values used are: KA=35, KR=0.5, KL=0.8, KALFA=0.25, KBETA=0.5, KDIST=0.8. The model set for this application was composed of six different objects.

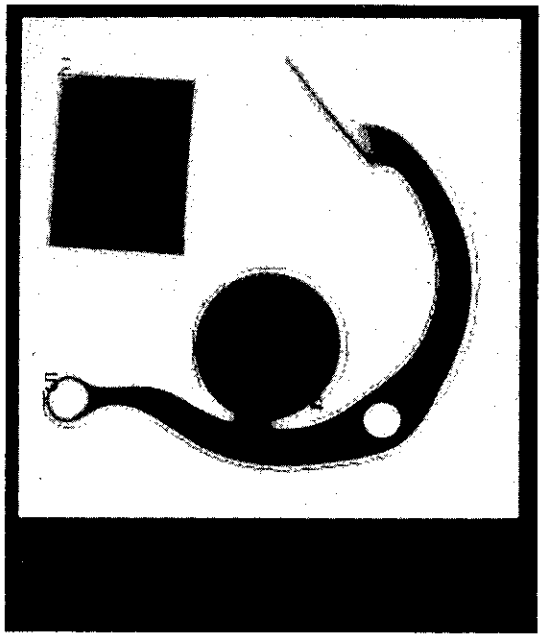


Figure 4

The performance and the computational time of the recognition phase have been sensibly reduced, in comparison with previous versions of the method [4,5], by the introduction of the parameters related with the spatial positions of the contour elements; in this way, in fact, a lot of wrong match flows are no more considered.

It is possible to get false matches using this technique. They typically occur if the part to be located is almost totally occluded, or if the exposed portion does not distinguish the part from other parts. The latter situation is not unlike the confusion a human observer would experience when dealing with scenes in which all the distinguishing features of a part are hidden because of an occlusion.

REFERENCES

[1] Perkins, W.A., A Model-Based Vision System for Industrial Parts, IEEE Trans. on Computers, (1978), C-27, 2, pp. 126-143.

[2] Turney, J.L, Trevor, N.M. and Volz, R.A., Recognizing Partially Occluded Parts, IEEE Trans. on Pattern Analysis and Machine Intelligence, (1985), PAMI-7, 4, pp. 410-421.

[3] Bolles, R.C. and Cain, R.A., Recognizing and Locating Partially Visible Objects: the Local-Feature-Focus Method, in: Pugh, A., (ed.), Robot Vision (Springer-Verlag, Berlin, 1983) pp. 43-82.

[4] Bertolino, A., Pareschi, M.T. and Raspollini, C., An Experimental Vision System to Recognize Isolated and Partially Overlapped Objects by Shape Matching, in: Cappellini, V. and Marconi, R., (eds.), Proceedings of the International Conference on Advances in Image Processing and Pattern Recognition, Pisa, Italy (North-Holland, Amsterdam, 1986), in print.

[5] Bertolino, A., Pareschi, M.T. and Raspollini, C., The Use of Contour Information as a Basis for Recognition of Overlapped Objects, Proceedings of the Second International Conference on Image Processing and its Applications, London, UK, in print.

[6] Hueckel, M.H., An Operator which Locates Edges in Digitized Pictures, J. ACM, (1971), 18, 1, pp. 113-125.

[7] Hueckel, M.H., A Local Visual Operator which Recognizes Edges and Lines, J. ACM, (1973), 20, 4, pp. 634-647.

[8] Hueckel, M.H., Erratum, J. ACM, (1974), 21, 2, p. 350.

[9] Pavlidis, T., Structural Pattern Recognition (Springer-Verlag, Berlin, 1977).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMAGE PROCESSING STRATEGIES ON TRANSPUTER ARRAYS

R CHAPMAN* T WILLEY** J G BARTKOWIAK*** T S DURRANI*

* University of Strathclyde, Glasgow, Scotland, UK
** National Semiconductor Ltd, Greenock, Scotland, UK
*** Ferranti plc, Edinburgh, Scotland, UK

This paper investigates the computation of low-level vision processing algorithms on arrays of processing cells. A class of non linear image processing algorithms serves as a vehicle for comparing a proposed linear array architecture with two dimensional or 'cellular' arrays. The design procedure uses the concurrent programming language OCCAM to algebraically manipulate algorithms, into a form which is amenable to computation on this linear processing array. Performance comparison tab assuming the underlying processor to be an INMOS 16 bit transputer, which illustrates the effectiveness of each implementation. In real time vision systems, high throughput with low latency, together with low cost support hardware are shown to be the major benefits of the proposed architecture.

1. INTRODUCTION

Real time image processing is now becoming practicable as the emergence of VLSI drives computational methods towards the use of concurrent multiprocessing based upon pipelining and parallelism. Low level image processing is an ideal candidate for concurrent architectures since the fundmental operations are well characterised and are carried out at or adjacent to the imaging sensor. The data structures are two dimensional arrays, and each pixel element is processed in combination with its nearest neighbours.

Many parallel architectures for image processing have been proposed including those which scan an image and operate on a small neighbourhood window at each step, VLSI Algorithmic Processors, bus-oriented architectures and in many examples, hybrid structures, where both temporal and spatial parallelism is exploited. A recent survey illustrating important trends is [1].

Most effort to date has concentrated on two dimensional arrays (often of simple processors) which naturally match the two dimensional nature of image processing algorithms where neighbourhood operations constitute the majority of computations. However, for many image processing architectures the most convenient input/output arrangements are from/to raster scanned systems and it is observed that here there is a lack of correspondence between the structure of the computational hardware and the image data. The data is fundamentally a sequence of vectors whilst the computational array often requires a two dimensional data array. Therefore many image processing systems require expensive support hardware in the form of reformatting buffers or frame stores. In such schemes the array cells receive their data serially or in sequential blocks.

Recently effort has been directed towards designing computational architectures which are more compatible with scanned image acquisition mechanisms [2], which has led to a reinvestigation of linear arrays [2,3]. In these architectures the raster scanned image data vectors can be loaded directly into the computational array, thereby avoiding the use of frame stores. An obvious consequence is a lower latency time, although higher level processors are required.

Reference [3] shows how the concurrent programming language OCCAM can be used to algebraically manipulate a general two dimensional convolution algorithm into a form where it could be computed on a linear processor array. Static timing results were given, assuming the array to be implemented using INMOS transputers, which illustrated that such a hardware array compared favourably with other concurrent image processing architectures.

This paper expands upon the work reported in [3]. In particular the performance of a linear transputer array is evaluated for specific image processing algorithms, rather than for a general two dimensional convolution. Results are presented for:

a) a thresholded nearest-neighbour mean filter
b) a separable median filter
c) a Sobel edge-enhancing filter

A further refinement on [3] is that the timing results presented are derived from the timing evaluator program now available as part of the VAX suite of OCCAM programs, instead of the previously reported static analysis.

2. OCCAM/TRANSPUTER PHILOSOPHY

The transputer chip family has been introduced by INMOS specially for multiprocessor applications. The essential features are that a processor, memory and four interprocessor communication links or channels are available on chip, and that these processor arrays can be programmed at a higher level than assembler, using the OCCAM language. The particular device considered in this study is the IMS T414 transputer which has a 16 bit 10 MIPS processor and 2 k bytes of on chip RAM. An introduction to OCCAM can be found in [4].

This paper utilises the design procedure outlined in [5], whereby an algorithm is first coded in the OCCAM language. The OCCAM algebra is subsequently manipulated to ensure that as much parallelism as possible is included in the computation and interprocessor communications. At each stage in the design, the OCCAM program can be compiled and run to ensure that the OCCAM algebra is still implementing the correct algorithm.

At this stage the OCCAM source code is mapped to a linear array of transputers. It should be noted that there is a computational overhead in implementing parallel processes on individual transputers, and therefore to maximise array performance, as many parallel processes as possible on each transputer should be converted to sequential computations.

An example of an OCCAM program which implements a general 3x3 convolution on a linear array can be found in [3].

3 IMAGE PROCESSING ALGORITHMS

It is difficult to perform a comparative analysis of different image processing systems, since system performance can be defined in a multiplicity of ways. Benchmark tests are difficult since, within image processing, different algorithms are often called the same name. For example, a Sobel filter on some systems applies vertical and horizontal filters and then replaces each pixel with the square root of the sum of the squares of the two relevant filter output values. Other systems avoid the square root and only use the sum of the moduli of the filter outputs. On some sophisticated systems a Sobel filter not only detects edges but also the tangents of the edges. To ease a comparative evaluation of linear transputer array architectures we will briefly describe the algorithms used in this investigation.

Consider the 3x3 neighbourhood of pixels $\{P_1, \ldots P_9\}$ represented by the figure (1)

P_1	P_2	P_3
P_8	P_9	P_4
P_7	P_6	P_5

Figure 1

3.1 Thresholded Nearest-Neighbour Mean Filter

This operation replaces the centre pixel P_9 with the average of its eight nearest neighbours if P_9 differs in magnitude from this average by greater than a fixed threshold K, otherwise P_9 remains unchanged; ie

if

$$\left| P_9 - \frac{1}{8} \sum_{i=1}^{8} P_i \right| > K$$

then

$$P_9' = \frac{1}{8} \sum_{i=1}^{8} P_i$$

else

$$P_9' = P_9$$

3.2 Separable Median Filter

This operation replaces P_9 by the median of the medians of (P_1,P_2,P_3); (P_8,P_9,P_4); (P_7,P_6,P_5), ie

if

P_2' is the median of (P_1,P_2,P_3)
P_9' is the median of (P_8,P_9,P_4)
P_6' is the median of (P_7,P_6,P_5)

then

P_q is the median of (P_2',P_q',P_6')

3.3 Sobel Edge-enhancing Filter

The Sobel operator can be regarded as the convolution of the 3x3 pixel neighbourhood with the two orthogonal masks

$$\begin{vmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{vmatrix} \quad \text{and} \quad \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Assuming the outputs from these masks to be D_x and D_y, then each pixel in the original image is replaced by values determined by

$$D = |D_x| + |D_y|$$

4 DISCUSSION OF RESULTS AND CONCLUSIONS

The only valid comparators to the proposed linear transputer array are other SIMD architectures. It was shown [3] that it is difficult to obtain valid comparative data on image processing architectures. The tables shown use the method proposed in [6] with the addition of latency. These results indicate the number of processors used in various systems. However they do not illustrate the total hardware complexity. The times reported are often only achieved by using additional expensive hardware, which is not the case for transputer arrays.

Table 1 shows a comparative evaluation for a 3x3 median filter implemented on different architectures. The results for the linear array are for the separable median filter as defined in section 3. The results for the ICL DAP [7] are also for a separable filter, but the CLIP7 [8] implements an 8-bit neighbour median and the MPP [9] is claimed to compute a pseudo-median filter.

A similar evaluation for the Sobel edge detector algorithm proved more difficult, and the results presented in Table 2 for the 2D transputer array [10] and the MMP [9] are for a general 3x3 convolution. No computational details on the Sobel implementation on the MIL DAP and the 3D computer are known to be published. This, together with the comments in section 3 on the ambiguity of the Sobel operator imply that care must be exercised in interpreting Table 2.

Table 3 compares four systems which implement 3x3 averaging filters. Details of the exact algorithms used can be gleaned from the relevant references. When comparing the performance of 2D and 1D transputer arrays the following comments are pertinent. The results for the 2D arrays [10] were obtained from a static analysis performed using the preliminary INMOS data for the IMS T424 transputer. The computational strategy outlined in [10] assumes that this device with 4K bytes of memory would be available. This device has not yet been fabricated and therefore the results given for the linear transputer array assume the currently available IMS T414 with 2K bytes of on chip memory and the timings given were obtained using the performance emulator now available as part of the OCCAM development software.

It is worthy of note that the transputer architecture discussed in this paper can implement the three algorithms at real time video rates. It has been previously shown [3] that as the number of processors in a linear array decreases the efficiency of the architecture increases and it is believed that real time performance will still be achieved with substantially reduced hardware.

This work is being extended by investigating the performance of these algorithms on smaller linear transputer arrays.

REFERENCES

[1] G H Granlund, J B Arvidson, "Computer Architectures for Image Processing", Proc 4th Scandinavian Conf on Image Analysis, Trondheim, Norway, June 1985.

[2] A L Fisher, P T Highnam, "Real-Time Image Processing on Scan Line Array Processors", CAPAIOM Workshop, 1985.

[3] R Chapman, T Willey, J G Bartkowiak, T S Durrani, "Image Processing on Linear Transputer Arrays", Proc ICASSP-86, Tokyo, Japan, April 1986.

[4] C A R Hoare, "OCCAM Programming Manual", Prentice-Hall Int, USA, 1984.

[5] R Chapman, T S Durrani, T Willey, "Design Strategies for Implementing Systolic and Wavefront Arrays using OCCAM", Proc ICASSP-85, Tampa, Florida, March 1985.

[6] S Yalamanchili, J K Aggerwal, "Analysis of a Model for Parallel Image Processing", Pattern Recognition, Vol 18, No 1, p 1-6, 1985.

[7] J B G Roberts, P Simpson, B C Merrifield, J F Cross, "Signal Processing Applications of a Distributed Array Processor", Proc IEE, pt F, Vol 139, No 6, pp 603-609, October 1984.

[8] T J Fountain, "Plans for the CLIP7 Chip", Integrated Technology for Parallel Processing, edited by S Leviadi, Academic Press, 1985.

[9] J L Potter, "Image Processing on the Massively Parallel Processor", Computer, pp 10-15, January 1983.

[10] J G Harp, J B G Roberts, J S Ward, "Signal Processing with Transputer Arrays (TRAPS)", Computer Physics Communications 37 (1985) 77-86, North-Holland, Amsterdam.

[11] J Grinberg, G R Nudd, R D Etchells, "A Cellular VLSI Architecture", IEEE Trans Computer, pp 69-81, January 1984.

	No of Processors	Interprocessor Communication	Image Size	Latency	Processing Time / Communication Time
ICL DAP (7)	32x32 x 1 bit	4 Nearest Neighbours	256x256 x 6 bit	0 (Frame)	4·5 mS / 0·22 mS = 204·5
CLIP(7) (8)	512x4 x 8 bit	8 Nearest Neighbours	512x512 x 8 bit	0 (Frame)	5·12 mS / 14 mS = 0·366
MPP (9)	128x128 x 1 bit	4 Nearest Neighbours	128x128 x 8 bit	0 (Frame)	0·017 mS / 0·8 µS = 21·25
Linear Array	256 x 16 bit	Nearest Vertical Neighbours	256x256 x 16 bit	21·85 µS	2·06 mS / 0·8 µS = 0·588

Table 1 : Performance Comparison for 3x3 Median Filter

	No of Processors	Interprocessor Communication	Image Size	Latency	Processing Time / Communication Time
ICL DAP (7)	32x32 x 1 bit	4 Nearest Neighbours	512x512 x 6 bit	0 (Frame)	2·3 mS / 0·22 mS = 10·45
RSRE Transputer Transputer Array (10)	(8x8)+1 x 16 bit	4 Nearest Neighbours + 3 Global Communicators	256x256 x 8 bit	0 (Frame)	11·5 mS / 9·1 mS = 1·26
MPP (9)	128x128 x 1 bit	4 Nearest Neighbours	128x128 x 8 bit	0 (Frame)	0·211 mS / 0·8 µS = 263·75
Hughes 3D (11)	256x256 (1 bit)	4 Nearest Neighbours	256x256 x 16 bit	0 (Frame)	54·3 µS / 20 mS = 0·0027
Linear Array	256 x 16 bit	Nearest Vertical Neighbours	256x256 x 16 bit	31·55 µS	1·83 mS / 6·25 mS = 0·293

Table 2 : Performance Comparison for Sobel Edge Detector

	No of Processors	Interprocessor Communication	Image Size	Latency	Processing Time / Communication Time
RSRE Transputer Array (10)	(8x8)+3 x 16 bit	4 Nearest Neighbours + 3 Global Communicators	256x256 x 8 bit	0 (Frame)	0·8 mSS / 4·6 mS
CLIP-7 (8)	512x4 x 8 bit	8 Nearest Neighbours	512x512 x 8 bit	0 (Frame)	0·95 mS / 14 mS = 0.068
MPP (9)	128x128 x 1 bit	4 Nearest Neighbours	128x128 x 8 bit	0 (Frame)	0·007 mS / 0·8 µS = 8·75
Linear Array	256 x 16 bit	Nearest Vertical Neighbours	256x256 x 16 bit	17·25 µS	0·88 mS / 3·5 mS = 0·251

Table 3 : Performance Comparison for 3x3 Mean Filter

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EDGE DETECTORS BASED ON NONLINEAR FILTERS

I.Pitas
Department of Electrical Engineering
University of Thessaloniki, Thessaloniki 540 06, GREECE
A.N.Venetsanopoulos
Department of Electrical Engineering
University of Toronto, Toronto M5S 1A4, CANADA

ABSTRACT The paper proposes a new class of edge detectors which are based on recently developed nonlinear filters. These filters are the order statistic filters. The performance of these edge detectors is studied and it is compared to the performance of well known edge detectors.

1. INTRODUCTION

Edge detection is a very important task in image analysis because it gives valuable information about the shape of the objects which are present in the image. Therefore edge detection has been a fuitful research topic in the recent years [1,2]. Various edge detectors have been proposed and a quantitative analysis of the performance of some edge detectors has already been done [2]. It is generally known that the performance of edge detectors deteriorates in the presence of noise. The motivation of our research is to design edge detectors which have good noise characteristics and are fast. We have based the proposed edge detectors on a new class of nonlinear filters, the order statistics filters [4,5]. These filters preserve edge information and supress certain kinds of noise (eg. impulse noise) better than linear filters [3,4]. They are also very fast.

We define as edge the border between two homogeneous regions of different luminance values. This definition implies that an edge is also a local variation of image luminance (but not vice versa). The output of the proposed edge detectors is a measure of the local luminance dispersion. If it is greater than a certain threshold, the central pixel of the filter extent is declared to be an edge point. The use of different nonlinear filters give different measures of dispersion.

Section 2 analyses the edge detectors based on order statistic filters. The computational complexity of these edge detectors is analysed in section 3. Section 4 analyses the statistical properties of these edge detectors. Conclusions are drawn in section 5.

2. EDGE DETECTORS BASED ON ORDER STATISTICS

Order (i) statistic of N random variables $x_1, x_2, .., x_N$ is called the variable x

$$x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(i)} \leq \ldots \leq x_{(N)} \qquad (1)$$

of the ordered variables $x_1, .. x_N$ .The maximum and the minimum are the extremes $X_{(1)}, X_{(N)}$.

An edge detector based on order statistics is a linear combination of order statistics described by:

$$W = \sum_{i=1}^{N} a_i x_{(i)} \qquad (2)$$

where a_i $1<i<N$ are appropriate coeeficients and x_i $1<i<N$ are image luminances of the pixels that are inside the edge detector window. The structure of the edge detector is shown in figure 1. The same structure can be used as a nonlinear image filter [3]. The only difference lies in the choice of the coefficients a_i .A sorting network which can be used in the order statistics edge detector is shown in figure 2.

The simplest order statistics edge detector is the so- called 'range edge detector':

$$W_{(1)} = x_{(N)} - x_{(1)} \qquad (3)$$

Its coefficients a_i are chosen to be:

$$a_1=-1 \quad a_N=1 \quad a_i=0 \quad i=2,N-1$$

The range edge detector gives as output the difference of the maximum and the minimum image intensities inside an image window. In the case of the range edge detector the sorting nework of figure 2 deteriorates to another simpler network which calculates only the maximum and the minimum of N numbers. Therefore the structure of the range edge detector is that of figure 3.

Another simple edge detector is the so-called quasi-range edge detector:

$$W_{(i)} = x_{(N+1-i)} - x_{(i)} \tag{4}$$

which corresponds to the following choice of coefficients a_i :

$$a_i=-1 \quad a_{N+1-i}=1 \quad a_j=0 \quad j=i,N+1-i \tag{5}$$

Another edge detector is given by the following formula:

$$W = \frac{1}{I} \sum_{i=1}^{I} W_{(i)} \tag{6}$$

and corresponds to the following choice of coefficients:

$$a_i=1/I \quad i=1,I \quad a_i=1/I \quad i=N+1-I,N \quad a_i=0 \quad i=I+1,N-I \tag{7}$$

We shall call this edge detector dispersion edge detector, for I= N/2.

Finally some other measures of dispersion which can also be used as edge detectors are the following:

$$W_A = \frac{2\sqrt{\pi}}{N(N-1)} \sum_{i=1}^{N} [i - \tfrac{1}{2}(N+1)]x_{(i)} \tag{8}$$

$$W_A = \sum_{i=1}^{\lfloor \frac{1}{2}N \rfloor} \frac{(N-2i+1)}{N(N-1)} W_{(i)} \tag{9}$$

$$W_A = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} i(N-i)(x_{(i+1)} - x_{(i)}) \tag{10}$$

2. COMPUTATIONAL COMPLEXITY OF THE ORDER STATISTICS EDGE DETECTORS

The speed of an edge detector is a very important factor which has to be taken into account especially for real time image analysis. The speed is directly connected to the computational complexity of the edge detector. We shall study first the computational complexity of the range edge detector. The calculation of the maximum of N numbers requires at least N-1 comparisons. The same number of comparisons is required for the calculation of the minimum. If the maximum and minimum are calculated simultaneously, the following number of comparisons is needed for N even and N odd respectively:

$$V(N) = 2(N-1) - \frac{N}{2} = \frac{3N}{2} - 2 \qquad \text{(N even)} \tag{11}$$

$$V(N) = 2(N-2) - \frac{N-1}{2} + 2 = \frac{3(N-1)}{2} \qquad \text{(N odd)} \tag{12}$$

where T_{comp} and T_{add} is the time required for a comparison and an addition respectively. Therefore the time required for a parallel computation of the range edge detector is approximately given by:

$$T_t = \frac{3N}{2} T_{comp} + T_{add} \tag{13}$$

The evaluation of the maximum and the minimum of a set of numbers can be done by a sorting network. Such a network is shown in the first part of figure 3. The depth of such a network is approximately given by:

$$D(N) = \lfloor \log_2 N \rfloor \tag{14}$$

where symbol $\lfloor x \rfloor$ denotes the least integer larger than or equal to x. (14) is accurate only when N is power of 2. The least possible time for fully parallel computation of the range edge detector (called critical time T_c) is:

$$T_c = (\lfloor \log_2 N \rfloor + 1)T_{comp} + T_{add} \tag{15}$$

Therefore range edge detector is very well suited for parallel computation.

The computational complexity of the dispersion edge detector is analysed in a similar way. Its performance depends on the choice of the sorting network. If the network of Figure 2 is used, N(N-1)/2 comparisons are needed. Thus the total computation time is given by:

$$T_t = \frac{N(N-1)}{2} T_{comp} + (N-1)T_{add} + NT_{mult} + T_{comp} \tag{16}$$

The critical time for number sorting by this network is given by:

$$T_c = (2N-3)T_{comp} \tag{17}$$

Thus the total critical time for the parallel calculation of the dispersion edge detector is given by:

$$T_c = (2N-3)T_{comp} + (N-1)T_{add} + T_{mult} + T_{comp} \tag{18}$$

It can be easily seen that the critical time of the range edge detector is of the order $O(\log_2 N)$ whereas the critical time of the dispersion edge detector is of the order O(N). The time reqired for serial computation of the range and the dispersion edge detectors is of the order O(N) and $O(N^2)$ respectively. Therefore the dispersion edge detector is always much slower than the range edge detector. Both algorithms require no multiplications or power evaluations and

therefore they are, much faster than most of the known edge detectors [9].

4. STATISTICAL PROPERTIES OF THE ORDER STATISTICS EDGE DETECTORS

The analysis of the statistical characteristics of order statisics and their linear combinations is a very tedious task. Some of this work has already been done by statisticians [7]. The results of this work are usually approximate and they are tabulated. Closed formulas exist only for simple cases eg. for the range w(1). If the cummulative probability function of x is F(x), the cummulative probability density function F(w(1)), the mean E(w(1)) and the variance of w(1) are given by

$$F[w_{(1)}] = n \int_{-\infty}^{\infty} [F(x+w) - F(x)]^{N-1} \, dF(x) \tag{19}$$

$$E[w_{(1)}] = \int_{-\infty}^{\infty} [1 - F^N(x) - (1-F(x))^N] dx \tag{20}$$

$$\sigma^2_{w_{(1)}} = 2 \int_{-\infty}^{\infty} \int_{-\infty}^{y} [1 - F^N(y) - (1-F(x))^N + (F(y)-F(x))^N] dx dy - E^2(w) \tag{21}$$

However (19-21) cannot give us an insight of the statistics of w(1). Statistitians have found some qualitative results [7] on the performance of the dispersion measures in the presence of outliers (eg. impulse noise):

a) The range edge detector is very sensitive to the presence of outliers.

b) The quasi-range edge detectors are relatively robust to the presence of outliers.

Because of the lack of the appropriate theory we have performed some simulations to analyze the performance of the order statistics edge detectors. We have tried to compare the performances of the range edge detector, the dispersion edge detector, the Sobel edge detector [11] and the contraharmonic edge detector [9]. The test edge used is a step edge along the y axis of the xy plane. The image intensity is 50 and 100 on its left and its right side respectively. The edge is corrupted by uniformly distributed noise of zero mean value and variance 75. The output of the each edge detector is clustered in 3 sets of points:

1) The actual edge points having mean and variance

2) The points corresponding to the low image intensity homogeneous region to the left of the edge, having mean and variance

3) The points corresponding to the high image intensity homogeneous region to the right of the edge, having mean and variance

If the edge detector is a good one, the distribution of the actual edge points is far apart and clearly distinguished from the distributions of the points corresponding to the homogeneous image regions. A measure of the distance between two probability distributions is the ratio of the difference of their mean values and the distribution variance [8]. The bigger the ratio is, the better the distinction between the two distributions. Therefore we have used the following figures of merit for the comparison of the performance of the edge detectors:

a) $V_L = \dfrac{|\bar{x}_E - \bar{x}_L|}{\sigma_L}$

b) $V_H = \dfrac{|\bar{x}_E - \bar{x}_H|}{\sigma_H}$

c) $V_{EL} = \dfrac{|\bar{x}_E - \bar{x}_L|}{\sigma_E}$

d) $V_{EH} = \dfrac{|\bar{x}_E - \bar{x}_H|}{\sigma_E}$

The higher the values of the figures of merit, the better is the performance of the edge detector. The results of the simulation are shown in Table 1. Range edge detector is clearly the best for uniform noise. The same experiments have been performed for Gaussian white additive noise. It is proven that in this case dispersion and Sobel edge detectors are better than range edge detector [9].

A second sets of experiments has been performed on a real image shown in figure 4a. The results of the range and of the dispersion edge detector are shown in figures 4b and 4c respectively. The range edge detector detects even low contrast images but it has large background noise. The dispersion edge detector has less background noise. This fact enables the use of lower threshold values.

5. CONCLUSIONS

Some new edge detectors based on nonlinear filters are proposed. they combine speed, parallel structure suitable for VLSI implementation and good noise characteristics. The very same structure can also be used for image filtering. Therefore a chip implementing such a structure can be used in a variety of applications.

REFERENCES

1. W.K. Pratt 'Digital Image Processing', Wiley Interscience, 1978

2. I.E. Abdou, W.K. Pratt 'Quantitative design and evaluation of enchancement/ thresholding' Proc. IEEE, vol.67, No.5, May 1979

3. I. Pitas, A.N. Venetsanopoulos 'Nonlinear statistical filters: A novel tool for image filtering and edge detection' Signal Proceesing, in press.

4. B.J. Justusson 'Median filtering: statistical properties' in Two dimensional digital signal processing, vol. 2 (T.S.Huang editor),Springer Verlag, 1981

5. H.A. David 'Order statistics' J.Wiley 1981

6. D.E. Knuth 'The art of computer programming', vol.3, Addison Wesley 1973

7. M.G. Kendall 'The advanced theory of statistics', vol.1, C.Griffin, 1973

8. K.Fukunaga, T.F.Krile 'A maximum distance feature effectiveness criterion' IEEE Trans. on Information Theory, vol. IT-14, No.5, PP. 780, Sept. 1968

9. I.Pitas, A.N.Venetsanopoulos 'Edge detectors based on nonlinear filters' IEEE Transactions on Pattern Analysis and Machine Intelligence, in press.

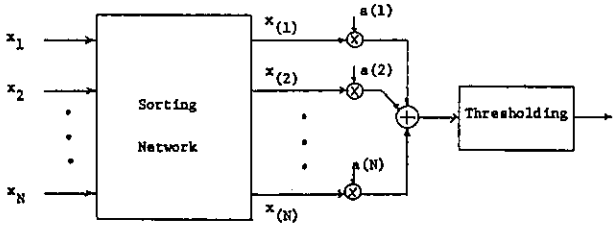


Figure 1: Structure of the edge detector

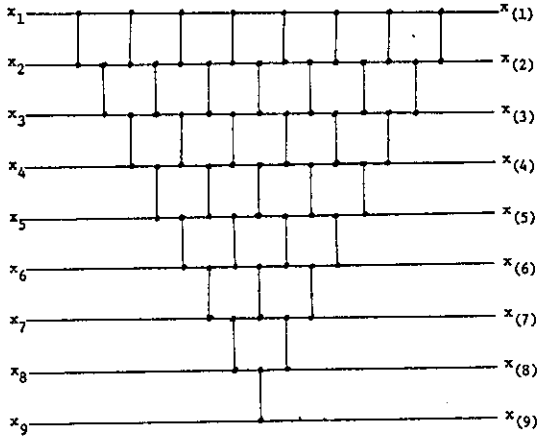


Figure 2: Sorting network

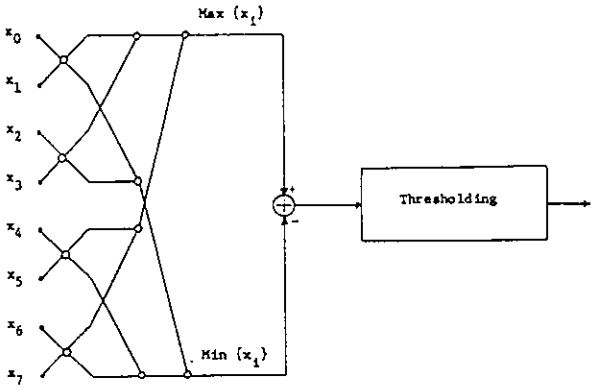


Figure 3: Range edge detector

Figure 4: (a) Test image (b) Result of the range edge detector (c) Result of the dispersion edge detector

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AUTOMATIC DECOMPOSITION OF COMPLEX OBJECTS WITH SUBPARTS RECOGNITION-CLASSIFICATION

V. Cappellini°, A. Del Bimbo°° and A. Mecocci°°°

° Dipartimento di Ingegneria Elettronica, University of Florence and IROE-C.N.R., Florence, Italy
°° Dipartimento di Ingegneria Elettronica, University of Florence, Florence, Italy
°°° Dipartimento di Ingegneria Elettronica, University of Florence and D.P.S., Florence, Italy

A digital processing system is presented for the decomposition of complex objects with the recognition-classification of all the subparts. The system comprises the following main steps: prefiltering; boundary extraction; object decomposition and subparts identification; syntactical analysis. In the typical configuration, three different image acquisitions of the same complex object in red, green and blue colours are performed (monochromatic acquisition can be carried out in a simplified configuration). Results obtained by applying the above system to the automatic analysis and decomposition of complex objects (in particular represented by printed circuit boards) are reported.

1. INTRODUCTION

The analysis of objects of complex structure, containing several components or subparts, is of high actual interest in many fields and in particular in automatic inspection and robotics. Object decomposition is to be considered one of the most important problems in the interconnected areas of digital image processing, pattern recognition and artificial intelligence. Indeed little experience has been up to now gained concerning the automatic recognition and classification of components or subparts composing complex objects.

In this paper a complete digital processing system is presented for the decomposition of complex objects with the recognition-classification of all the subparts. The system comprises the following steps: image acquisition; prefiltering; boundary extraction; object decomposition and subparts identification; syntactical analysis. In the typical configuration, three different image acquisitions of the examined complex object in red-green-blue (R-G-B) colours are performed, while a monochromatic acquisition can be carried out in a simplified approach. Figure 1 shows the flow-chart of the system with the main steps, which are described in the following.

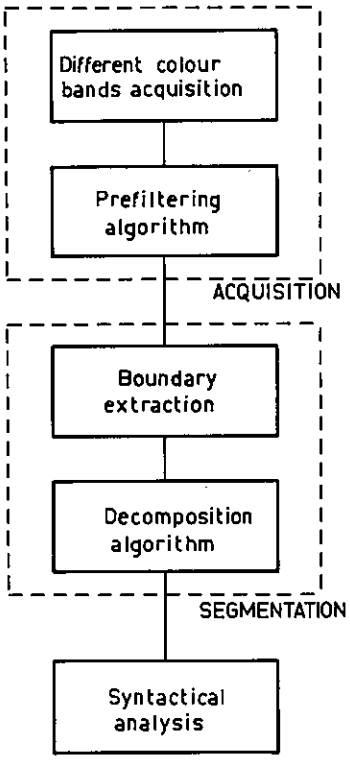


Figure 1
Flow-chart of the processing system with the main steps.

2. IMAGE ACQUISITION AND PREFILTERING

Monochromatic acquisition is currently used in analysis and processing of the global shapes of objects in single or multiple image frames. Several experiments have shown that monochromatic acquisition does not give enough information to distinguish the single parts composing a complex object. Therefore, in general three different acquisitions of the external scene (containing the objects to be examined) in R-G-B colours are here performed, to add colour information [1][2] (a monochromatic acquisition can be carried out in a simplified approach).

Further, to reduce the noise and to prepare the image data in the best way for subsequent steps, a suitable prefiltering is applied on each R-G-B image. The used prefilter is defined according to the algorithm proposed by Moring and Pietikäinen [3], suitably modified. The 3x3 A-neighbourhood of each pixel is analysed: the pixel satisfying the three following conditions in the grey-level histogram is selected to replace the central pixel

$$p(f_i) - p(f_c) > 0$$

$$\frac{p(f_i) - p(f_c)}{a_r |f_i - f_c|} < 1 \qquad \begin{cases} a_{r+1} = a_r/2, & r \in N \\ a_o = k = 10 \end{cases} \qquad (1)$$

$$p(f_i) - p(f_c) = \max_A [p(f_1) - p(f_c)]$$

where $p(f_c)$ represents the probability of the grey level of the central pixel, while $p(f_i)$ is the probability of the i-th grey level for each pixel in the A-neighbourhood; a_r is a sequence of positive numbers that decreases at each iteration (N is the set of integers). The histogram is used as an approximation of the grey-level probabilities. Iterative application of this procedure leads to a smoothed image and to a certain degree of segmentation very useful for the subsequent processing steps.

3. BOUNDARY EXTRACTION AND OBJECT DECOMPOSITION

To achieve automatic object decomposition, it is necessary to derive the significant subsets of a complex object. An initial step is performed, corresponding to a generalized boundary extraction [2][4]. The following symmetrical algorithm is applied to the three acquired R-G-B images

$$E_m(i_o,j_o) = \frac{1}{8} \sum_{i,j} |g(i_o,j_o) - g(i,j)|$$

$$i,j \in A(i_o,j_o)$$

$$E(i_o,j_o) = \frac{1}{3} \sum_{m=1}^{3} E_m(i_o,j_o) \qquad (2)$$

$$A(i_o,j_o) = \{i,j: |i - i_o| < 1 \text{ or } |j - j_o| < 1\}$$

$$\forall i,j \in [1,M]$$

where M is the image size, E_m is the contour map in the m-th colour band, E is the final contour map in the image, g(i,j) is the grey level at the (i,j) pixel and (i_o,j_o) denotes each central pixel of the 3x3 pixel block (A-neighbourhood). Thus the contour information is obtained, by using the three different colour bands.

A second step is performed to obtain homogeneous subregions of the examined object. To get a good decomposition of a complex object, grey-level clustering is not enough: spatial relations and proximity information have also to be used [5][6][7]. A decomposition threshold is chosen according also to experimental tests. With this threshold α it is possible to discriminate between homogeneous regions and contour lines, by applying the following procedure [8].

For a probable contour line

$$|G_o| > \alpha \qquad \begin{cases} G_o < 0 & i_o,j_o \text{ is root of the tree} \\ G_o > 0 & i_o,j_o \text{ links to } i,j: \\ & E(i_o,j_o) - E(i,j) \\ G_o \text{ not defined} & i_o,j_o \text{ is root of the tree} \end{cases}$$

For a probable homogeneous region

$$|G_o| \leq \alpha \qquad \begin{array}{l} \text{if a generical } i,j \text{ does not create} \\ \text{loops, then } i_o,j_o \text{ links to } i,j \\ \text{if every } i,j \text{ creates a loop then} \\ i_o,j_o \text{ is root of the tree} \end{array}$$

Here

$$G_o = \max_{A(i_o,j_o)} [E(i_o,j_o) - E(i,j)]$$

where $A(i_o,j_o)$ is defined as in the relation (2).

The used procedure starts from the pixel on the extreme lower left-hand corner and proceeds from left to the right connecting the neighbouring pixels and avoiding closed loops: the final result is the identification of a unique root of an oriented tree for each homogeneous region. Thus each region is identified unambigously by the tree which connects its pixels.

To test the reliability of the above procedure, it was compared with an interactive procedure, in which a human operator gives information about object components: very satisfactory results were obtained.

4. DEFINITION OF A DESCRIPTIVE SYNTAX

At this point, some features of the various subsets of the object(s) present in the acquired scene are extracted (as centroid position, area, inertial moments, four Fu's invariants) to allow a geometrical description of the subsets [8] [9][10] .

Mutual spatial relations among the various subsets are evaluated. These relations are based on a scheme, in which the positions are suitably coded (as: above, above and left, left, below and left, below, below and right, right, above and right) [8]. In addition, three relations - inside, outside, partially surrounding - are defined, which describe relative positions.

Information provided by the relative positions of the centroids of the various subsets is seldom good enough to specify spatial relations. Therefore, in evaluating relative positions, this procedure is followed: the angle of sight by which a particular subset is viewed by a hypothetical observer from the centroid of another subset is considered. Suitable angular functions are hence evaluated [8] .

5. EXPERIMENTAL RESULTS

Many experimental tests have been performed, according to the above processing system, by using a PDP 11-34 minicomputer with a TV camera (having colour filters) and a digitizing interface. Mechanical blocks and circuit boards with several components were analysed. In the following, experimental results regarding a circuit board with four components are reported.

Figure 2 shows the original digitized image.

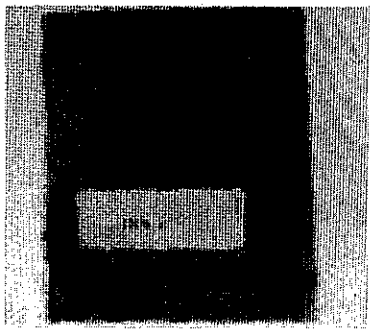


Figure 2
Original digitized image (black-white)

As it is appearing, great acquisition noise is present (to test the reliability of the presented system).

Figure 3 shows the edging result, after the R-G-B colour acquisition, prefiltering and application of the algorithm (2): the four components are appearing.

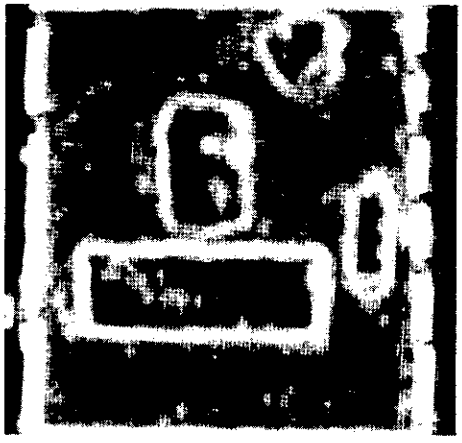


Figure 3
Edging result

Figure 4 shows the decomposition result: the four components are clearly identified (with different colours, here appearing as different grey-levels), corresponding (from the bottom to the top) to a capacitor (part 2), a resistance (part 3), an integrated circuit (part 4) and a trimmer (part 5), all inside the board (part 1).

Figure 5 shows the automatic identification of mutual spatial relations among the different componets: in particular the trimmer (part 5) is identified and its position is described in a syntactical form.

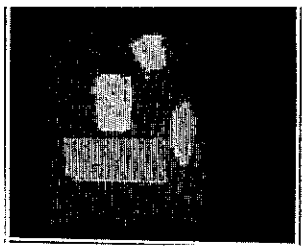


Figure 4
Decomposition result.

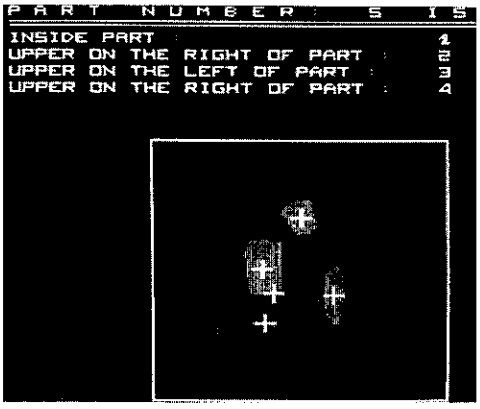


Figure 5
Automatic identification of the trimmer (part 5)

REFERENCES

[1] Robertson, T.V., Multispectral Image Partitioning (Purdue University, West Lafayette, 1973).
[2] Kettig, R.K., IEEE Trans. Geosci. Electron. 14 (1976) 19.
[3] Moring, I. and Pietikäinen, M., Experiments with Histogram Guided Image Smoothing, in: Johansen, P. and Becker, P.W. (eds.), Proceedings of The Third Scandinavian Conference on Image Analysis (Studentlitteratur, Lund, 1983) pp. 182-187.
[4] Martelli, A., Comput. Graphics Image Process. 1 (1972) 169.
[5] Brice, C.R., Artif. Intell. 3 (1970) 205.
[6] Zahn, C.T., IEEE Trans. Comput. 20 (1971) 68.
[7] Horowitz, S.L., J. Assoc. Comput. Mach. 13 (1976) 368.
[8] Cappellini, V., Del Bimbo, A. and Mecocci, A., Image and Vision Comput. 2 (1984) 109.
[9] Reddi, S.S., IEEE Trans. Pattern Anal. Mach. Intell. (1981) 240.
[10] Cappellini, V., Del Bimbo, A. and Mecocci, A., Fast Digital Image Processing Algorithms and Techniques for Object Recognition and Decomposition, in: Di Gesù, V., Scarsi, L., Crane, P., Friedman, J.H. and Levialdi, S. (eds.), Data Analysis in Astronomy (Plenum Press, New York, 1985) pp. 431-438.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMPROVED DETECTION WITH THE CROSS-AMBIGUITY FUNCTION

Ronald Abileah

SRI International
333 Ravenswood Avenue
Menlo Park, California 94025 USA

The cross-ambiguity function (CAF) and the Wigner distribution (WD) are mathematically related bilinear representations of signals. The former is used in sonar and radar to locate a signal with unknown time-delay and Doppler shift. The latter characterizes the time-frequency distribution of signal and noise. This paper shows that the WD is a convenient device for data-adaptive filtering leading to improved signal-to-noise ratio in CAF detections. The proposed approach is useful when the signal competes with strong, time-varying interference.

1. INTRODUCTION

The cross-ambiguity function (CAF) is used in sonar and radar systems to locate a signal with an unknown time delay and Doppler shift. In an active surveillance system where a signal, s(t), is transmitted and returns, with additive noise, as r(t), the time delay (τ) corresponds to range and the Doppler shift corresponds to line-of-sight target velocity. Provided that the Doppler effect can be approximated by a frequency shift (ϕ), the presence of a target is indicated by a prominent peak in the CAF

$$A_{rs}(\tau,\phi) = \int s(t - \tau/2)r^*(t + \tau/2)e^{-j\phi t}dt \quad (1)$$

The frequency shift approximation is valid when $|(2v/c)TW| \ll 1$, where v is the target velocity, c is the propagation velocity, and TW is the time-bandwidth product of the signal. The limitation imposed by this condition is not overly restrictive. If necessary, it can be overcome by a combination of the following: limiting the signal bandwidth, limiting the integration time, or resampling r(t) with a time compression corresponding to a velocity close to the true value.

For the CAF to be useful for detection of targets and estimation of their range and velocity, the peak must be discriminated from background noise. When the noise is colored and time varying, a data-adaptive filter may be required to maximize the signal-to-noise, minimize false alarms, and improve the accuracy of the range-velocity determinations. The purpose of this note is to show that the Wigner distribution (WD) is a particularly attractive device for data filtering in this application.

2. RELATIONSHIP OF CAF AND WD

The WD of a signal, z(t),

$$W_z(t,\omega) = \int z(t - \tau/2)z^*(t + \tau/2)e^{-j\omega\tau}d\tau \quad (2)$$

is an elementary time-frequency representation of a one-dimensional signal. By way of various smoothing, the WD is reduced to all other time-frequency representations (e.g., spectrogram, Rihaczek). Other useful and noteworthy properties of the WD have been covered by Claasen and Mecklenbräucker [1980a, 1980b].

Of particular interest here are certain mathematical properties of the WD and ambiguity functions, as previously derived by Sussman [1962], Ackroyd [1970], and Claasen and Mecklenbräucker [1980b].

(1) The auto-ambiguity function (AAF) and the WD are a Fourier transform pair.

(2) The relationship between the magnitude-squared CAF and the AAFs of the input signals is given by:

$$|A_{rs}(\tau,\phi)|^2 = \iint e^{j(\mu\tau-\phi\nu)}A_{rr}(\nu,\mu)A_{ss}^*(\nu,\mu)d\nu d\mu \quad (3)$$

(3) The magnitude-squared CAF is a convolution of two WDs:

$$|A_{rs}(\tau,\phi)|^2 = \iint W_r(t,\omega)W_s(t-\tau,\omega-\phi)dtd\omega \quad (4)$$

A scheme by which the above relationships are used for the adaptive filtering is depicted in Figure 1. The WD of the input r(t) is modified in some manner that will enhance the signal-to-noise ratio and is Fourier-transformed to produce the AAF of the channel. Using Equation 3, that AAF is combined with a stored AAF of the transmitted signal to form

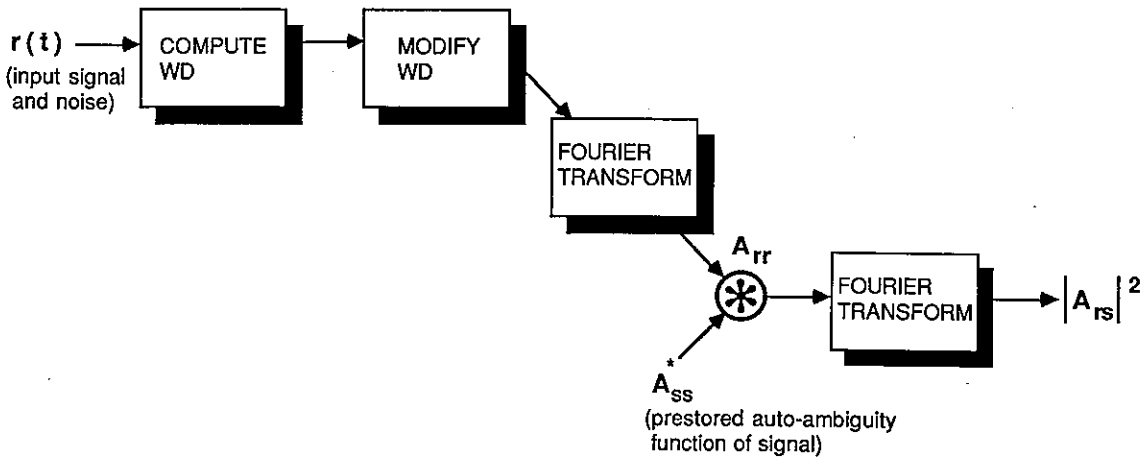


Figure 1 NOISE REDUCTION ALGORITHM

the magnitude-squared CAF. This procedure is efficient for large uncertainties in τ,ϕ of the target. Other variations on this algorithm may be more efficient when only a small span of τ,ϕ space is required or when one of the variables is fixed. In any case, the basic concept is to modify the WD and then employ the convolutions relationship (Equation 4) to produce the needed area of the CAF.

3. MODIFYING THE WD

A hypothetical example of a scenario of interest is shown in Figure 2. A time-varying signal is combined with a low-level white noise and an intense, impulsive interference. In some regions of t,ω space, the interference masks the signal and makes detection improbable. But since both signal and noise change with time, there are opportunities to detect the signal in intervals of time-frequency where only low noise level is encountered.

We wish to suppress the background noise in the WD to favor the signal component using the transfer function $H(t,\omega)$. The product $H(t,\omega)W(t,\omega)$ is a time-varying filter on the input data. Various approaches to finding $H(t,\omega)$ can be conceived. For example, Boudreaux-Bartels and Parks [1983, 1984] and Saleh and Subotic [1985] demonstrated the utility of a cookie-cutter filter, where $H(t,\omega)$ takes the value of one in the signal region and zero elsewhere. In our present problem, the location of the signal is not known a priori and may not be identifiable in a WD plot. If, however, we assume that the signal level is small relative to the interference signal, the desired transfer function is one that suppresses high-value regions of the WD, passing on to the CAF calculation the low-noise areas with remaining signal information.

Our data-adaptive filter requires an estimate of noise density in every location of t,ω space. The WD is not always positive, and it is not an energy distribution. A proper distribution function can be obtained by convolution of the WD with the two-dimensional Gaussian

$$G(t,\omega) = \frac{1}{\Delta t \Delta \omega} \exp\left(-\frac{t^2}{\Delta t^2} - \frac{\omega^2}{\Delta \omega^2}\right) \qquad (5)$$

with the choice of Δt and $\Delta\omega$ made to satisfy the equality $\Delta t \Delta \omega = 1$ [Janssen and Claasen, 1985]. Since this convolution is a window on the AAF of the signal, one might consider an appropriate choice to be as one that minimizes the attenuation of the signal's AAF. As a bonus, this convolution is also likely to minimize WD cross-terms (Flandrin, 1984) and render a useful plot of the time-frequency energy distribution.

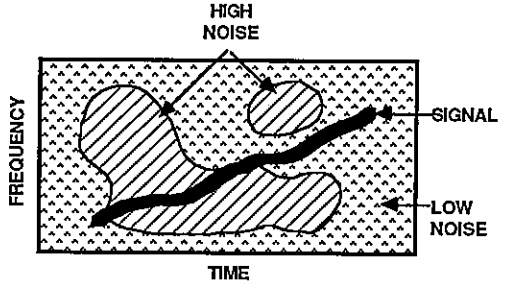


Figure 2 TIME-FREQUENCY BEHAVIOR OF SIGNAL AND NOISE

Our procedure for WD modification is given by

$$W(t,\omega) \leftarrow \log\left[1 + \frac{W(t,\omega) * G(t,\omega)}{\langle W(t,\omega)\rangle}\right] \qquad (6)$$

The normalization with $\langle W(t,\omega)\rangle$ is required to make the operation invariant to scaling of the input. The log function output will be approximately linear for small values in the WD and will suppress large values. Although the result after the Gaussian smoothing and log transformation is no longer a true WD, we are satisfied, for the reasons enumerated above, that signal structure in low-level background will be preserved.

The efficacy of this approach is shown by an example of a linear chirp signal. Figure 3 shows (1) the WD and CAF when the input is signal only, (2) the WD and CAF for signal plus white noise and band-limited impulsive interference, and (3) the CAF for the same signal plus noise, after WD noise suppression. Clearly, the interference produced high noise peaks in the CAF and made signal detection impossible. The noise suppression succeeded in restoring detection of the signal.

4. CONCLUSIONS

Using relationships inherent among the mixed two-dimensional representations, i.e., the Wigner distribution and the ambiguity function, a data-adaptive noise suppression technique has been developed and demonstrated. A simple modification of the WD, with smoothing and logarithmic transformation, mitigated strong, time-varying noise that interfered with CAF detection. This technique has applications to sonar and radar, particularly wideband systems where encounter with colored and nonstationary interference is probable.

With some obvious modifications, the same principle may be applied to the passive detection problem. In that case, the signal is not known and detection is based on a CAF of two received inputs, each containing noise and (possibly) signal. For uncorrelated noise, the WD of the two inputs will be modified independently. If noise is correlated, the time frequency modification might be based on a cross-WD [Claasen and Mecklenbräuker, 1980], a subject to be pursued in future research.

REFERENCES

Ackroyd, M.H., "Short-Time Spectra and Time-Frequency Energy Distributions," The Journal of the Acoustical Society of America, Vol. 50, 1229-1231 (1970).

Boudreaux-Bartels, G.F., "Time-Frequency Signal Processing Algorithms: Analysis and Synthesis Using Wigner Distributions," Thesis, Rice University (1983).

Boudreaux-Bartels, G.F., and T. W. Parks, "Signal Estimation Using Modified Wigner Distributions," Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Paper 22.3 (1984).

Claasen, T.A.C.M., and W.F.G. Mecklenbräuker, "The Wigner Distribution -- A Tool for Time-Frequency Signal Analysis -- Part I: Continuous-Time Signals," Philips J. Res., 35, 217-280 (1980a).

Claasen, T.A.C.M., and W.F.G. Mecklenbräuker, "The Wigner Distribution -- A Tool for Time-Frequency Signal Analysis -- Part III: Relations with Other Time-Frequency Signal Transformations," Philips J. Res., 35, 372-389 (1980b).

Flandrin, P., "Some Features of Time-Frequency Representations of Multicomponent Signals," Proc. ICASSP84, Paper 41B.4 (1984).

Janssen, A.J.E.M., and T.A.C.M. Claasen, "On Positivity of Time-Frequency Distributions," IEEE Trans. on Acoustics, Speech, and Signal Processing, 33, 1029-1032 (1985).

Saleh, B., and N. Subotic, "Time-Variant Filtering of Signals in the Mixed Time-Frequency Domain," IEEE Trans. on Acoustics, Speech, and Signal Processing, 33, 1479-1485 (1985).

Sussman, S.M., "Least-Squares Synthesis of Radar Ambiguity Functions," IRE Trans. on Information Theory, IT-8, 246-254 (1962).

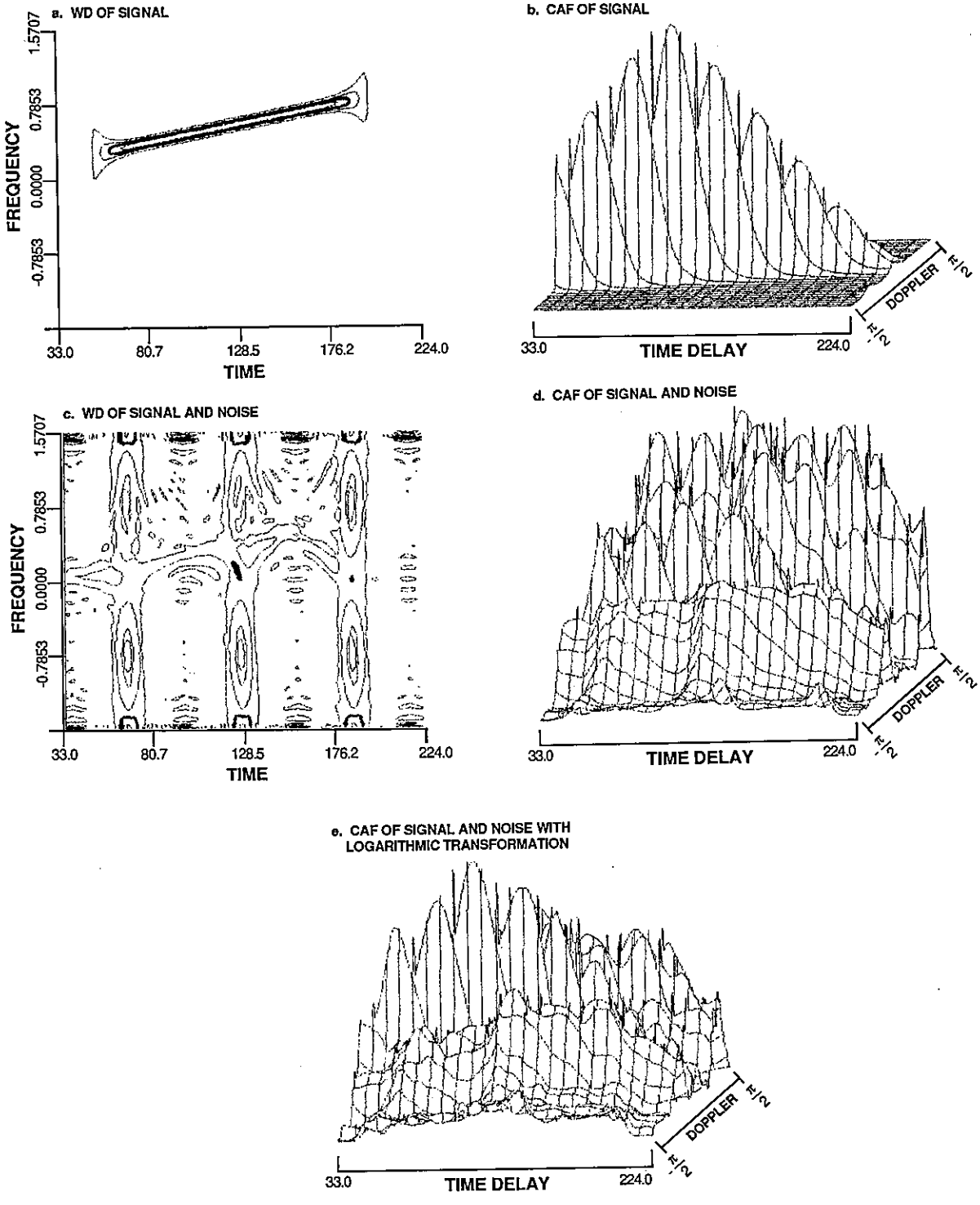


Figure 3 DEMONSTRATION

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOME RESULTS ON NEYMAN-PEARSON DETECTION WITH DISTRIBUTED RADARS*

I. Y. HOBALLAH and P. K. VARSHNEY

Syracuse University, Dept. of Electrical and Computer Engineering,
111 Link Hall, Syracuse, NY 13244-1240

This paper considers the signal detection problem when multiple radars are used for surveillance and a global decision is desired. Local decisions are fed to a data fusion center where a global decision is obtained based on a given fusion rule. Neyman-Pearson criterion is used for system optimization. Two-hypothesis two-detector problem is considered. An example is presented for illustration.

I. INTRODUCTION

Theory of signal detection using a single radar is very well understood [1,2]. The Bayesian approach to the optimum detection problem requires the knowledge of the a priori probabilities and the costs. Optimum detection rule is then obtained which minimizes the average cost of detection. For most radar detection problems, the Bayesian approach is inappropriate because the required information, i.e., a priori probabilities and costs, may not be available. For this reason, Neyman-Pearson criterion, which does not require the above knowledge, is employed extensively in radar detection systems.

There are two major options for signal processing with multiple sensors. The first one involves the transmission of all of the sensor observations (raw data) to a central processor. This requires transmission of sensor information without delay and with a large communication bandwidth. The second option is to have distributed signal processing, i.e., some or all of the signal processing is done at the sensors. In this case, results can be available locally or partial results are transmitted to a data fusion center where global results are then available. This second option with distributed processing is more attractive for many applications due to cost, reliability, survivability and communication bandwidth considerations.

Some recent work on the detection problem with multiple sensors has been reported in the literature (e.g. [3-8]). Most of the literature on distributed detection has followed the Bayesian approach. In this paper, we develop Neyman-Pearson decision theory for signal detection using multiple radars. We assume the structure shown in Figure 1, i.e., individual decisions from the radars are fed to a data fusion center which yields the global decision. A constraint on the probability of false alarm of the overall system (global decision) is placed and the probability of miss of the overall system is minimized. Decision rules at individual detectors are obtained. These rules are functions of the data fusion scheme being employed and are, in general, coupled. In Section II, we formulate the problem and define the notation and terminology. We consider the problem of binary hypothesis testing using two detectors. The results obtained in this paper can be generalized to include more detectors in a similar manner [10]. In Section III, we derive Neyman-Pearson decision rules at individual detectors which optimize the overall system for a given fusion rule. Special cases of "AND" and "OR" data fusion rules are considered. A specific example is presented in Section IV. Finally, the results are summarized in the last Section.

II. PROBLEM STATEMENT

We consider a binary hypothesis testing problem with the following two hypotheses:

H_0: Target is absent, and H_1: Target is present.

We consider the system structure shown in Figure 1 where a data fusion center is used along with the distributed sensors. The observations at each detector are denoted by $\underline{y}_i$, i=1,2. We further assume that the observations at the individual detectors are statistically independent and that the conditional probability density functions $p(\underline{y}_i|H_j)$; i=1,2, j=0,1, are known. Each detector employs a decision rule $g(\underline{y}_i)$ to make a local decision u_i, i=1,2, where

*This work was supported in part by RADC contract F30602-81-C-0169 and in part by NSF grant INT-8407317.

$$u_i = \begin{cases} 0 & \text{if detector } i \text{ decides } H_0 \\ 1 & \text{if detector } i \text{ decides } H_1 \end{cases} \quad (1)$$

The data fusion center determines the overall or global decision for the system u, based on individual decisions, i.e., $u=f(u_1, u_2)$.

The goal of this work is to develop Neyman-Pearson decision theory for detection systems with multiple sensors. For this, we need to define the probability of false alarm P_F, the probability of miss P_M, and the probability of detection P_D of the overall system:

$$\begin{aligned} P_F &= \text{Prob } (u=1|H_0) \\ P_M &= \text{Prob } (u=0|H_1) \\ P_D &= \text{Prob } (u=1|H_1). \end{aligned} \quad (2)$$

The probability of miss and the probability of false alarm for individual detectors can be defined in a similar manner and are denoted by P_{Mi} and P_{Fi}, i=1, 2, respectively.

The problem then is to obtain decision rules at the individual detectors which minimize the probability of miss, P_M, under the constraint that the probability of false alarm P_F satisfies $P_F \leq \beta$. In the next section, we employ the Lagrange multiplier method for the solution of the problem. An optimum data fusion rule for multiple sensor detection systems, when the decision rules of the individual detectors are known, is also derived.

III. DISTRIBUTED NEYMAN-PEARSON DETECTION

We consider the two-hypothesis two-detector Neyman-Pearson detection problem. We assume that the conditional densities $p(\underline{y}_i|H_j)$, i=1, 2, j=0, 1, and the fusion rule $f(u_1, u_2)$ are known. We wish to maximize P_D (or equivalentely minimize P_M) under the constraint that P_F satisfies the inequality $P_F \leq \beta$. Following the approach taken in classical Neyman-Pearson detection, we form the function:

$$F = P_M + L\,[P_F - \beta] \quad (3)$$

where L is the Lagrange multiplier.

In order to be able to express P_F and P_M in terms of the probability of false alarm and the probability of miss of the individual detectors, i.e., P_{Fi}'s and P_{Mi}'s, we define the following probabilities.

$$P_{ijk} = \text{Prob } (u=k|\, u_1=i,\, u_2=j),\ i,j,k,=0,1. \quad (4)$$

Then we may express P_M and P_F as follows

$$\begin{aligned} P_M &= P_{000}P_{M1}P_{M2} + P_{010}P_{M1}\,(1-P_{M2}) \\ &+P_{100}\,(1-P_{M1})\,P_{M2} + P_{110}\,(1-P_{M1})\,(1-P_{M2}) \end{aligned} \quad (5)$$

and

$$\begin{aligned} P_F &= P_{001}\,(1-P_{F1})\,(1-P_{F2}) + P_{011}\,(1-P_{F1})\,P_{F2} \\ &+ P_{111}\,P_{F1}\,P_{F2} + P_{101}\,P_{F1}\,(1-P_{F2}) \end{aligned} \quad (6)$$

Expanding, rearranging terms in (5) and (6) and then substituting in (3), we can express F as

$$F = C_{M1}\,P_{M1} + K_{11} + L\,[C_{F1}\,P_{F1} - \beta + K_{21}] \quad (7)$$

where

$$\begin{aligned} C_{M1} &= P_{M2}(P_{000} - P_{010} - P_{100} + P_{110}) \\ &+ (P_{010} - P_{110}) \\ C_{F1} &= P_{F2}(P_{001} - P_{011} - P_{101} + P_{111}) \\ &+ (P_{101} - P_{001}) \\ K_{11} &= P_{100}\,P_{M2} + P_{110}\,(1 - P_{M2}) \\ K_{21} &= (P_{011} - P_{001})\,P_{F2} + P_{001}. \end{aligned}$$

It should be noted that in the above formulation system-wide performance is being optimized rather than the optimization of each individual detector. The decision rules obtained in this manner will not, in general, be the same as the ones obtained when the detectors are treated independently of each other. In fact, the decision rules at the individual detectors and their computations will be coupled. Now we proceed with the solution of the problem. While deriving the decision rule at one detector, it would be assumed that the other detector has already been designed and the decision rule will be obtained in terms of the detector already designed. A simultaneous solution of the conditions obtained would yield the desired decision rules.

We may rewrite (7) as

$$F_1 = \frac{F}{C_{M1}} = P_{M1} + \frac{L\,C_{F1}}{C_{M1}}\left[P_{F1} - \frac{\beta - K_{21}}{C_{F1}}\right] + \frac{K_{11}}{C_{M1}} \quad (8)$$

Minimization of F_1 yields the following likelihood ratio test (LRT) at the first detector

$$\Lambda_1(\underline{y}_1) = P(\underline{y}_1|H_1)/P(\underline{y}_1|H_0) \underset{H_0}{\overset{H_1}{\gtrless}} LCF_1/C_{M1} = t_1' \quad (9)$$

where t_1' is the solution of

$$P_{F_1} = \int_{t_1'}^{a} P_{\Lambda_1|H_o}(\Lambda_1|H_0)\,d\,\Lambda_1 = \beta_1 = (\beta - K_{21})/C_{F1} \quad (10)$$

Observe that the threshold of the first detector is a function of the fusion rule and also the probability of false alarm of the second detector and thus, the second threshold. Similarly we may obtain the LRT at the second detector.

Now, we present the results for the problem of finding the best fusion rule (finding P_{ijk}) when the individual detectors have already been designed. Knowing P_{Mi} and P_{Fi}... i=1, 2, we we can write [10].

$$(1 - P_{F1})\,(1 - P_{F2})/P_{M1}\,P_{M2} \underset{H_0}{\overset{H_1}{\gtrless}} L,$$

$$(1 - P_{F1})\,P_{F2}/P_{M1}\,(1 - P_{M2}) \underset{H_0}{\overset{H_1}{\gtrless}} L,$$

$$P_{F1}\,(1-P_{F2})/P_{M2}\,(1-P_{M1}) \underset{H_0}{\overset{H_1}{\gtrless}} L, \text{ and,}$$

$$P_{F1}\,P_{F2}/(1 - P_{M1})\,(1 - P_{M2}) \underset{H_0}{\overset{H_1}{\gtrless}} L. \qquad (11)$$

By changing L and solving for P_{ijk}, we will be able to find the optimum fusion rule which minimizes P_M when $P_F \leq \beta$. Next we consider two special cases, namely we obtain results for two specific fusion rules "AND" and "OR".

Special cases:

"AND" Fusion Rule

In this particular case, we have

$P_{000} = P_{100} = P_{010} = P_{111} = 1$

$P_{001} = P_{101} = P_{011} = P_{110} = 0$

Therefore,

$C_{Mi} = 1 - P_{Mj}$ $\qquad C_{Fi} = P_{Fj}$ and,

$K_{2i} = 0 \qquad i \neq j \quad i,j=1,2$

the two LRT's and the corresponding equations for the thresholds are:

$$\Lambda_i(y_i) \underset{H_0}{\overset{H_1}{\lessgtr}} t_i' = L\,P_{Fj}/1-P_{Mj}$$

$$\int_{t_1'}^{a} P_{\Lambda_i|H_0}(\Lambda_i|H_0)\,d\Lambda_i = \beta/P_{Fj} \qquad i \neq j \quad i,j=1,2 \qquad (12)$$

"OR" Fusion Rule

In this case,

$P_{000} = P_{101} = P_{011} = P_{111} = 1$

$P_{001} = P_{100} = P_{010} = P_{110} = 0$

therefore,

$C_{Mi} = P_{Mj}$, $C_{Fi} = 1-P_{Fj}$

$K_{2i} = P_{Fj} \qquad i \neq j \quad i,j=1,2$

the two LRT's and the corresponding equations for the thresholds for the "OR" fusion rule are

$$\Lambda_i(y_i) \underset{H_0}{\overset{H_1}{\lessgtr}} L\,(1-P_{Fj})/P_{Mj} = t_i' \qquad i \neq j \quad 1,j=1,2$$

$$\int_{t_1'}^{a} P_{\Lambda_i|H_0}(\Lambda_i|H_0)\,d\Lambda_i = (\beta-P_{Fj}) \qquad (13)$$

In the next section, we present an example.

IV. EXAMPLE

Let us assume that the observations under the two hypotheses are exponentially distributed, i.e.,

$P(y_i|H_0) = \tau \exp(-\tau y_i)$

$P(y_i|H_1) = 2\tau \exp(-2\tau y_i) \qquad \tau=1,2;\ \tau>0 \quad y_i \geq 0$

$P(y_i|H_j) = 0 \qquad i=1,2,\ j=0,1,\ y_i<0 \qquad (14)$

The likelihood ratios and the LRT's at the detectors are given by:

$\Lambda_i(y_i) = 2 \exp(-\tau y_i) \qquad i=1,2 \qquad (15a)$

$$y_i \underset{H_0}{\overset{H_1}{\lessgtr}} -(1/\tau)\ln(t_i'/2) = t_i' \qquad i=1,2. \qquad (15b)$$

The probability of false alarm and the probability of miss for the two detectors are given by:

$$P_{Fi} = \int_0^{t_i} \tau \exp(-\tau y_i)\,dy_i = 1-(t_i'/2) \qquad (16a)$$

$$P_{Mi} = (t_i'/2)^2 \qquad i=1,2. \qquad (16b)$$

The two thresholds also satisfy the following sets of equations:

$$P_{Fi} = 1-t_i'/2 = (\beta-K_{2i})/C_{Fi} \qquad i=1,2 \qquad (17a)$$

and, $\qquad L = C_{Mi}\,t_i'/C_{Fi} \qquad (17b)$

A simultaneous solution of the above equations yields the desired thresholds. The solution requires the knowledge of the fusion rule.

Next, we present the results for the "AND" and "OR" fusion rules. Details are provided in [10].

"AND" Fusion rule

In this case, we have

$C_{Fi} = 1-(t_j'/2)$

$C_{Mi} = 1-(t_j'/2)^2 \qquad i \neq j \quad i,j=1,2$

$K_{2i} = 0$

and, the thresholds t_i $i \neq 1,2$ which lead to the optimum solution are

$$t_i = -(1/\tau)\ln(1-\sqrt{\beta}) \qquad i=1,2 \qquad (18)$$

"OR" Fusion rule

In this case

$C_{Fi} = t_j'/2$

$C_{Mi} = (t_j'/2)^2 \qquad i \neq j \quad i,j=1,2.$

$K_{2i} = P_{Fj} = 1-(t_j'/2)$

$$t_i = -((1/\tau)\ln\sqrt{1-\beta}\,). \qquad i=1,2. \qquad (19)$$

The ROC for both cases are shown in Fig. 2. For this special example the "AND" case is superior to the "OR" case.

V. SUMMARY AND CONCLUSIONS

In this paper, we have considered the signal detection problem when multiple radars are used for surveillance and a global decision is desired. Local decisions are fed to a data fusion center where a global decision is obtained when a fusion rule is given. Neyman-Pearson criterion for signal detection is used for system optimization. A constraint on the probability of false alarm is placed and the probability of miss of the overall system is minimized. The decision rules and their computation at individual detectors are coupled. The decision rules depend upon the fusion rule. We have considered the special cases of "AND" and "OR" fusion rules. We have also presented an example for illustration. While computing the decision rules, one may obtain multiple solutions. Only the feasible solutions are to be kept. Here we considered the two-hypothesis two-detector problem. When the thresholds are known, a fusion rule was derived using the same criterion. Work can be generalized to more hypotheses and more detectors in a straightforward manner [10].

REFERENCES

[1] H.L. Van Trees, Detection, Estimation and Modulation Theory, Volume I, J. Wiley, 1969.

[2] J.V. Difranco and W.L. Rubin, Radar Detection, Englewood Cliffs, NJ, Perentice-Hall, 1968.

[3] R.R. Tenney and N.R. Sandell, "Detection with Distributed Sensors," IEEE Trans. Aerospace and Electronic Systems, Volume 17, Number 4, Pages 501-509, July 1981.

[4] G. Lauer and N.R. Sandell, Jr., "Distributed Detection for Known Signals in Correlated Noise," TP-131, ALPHATECH Inc., Burlington, MA, March 1982.

[5] L.K. Ekchian and R.R. Tenney, "Detection Networks," Proceedings of the 21st IEEE conference on Decision and Control, Orlando, FL, December 1982, Pages 686-691.

[6] H.J. Kushner and A. Pacut, "A Simulation Study of a Decentralized Detection Problem," IEEE Trans. Automatic Control, Volume 27, No. 5, Pages 1116-1119, October 1982.

[7] D. Teneketzis, "The Decentralized Wald Problem," Proc. 1983 American Control Conference, San Francisco, CA, 1983.

[8] D. Tenketzis, "The Decentralized Quickest Detection Problem," Proc. 21st IEEE Conference on Decision and Control, Orlando, Florida, December 1982.

[9] Z. Chair and P.K. Varshney, "Optimum Data Fusion in Multiple Sensor Detection Systems," IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-22, pp. 98-101, January 1986.

[10] I. Hoballah, Ph.D. Dissertation in progress, Syracuse University.

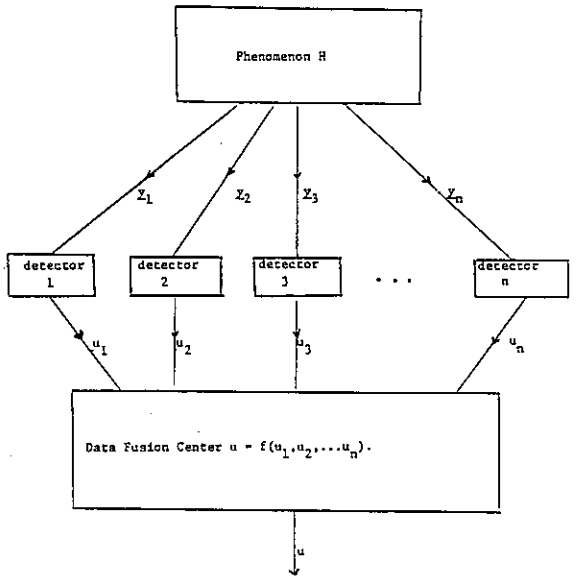


Fig. 1. Distributed Sensor System with Data Fusion Center

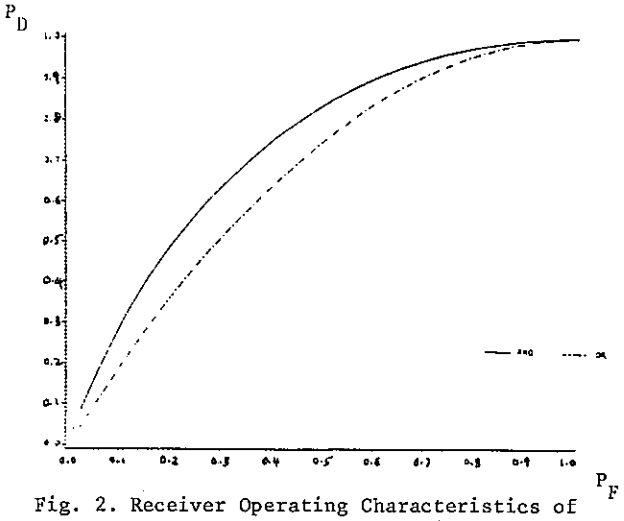


Fig. 2. Receiver Operating Characteristics of the Two Detector System with AND and OR Fusion Rules

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N1.3

AN ON-LINE ADAPTIVE ALGORITHM FOR SIGNAL PROCESSING USING SVD

Callaerts D.[+], Vanderschoot J., Vandewalle J., Sansen W.

ESAT Laboratories, Katholieke Universiteit Leuven
Kardinaal Mercierlaan 94, 3030 Heverlee, Belgium

The singular value decomposition is a numerical technique that already proved its usefulness in signal processing. This work presents an adaptive algorithm that computes the SVD with an on-line strategy for real-time applications.

1. INTRODUCTION

Many problems in signal processing are concerned with the determination of a set of numbers in a 'weight vector' w, such that the map

$$w \rightarrow \frac{w^T X X^T w}{w^T w} \qquad (1.1)$$

for a given data matrix X reaches an extremal value.
In the linear least squares problem e.g. (adaptive filtering [1], linear prediction [2],..) w should result in a minimal value for this mapping, i.e. the squared error.
Total linear least squares [3,4] also looks for a w so that the map is minimal, in order to estimate the noise in the dataset.
In methods using principal component reduction (image processing [5], modal analysis [6], identification, model reduction,..) a set of r weight vectors, according to the r largest mapping values, is determined.

A main tool for these kinds of problems is the singular value decomposition (SVD) of the data matrix X : [7]

$$X_{p,q} = U_{p,p} \Sigma_{p,q} V^T_{q,q} \quad (p < q) \qquad (1.2)$$

where U and V are orthonormal matrices and $\Sigma = \mathrm{diag}[\sigma_1 \ldots \sigma_p]$ with σ_i the singular values. The columnvectors u_i $(1 \leq i \leq p)$ of the matrix U constitute an orthonormal set of extremal value weight vectors.

The singular values σ_i provide information about the noise level, the energy, the rank of X, ... and serve as a guideline in the actual selection of the weight vectors.

2. OFF-LINE COMPUTATION OF THE SVD

The wide acceptance of the SVD as a tool in numerical algebra is mostly due to the existence of a stable and accurate Golub algorithm for its computation [7].
However this algorithm implements a typical off-line strategy of data analysis : all data should be available at once.

In signal processing applications, this off-line technique has a number of important drawbacks. First, the dimensions (p,q) of the data matrix X can be very large and this often results in an inadmissible computation time.
No wonder that big efforts have been made to reduce this objection (parallel computing, partial SVD, acceleration steps ...).

Furthermore, an off-line strategy is not optimal with regard to time variations. The used method should be able to update the weight vectors regularly.

Moreover the decomposition using Golub's algo-

[+]Sponsored by the Belgian I.W.O.N.L.

rithm is complete : it computes the U-, Σ - and V-matrix, while in most cases only the U-matrix (and eventually the singular values) is required at any time e.g. for adaptive AR-modelling. Clearly an on-line technique for the computation of the SVD in signal processing applications is desired.

3. A NEW ADAPTIVE ON-LINE ALGORITHM

Suppose that the data matrix X_k is given by :

$$X_k = [\, x_1 \; x_2 \; \dots \; x_k \,] = [\, X_{k-1} \quad x_k \,] \qquad (3.1)$$

with each x_i a p-vector.
If a good estimate $\hat{U}_{k-1}$ of the U-matrix in the SVD of X_{k-1} is known, then

$$(\hat{U}_{k-1}^T X_k)(X_k^T \hat{U}_{k-1}) = \qquad (3.2)$$
$$(\hat{U}_{k-1}^T X_{k-1})(X_{k-1}^T \hat{U}_{k-1}) + (\hat{U}_{k-1}^T x_k)(x_k^T \hat{U}_{k-1})$$

is a near diagonal matrix.
This matrix is made more diagonal by some convergent symmetric eigenvalue algorithm (Jacobi, QR, power method,...) that determines an orthonormal transformation Q_k. [7]
The estimate $\hat{U}_k$ of the U-matrix in the SVD of X_k is then given by

$$\hat{U}_k = \hat{U}_{k-1} \; Q_k \qquad (3.3)$$

The following scheme has been adopted :

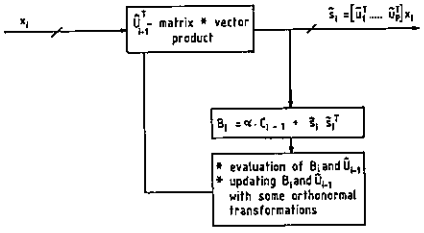


Fig.1. SVD-based signal processing algorithm.

This leads to the following algorithmic scheme:

1. Initialisation
$$\hat{U}_0 = I \; , \quad C_0 = 0 \qquad (3.4)$$
2. Orthonormal projection of data vector x_i
$$\hat{s}_i = \hat{U}_{i-1}^T \; x_i \qquad (3.5)$$
3. Accumulation step
$$B_i = \alpha^2 . C_{i-1} + \hat{s}_i \hat{s}_i^T \qquad (3.6)$$
4. Reduction of the norm of the off-diagonal elements by orthonormal transformations
$$C_i = Q_i^T B_i \; Q_i \qquad (3.7)$$
5. Updating the current estimate of the U-matrix
$$\hat{U}_i = \hat{U}_{i-1} \; Q_i \qquad (3.8)$$

4. DISCUSSION

The factor α in the second step has the same effect as a convolution of the datamatrix X_k with an exponential function, so that

$$\overset{x}{X}_k = [\, \alpha^{k-1} x_1 \; \dots \; \alpha . x_{k-1} \quad x_k \,] \qquad (4.1)$$

The $\hat{U}$-matrix is updated in each cycle while the diagonality of B_i is a measure of correctness of the estimate of U.

The reader should notice the essential difference with the method of the normal equations. The accumulation in B_i is not performed with raw data vectors (x_i) but with orthonormal transformations ($\hat{s}_i$) of them. This preserves the signal to noise ratio, and results in an almost diagonal B_i-matrix. Off-diagonal elements therefore need only a limited precision. This form of B_i also reduces the amount of computations for the updating of C_i and $\hat{U}_{i-1}$ (e.g. when using a Jacobi strategy).

Another very useful advantage of this algorithm is that the speed can be improved by introducing parallel computation at all levels.

5. CONVERGENCE OF THE NEW ALGORITHM

In order to prove the convergence of the presented algorithm, a 'worst-case study' is made :

$$\exists \, M, \forall \, x_i \in \mathbb{R}^p : \|x_i\| \leq M \qquad (5.1)$$

for i = 1,2,...,k,...

$$B_i = C_{i-1} + x_i x_i^T \qquad (5.2)$$

(accumulation without down-scaling)

$$C_i = Q_i^T B_i Q_i \qquad (5.3)$$

reduction of the off-diagonal norm until

$$\|off(C_i)\|_F \leq \|off(B_i)\|_F \cdot \varepsilon \qquad (5.4)$$

Here the following equations hold :

$$C_i = diag(C_i) + off(C_i) \qquad (5.5)$$

$$\|C_i\|_F^2 = \|diag(C_i)\|_F^2 + \|off(C_i)\|_F^2 \qquad (5.6)$$

$$= \|B_i\|_F^2$$

since orthonormal transformations on B_i do not change its norm.

Using the following inequalities :

$$\|off(B_i)\|_F \leq \|off(C_{i-1})\|_F + \|off(x_i x_i^T)\|_F \qquad (5.7)$$

$$\|off(x_i x_i^T)\|_F^2 \leq M^4 \cdot \frac{p-1}{p} \qquad (5.8)$$

and with condition (5.4), one can prove that

$$\|off(C_i)\|_F^2 \leq \frac{\varepsilon^2 (1-\varepsilon^k)^2}{(1-\varepsilon)^2} \cdot \frac{M^4 (p-1)}{p}$$

For $k \to \infty$ a very pessimistic upper bound is found for the Frobenius-norm of the off-diagonal elements of C_i :

$$\|off(C_\infty)\|_F^2 \leq \frac{\varepsilon M^4 (p-1)}{(1-\varepsilon)^2 p} \qquad (5.10)$$

While $\|C_i\|_F^2 = \|B_i\|_F^2$ gives a measure for the present energy in the datamatrix, it is a monotone rising function for a growing matrix. From (5.6) it can be seen that the energy in the off-diagonal remains bounded. The process thus converges to a diagonal form.

Each particular application involves the choice of the parameters α and ε.

6. PRACTICAL APPLICATION

The presented process can be applied to the problem of extracting the weak fetal electrocardiogram (FECG) from abdominal recordings, disturbed by the much stronger maternal electrocardiogram (MECG). [8,9,10]

For the third algorithmic step, only one Jacobi-rotation was applied per cycle so that

$$\varepsilon^2 = 1 - \frac{2}{p(p-1)} \qquad (6.1)$$

The computations were carried out with

$$\alpha^2 = 1 - 2^{-f}, \quad f=8 \qquad (6.2)$$

The factor f has to be chosen appropriately since otherwise the system looses too much information from the past.

In this application, it was possible to use fixed point precision, which simplified and speeded up the computational work a lot.

Figure 2A shows a typical 6-channel measurement set, while figure 2B gives the adaptively computed separated signals.

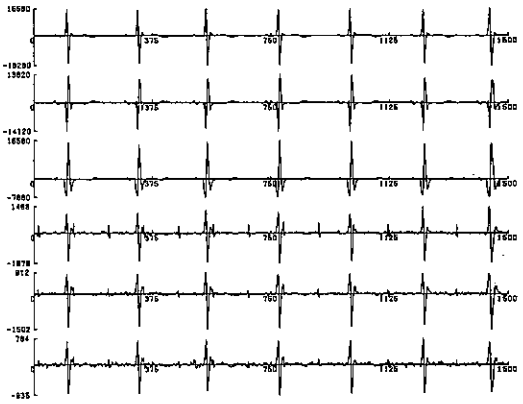


Fig. 2A. 6-channel FECG-MECG measurement set.

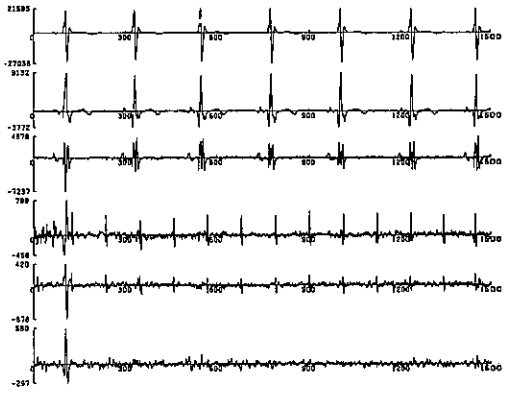


Fig. 2B. Resulting separated signals.

For this specific application an 8-channel microprocessor system using the TMS-32010 was designed.

Figures 3A and 3B compare the evolution of the diagonal elements of C_i as a function of time for f=8 and f=5 respectively.

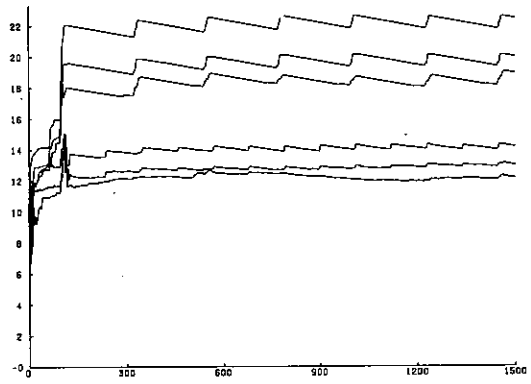


Fig. 3A. Logarithmic evolution of the diagonal elements of C_i for f=8 : After convergence they remain separated.

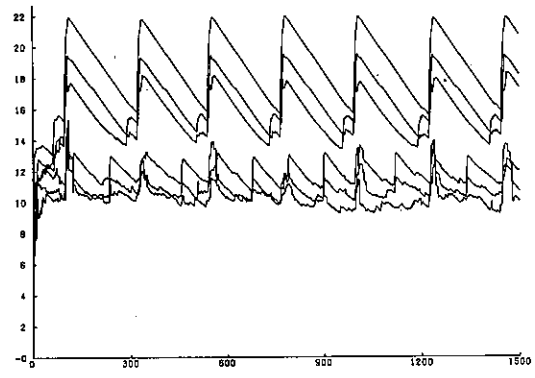


Fig. 3B. Logarithmic evolution of the diagonal elements of C_i for f=5 : no separation occurs.

7. CONCLUSIONS

In this work a new on-line adaptive algorithm is proposed in order to design a real-time implementation of the SVD in signal processing applications.

The main advantage of this technique with respect to other adaptive linear least squares methods (like L.M.S.) is that the resulting SVD allows an accurate interpretation and processing of the problem at hand. [11]

REFERENCES

[1] Widrow, Adaptive Noise Cancelling: principles and applications, Proc.IEEE, vol 63,no 12 (1975)

[2] Makhoul,J.,Linear Prediction: a tutorial review, Proc.IEEE, vol.63,no 4,pp.561-580 (1975)

[3] Van Huffel,S.,The Total Least Squares Problem: properties, applications and generalisation, SIAM Journal on Num.Analysis (1984) (submitted for publication)

[4] De Moor,B.,Vandewalle,J.,The uniqueness versus the non-uniqueness principle in the identification of linear relations from noisy data,submitted to The American Statistician,1986.

[5] Hansen,P.C.,Nielsen,H.B.,Singular Value Decomposition of images,Proc. of 3th Scandinavian Conf. on Image Analysis, Copenhagen, (1983)

[6] Leuridan,J.,Brown,D.,Allemang,R.,Time Domain Parameter Identification Methods for linear Modal Analysis: a unifying approach, subm. to ASME paper no.85,Journal of Vibration, Acoustics,Stress and Reliability in Design, (1985)

[7] Golub,G.H.,Van Loan,C.F.,Matrix Computations (North-Oxford Academic 1983).

[8] Vanderschoot,J.,Vantrappen,G.,Janssens J. et al.,The use of singular value decomposition in multilead abdominal FECG, Europ.J. Obstet.Gynec., reprod.Biol.20 (1985)

[9] Vandewalle,J.,Vanderschoot,J.,De Moor,B., Source separation by adaptive SVD, Proc. ISCAS 1985,pp 1351-1354

[10] Callaerts,D.,Vanderschoot,J.,Sansen,W.,et al An adaptive on-line method for the extraction of the complete FECG from abdominal multilead recordings, Perinatal Monitoring Nottingham (1985) subm.J.of Perinatal Med.

[11] Lawson,C.L.,Hanson,R.J.,Solving least squasquares problems (Prentice Hall Series, Englewood Cliffs, 1974)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A LOG-T DETECTOR IN K-DISTRIBUTED CLUTTER

Andrzej JAKUBIAK

Warsaw University of Technology
Institut of Telecommunications
Nowowiejska 15/19
00-665 Warszawa, Poland

K-distributed clutter model and the result of performance of the log-t detector in this clutter are presented. Two characteristics were under investigation: false-alarm probability P_{FA} and detection probability P_D. The results for a finite numbers of clutter samples N were obtained by using Monte Carlo simulation. In a case where N tends to infinity, analytical forms for P_{FA} and P_D were given.

1. INTRODUCTION

A new class of non-Rayleigh distributions, the K-distributions has recently been found to provide a good mathematical description of the radar clutter [1]. The probability density function PDF of a clutter envelope sample A_i in this model is

$$p(A_i;b,M) = \frac{2b}{\Gamma(M)} (0.5bA_i)^M K_{M-1}(bA_i) \qquad (1)$$

$$A_i > 0,\ b > 0,\ M > 0$$

where $\Gamma(\ )$ is the gamma function, K_{M-1} is the $(M-1)$th-order modified Bessel function. Parameter b is a scale parameter, associated with the clutter power. Parameter M depends on area illuminated by the electromagnetic wave and is "almost" a shape parameter. Comparison of experimental data with the model (1) gives a reasonable fit for $0 < M < 3$ [2]. The K-distributed clutter model offers the potential to accurately represent the real clutter distribution over a much wider range of conditions than either log-normal or Weibull model [3].
In this paper the detection of target in K-distributed clutter is examined from a statistical detection viewpoint. An important property of an automatic detection system is the ability to maintain a constant false-alarm rate (CFAR). Goldstein [4] has proposed an automatic detector, a so-called log-t detector, which maintains a CFAR in clutter models based on scale and shape parameter family distributions. The test statistic of this detector, based on N independent clutter samples, is given by

$$t = \frac{Z_0 - \frac{1}{N}\sum_{i=1}^{N} Z_i}{\left[\frac{1}{N}\sum_{i=1}^{N}\left(Z_i - \frac{1}{N}\sum_{j=1}^{N} Z_j\right)^2\right]^{0.5}} \qquad (2)$$

where $Z_i = \ln A_i$ and Z_0 is a sample under test. It is seen from eqn. 1 that the test given by eqn. 2 does not depend on the parameter b, but does depend on the value of M.
It is a matter of interest to determine the behaviour of the log-t detector in K-distributed clutter.

2. DETECTOR CHARACTERISTICS

Under investigation were two characteristic: false-alarm probability P_{FA} and detection probability P_D, given by

$$P_{FA} = \text{Prob}\{\, t>T \mid \text{clutter only}\,\} = \int_T^\infty p_c(t)dt \qquad (3)$$

$$P_D = \text{Prob}\{t>T \mid \text{signal+clutter}\} = \int_T^\infty p_{s+c}(t)dt \qquad (4)$$

where $p_c(t)$ denotes the probability density function of the test statistic when only clutter is present, and $p_{s+c}(t)$ denotes the probability density function when signal plus clutter is present. T denotes a prescribed threshold.

When A_i represents a K-distributed clutter sample, then probability density function of Z_i (i.e. clutter sample in output of an ideal log-amplifier) is

$$p(Z_i;b,M) = \frac{(b\ \exp Z_i)^{M+1}}{2^{M-1}\Gamma(M)}\ K_{M-1}\, b\ \exp Z_i \qquad (5)$$

and the corresponding distribution function assumes the form

$$P(Z_i<z) = \int_0^z p(Z_i;b,M)dZ_i = 1 - \frac{(b\ \exp z)^M K_M (b\ \exp z)}{2^{M-1}\Gamma(M)} \qquad (6)$$

The mean value m and the variance σ of Z_i, evaluated by the characteristic function method, are:

$$m = 0.5\,\Psi(M) - 0.5\gamma + \ln\frac{2}{b} \qquad (7)$$

$$\sigma^2 = 0.25\,\zeta(2,M) + \frac{\pi^2}{24} \qquad (8)$$

$\Psi(\ )$ - psi function; γ - Euler constant;
$\zeta(\ ,\ )$ - generalized zeta function

In the general case it is very difficult to obtain an analytical expression for $p_c(t)$ and $p_{s+c}(t)$, when clutter sample Z_i is described by eqn.5. As the number of samples N tends to infinity, the test statistic (2) assumes the form

$$t = \frac{Z_0 - m}{\sigma} \qquad (9)$$

where m and σ are given by eqn.7 and 8.
In this case it is possible to found an analytical form for P_{FA} and P_D.

3. FALSE-ALARM PROBABILITY

Dependence of the false-alarm probability on K-distributed clutter parameters for a log-t detector was evaluated and presented in [5]. When the number of samples N approaches infinity, P_{FA} can be evaluated from (6) and (9), and assumes the form

$$P_{FA}(T,M) = \frac{2}{\Gamma(M)} \exp\left[M\left[T\sigma + 0.5\Psi(M) - 0.5\gamma\right]\right] \times K_M\left[2\exp\left[T\sigma + 0.5\Psi(M) - 0.5\gamma\right]\right] \qquad (10)$$

where σ is given by eqn.8. For T=constant P_{FA} is a function of the parameter M only.
For a finite N a Monte Carlo simulation was employed by means of the CDC 6400 computer. The results of the computer simulation, obtained in [5], are shown in Fig.1. The brocken curve represents P_{FA} as function of parameter M for an infinite value of N, according to eqn.10.

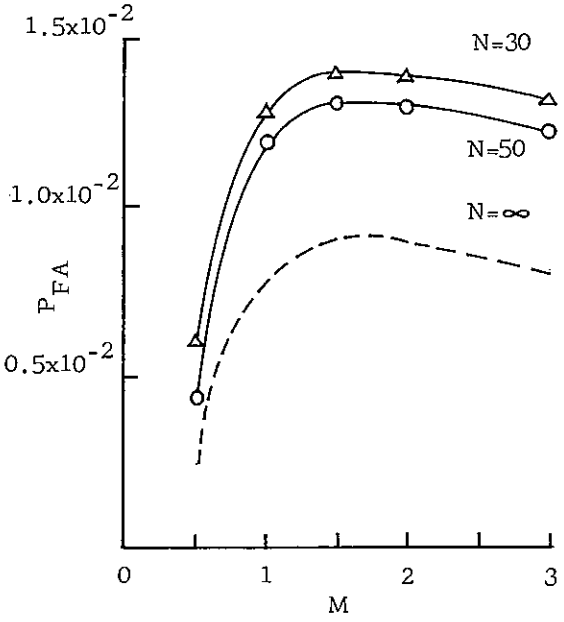


Fig.1 P_{FA} as a function of M for T=1.85

The curves for finite and infinite value of N are similar in shape. There are no considerable differences between curves for N=30 and for N=50.

4. DETECTION PROBABILITY

A detection performance analysis will be presented for fluctuating target model, based on Rayleigh distribution.
When N tends to infinity, the detection results have been obtained using a technique which circumvents the requirement of deriving the probability density function of the signal plus clutter envelope. This technique is based upon the fact that for high signal-to-clutter ratio (SCR) the envelope probability distribution alone [4].
Using this approximation for a Rayleigh target the detection probability P_D is found to be

$$\ln P_d = \frac{-b^2 \exp 2(T\sigma_c + m_c)}{4\lambda m} \qquad (11)$$

where m_c and σ_c are given by eqn.7 and eqn.8 respectively, λ^c is the signal-to-clutter power ratio, defined as $\lambda = E(A_s^2)/E(A_c^2)$. It is seen from eqn.11 that for λ = constant P_D is a function of both clutter parameter: b and M.
For a finite N a computer simulation was employed, similary as the one discussed in Section 3. 10000 trials N=30 were generated on the CDC 6400 computer to simulate independent and identically distributed clutter samples. A target signal was added to the samples under the test. The results are presented in Fig.2 and Fig.3.

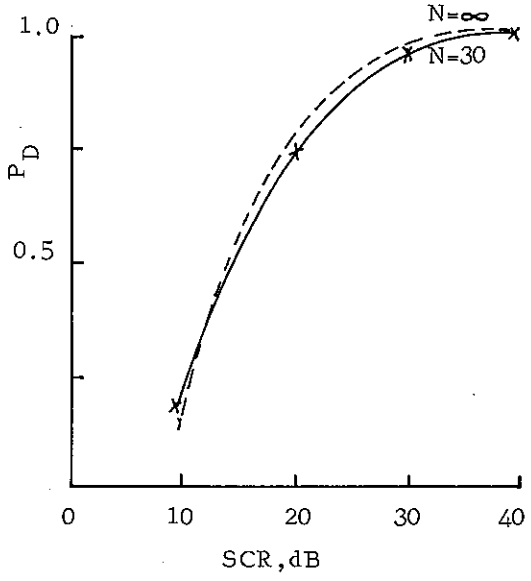


Fig.2 P_D against the SCR for T=1.85; b=1; M=0.5

The brocken curves represent P_D for infinite value of N, according to eqn.11. There are no practical differences between theoretical and experimental curves in presented cases.

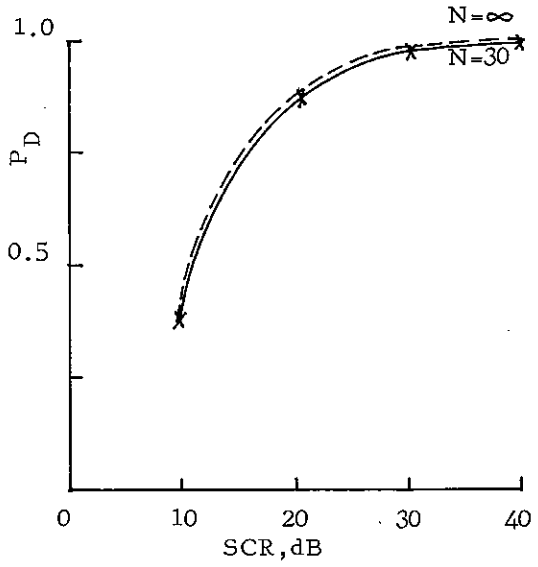


Fig.3 P_D against the SCR for T=1.85; b=1; M=1

5. CONCLUSIONS

It is seen from Fig.1 that the false-alarm probability for the log-t detector in K-distributed clutters depends on the clutter parameter M. However for $1 < M < 3$ the log-t detector almost maintains the constant false-alarm probability $(P_{FA} = 10^{-2}$ for $T = 1.85)$.
The detection probability depends on both clutter parameters and on signal-to-clutter power ratio.
Detection probability for the Rayleigh target can be evaluated from eqn.11 even for finite N, because it is no discernible difference between the results obtained using the approximate technique and those obtained via simulation. It has to be emphasized, that for SCR less than 20 dB detection probability is small. The presented detection reszlts are similar to the results obtained by Goldstein [4] and Trunk [6] for log-normal and Weibull clutter.

REFERENCES

[1] Jakeman, E. and Pusey, P.N., A model for non-Rayleigh sea echo, IEEE Trans., 1976, AP-24, pp. 806-814

[2]Jakeman, E. and Pusey, P.N., Statistic of non-Rayleigh microwave sea echo, Radar-77, IEE Conf. Publ. 155, 1977, pp. 105-109
[3]Jakeman, E., On the statistics of K-distributed noise, J. Phys. A, 1980, Vol. 13, pp. 31-48
[4]Goldstein, G. B., False-alarm regulation in log-normal and Weibull clutter, IEEE Trans., 1973, AES-9, pp. 84-92
[5]Jakubiak, A., False-alarm probabilities for a log-t detector in K-distributed clutter, Electronics Letters, 1983, Vol. 19, pp. 725-726
[6]Trunk, G. V., and George, S. F., IEEE Trans., 1970, AES-6, pp. 620-628

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SUBSETS OF AUTOREGRESSIVE PARAMETERS

Piet M.T. Broersen

Department of Applied Physics,
Delft University of Technology, P.O. Box 5046,
2600 GA Delft, The Netherlands.

A subset model has M lags with nonzero parameters while other intermediate lags are not in the model. Subsets of autoregressive parameters can be selected with a modified Stepwise Directed Search algorithm. This is combined with the Weak Parameter Criterion, which gives an empirical estimate for the prediction error that can be used in the selection of the model order and of a subset. The quality of the selected models is evaluated with the normalized autoregressive transfer function error. This is a measure that can be defined both in the time and in the frequency domain for simulation experiments where the true process is known. It is shown that subsets mostly give more accurate models with less estimated parameters. It is useful to restrict the maximum lag for the subsets to the order selected with the Weak Parameter Criterion.

1. INTRODUCTION

Autoregressive models can be characterized by parameters or by reflection coefficients. Likewise, estimation methods can be divided into methods estimating parameters and methods that produce reflection coefficients. This paper deals with the estimation of parameters by the minimization of the residual power of the squared forward and/or backward residuals.
Generally, all parameters until some selected order are included in a model, and the parameters above that order are assumed to be zero. Suppose that the first M parameters are in this contiguous model, that can be characterized by a certain prediction error. A subset model can contain M parameters that are selected from the first L parameters with some selection strategy; L is the highest order that will be taken into account. All parameters above order L and those below L that are not selected into the subset are considered to be zero. Such subset models will often have a smaller prediction error than the contiguous model. In other words, the best fitting model with M estimated parameters may be a subset model. Also theoretical considerations can lead to subset models, e.g. seasonal models contain a number of intermediate parameters that are zero.

So far, little attention has been given to the problem of subset selection in autoregressive models. McClave [1,2] used search strategies that exist for ordinary linear least squares regression. Also his selection criteria are based on the asymptotical equivalence between ordinary regression and autoregression. But neither the selection strategy nor the selection criterion reflects the special finite sample behaviour of autoregressive estimation. The sum of squared residuals in estimating an increasing number of parameters from small samples of observations depends on the method of estimation. It is different for the minimization of forward residuals only or of the sum of forward plus backward residuals. An order or subset selection criterion has been developed that can be adapted to the estimation method: the Weak Parameter Criterion WPC. Broersen [3] described the WPC for the Burg and the Yule-Walker techniques for the estimation of reflection coefficents. This paper will give the weakness coefficients for the least squares estimation of autoregressive parameters and a definition of the WPC for subsets of parameters.
Another subject of investigation is the selection strategy. A conceptually simple method to select a subset from L possible parameters is based on the evaluation of all 2^L different subset models with their WPC values, after which the subset with the smallest WPC is selected. However, this would require much computing time and 2^L different models is certainly not feasible for L greater than 30. Stepwise Directed Search has been developed [4] as an efficient search strategy for the selection from many candidates for inclusion in the subset. SDS uses the selection criterion to iterate around local minima of the criterion. SDS has been applied to problems in ordinary regression but it can also be used for autoregressive subsets.
This paper describes the modifications of SDS and of WPC for the estimation of subsets of parameters. Subsets of all L lags are compared with subsets below the selected WPC order. The measure for the comparison is a normalized autoregressive transfer function error as described by Parzen [5]. This

research is restricted to stochastic processes. This means that a statistical measure for the comparison of different models is sufficient. Specific problems, like line splitting, may occur in deterministic processes, but these are not considered here.

2. WEAK PARAMETER CRITERION

The Weak Parameter Criterion WPC is an order selection criterion that is adapted to the specific small sample performance of the different autoregressive estimation methods. It is based on weakness coefficients which are empirical approximations for the variance of a parameter if it would be estimated from a white noise sequence. They have been introduced for Burg and Yule-Walker estimates [3]. Some other estimation methods can be called least squares methods because they minimize the sum of squares of forward and/or backward residuals to estimate all parameters simultaneously. The method with unidirectional residuals is denoted LSF. LSFB uses a combination of forward and backward residuals. It is a modification of the Burg method that is described by Marple [6]. Broersen [7] gives formulae for the weakness coefficients of least squares methods. The asymptotical value for the weakness coefficients v_i equals 1/N, independent of the order i and the method of estimation. The empirical approximations are

$$\begin{aligned} v_{i,B} &= 1/(N+1 - i) \\ v_{i,YW} &= (N-i)/\{N(N+2)\} \\ v_{i,LSFB} &= 1/(N+2 - 1.5\,i) \\ v_{i,LSF} &= 1/(N+2 - 2i) \end{aligned} \qquad (1)$$

for Burg, Yule-Walker, LSFB and LSF estimates respectively.
The Weak Parameter Criterion for order m is defined as

$$WPC(m) = S_m^2 \Big/ \prod_{i=0}^{m} (1-2v_i), \qquad (2)$$

with $v_0=0$. S_m^2 is the average residual power of the model of order m, which is calculated as the residual sum of squares divided by the number of contributing terms [7]. In order selection, WPC(m) is calculated for m=0,1,...L, where L is an arbitrarily chosen maximum order. The order M with smallest WPC(M) is selected; this is denoted the WPC order. The definition (2) is modified for subsets of parameters as

$$WPC(sub) = S^2(sub) \Big/ \prod_{i=0}^{M_{sub}} (1-2v_i), \qquad (3)$$

where M_{sub} is the number of parameters in the subset. So the WPC of subsets depends only on the number of parameters and not on the indices of the parameters. This is completely different from subsets of reflection coefficients [8] that belong to the Burg and the Yule-Walker methods of estimation.

The subset selection procedure Stepwise Directed Search has been developed in ordinary linear least squares analysis [4]. It can be adapted to autoregressive problems by using the WPC (sub) as the selection criterion.

3. THE QUALITY OF MODELS

It is important to have a general measure available for the comparison of different classes of models in simulation experiments. The quality criterion should preferably not depend on the residuals of selected models because these have been used in the selection procedures. Broersen[4] has shown that the residuals in ordinary regression give a bias in the assessment of the quality and simulations have shown the same bias in autoregression. It is no problem to suppose that the true system is known for a quality measure, because this will only be used in simulation experiments.
Parzen[5] discussed the equivalence of several distance measures in the frequency domain. He also described approximate relations of those measures with the prediction error in the time domain. We adopt the Normalized Autoregressive Transfer Function Error that is defined as [5]

$$NATFE = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{\hat{A}_p(e^{j\omega}) - A(e^{j\omega})}{A(e^{j\omega})} \right|^2 d\omega \qquad (4)$$

where A(z) is the true autoregressive transfer function and $\hat{A}_p(z)$ is the estimate of order p. A theoretical derivation or a computer calculation shows that the equivalent expression in the time domain is given by

$$NATFE = \frac{\hat{a}_p^T R_{yy} \hat{a}_p}{\sigma_\varepsilon^2} - 1. \qquad (5)$$

The vector $\hat{a}_p^T$ contains the parameter estimates $1, \hat{a}_1, \ldots, \hat{a}_p$ and R_{yy} is the p+1 x p+1 true Toeplitz covariance matrix. The product in the numerator of (5) is an estimate for the prediction error. Hence, (4) and (5) describe one measure in the time and frequency domain; models with a small prediction error give at the same time an accurate spectral approximation and vice versa. A previous paper [8] contains tables with the scaled prediction error. This equals NATFE+1. Advantages of NATFE are the easy scale for figures and the emphasis on the two domains where NATFE is defined, showing the equivalence of good models in time and frequency domain.

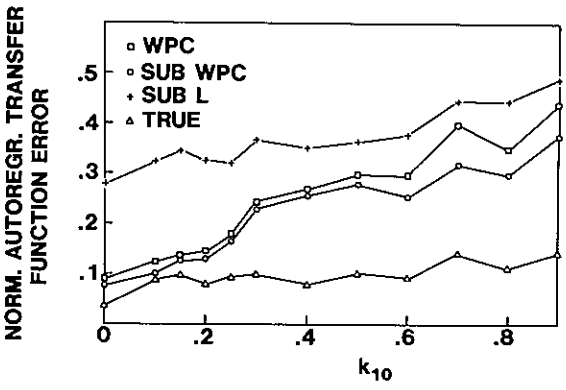


Fig.1. Quality of selected models for a true subset process with k_1, k_2, and k_{10}.

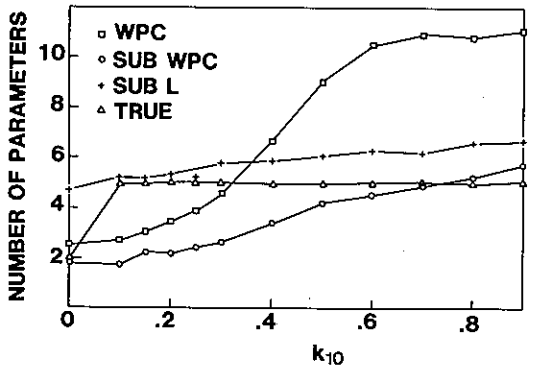


Fig.2. Size of the selected models.

4. RESULTS

The results in this paper have been obtained as averages over 100 simulation runs with computer generated autoregressive processes. The true processes were characterized by a subset of 3 reflection coefficients in figure 1, or a contiguous process with the 10 first reflection coefficients in figure 3. The relation between parameters and reflection coefficients is given by the recursive Levinson algorithm [6]. The subsets with k_1=-0,50, k_2=0.15 and k_{10} variable have five non-zero parameters: the parameters 1,2,8,9, and 10. A process with all reflections present has also 10 non-zero parameters. The length of each realisation was N=64; the maximum considered order for WPC order selection or for free subset selection was L=20; the true number of parameters was 5 in figures 1 and 2 and 10 in figures 3 and 4. The same realisations have been used before in the selection of subsets of reflection coefficients [8]. A total of 31 different models have been selected in each realisation; four of them are given here, obtained with LSFB estimation:

- WPC: the contiguous model with minimum WPC(M)
- SUB WPC: the subset with the selected WPC order M as maximum order
- SUB L: the subset with L as maximum order
- TRUE: the subset or contiguous model with estimated parameters for those which are non-zero in the generating process.

It has been shown previously [7] that WPC selection gives better results than the use of the asymptotical criterion AIC of Akaike [9]. Results with AIC are not reported here because the NATFE was greater than obtained with WPC. For the same reason, results of LSF have not been reported. Unidirectional residuals have a greater weakness coefficient v_1 in (1). This can be interpreted as a greater contribution to the prediction error for each estimated parameter [7], which explains the greater values of NATFE. The results for SUB L deteriorate quickly with an

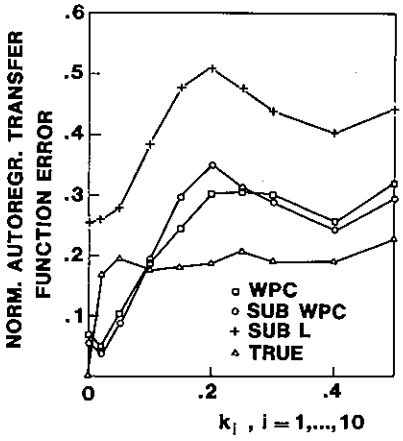


Fig.3. Quality of selected models for a true contiguous process.

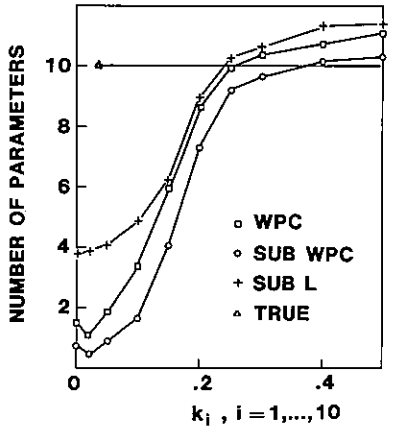


Fig.4. Size of the selected models.

increase of the arbitrarily chosen value of L. The average NATFE of SUB L is almost always greater than the NATFE of WPC or SUB WPC models, unless the chosen value of L coincides with the true process order.Generally, SUB WPC gives better models with less parameters and almost independent of the choice of L. Also the computing time for SUB WPC was 5 to 10 times less than for SUB L. Hence, without a priori knowledge of the true process order, subsets can better be selected below the WPC order than below the arbitrary maximum order L.

A comparison of WPC and SUB WPC gives as a first result that the number of parameters in the subset is smaller, as it should be because SUB WPC can never contain more parameters than the contiguous WPC model for a specific realisation. The NATFE of the SUB WPC models in fig.1 and fig.3 is almost always smaller than the NATFE of the WPC models. The only exception is k_1=0.15 or 0.20 in fig.3; the parameters are of order of magnitude $\sqrt{2}v_1$ here, which is the critical value for significance of individual parameters [3]. Both the WPC an the SUB WPC model have less parameters than the true contiguous process. In such occasions, models with more estimated parameters included give a smaller prediction error or NATFE. The computing time for SUB WPC is about three times longer than for WPC, if SUB WPC is computed with Stepwise Directed Search and the WPC-order is selected with the efficient program for contiguous models of Marple [6].

The measure NATFE has a great practical advantage above other measures in the frequency domain. The estimated parameter vector appears in the numerator of equation (4). But most spectral measures have the estimated parameters in the denominator, which gives a singularity if an estimated pole is located exactly on the unit circle. Moreover, it becomes difficult to consider power spectral densities of systems with poles outside the unit circle, because those systems don't belong to the class of stationary stochastic processes with finite variance. Those problems are evaded with NATFE; both (4) and (5) are perfectly well defined for all parameter estimates, as long as the true process with A(z) is stationary. Also other measures for the quality of models have been considered in the simulations, but all turned out to be less useful than NATFE. Especially the measures that are based on the difference between the estimated and the true values of the parameters or the reflection coefficients had a bad performance in some simulations, because they cannot deal properly with the covariance between estimates. In the context of this paper, it is clear that the true order or the true subset is not the aim of the search and selection procedures, but only a model with a close approximation in time or frequency domain.

5. CONCLUSION

The Normalized Autoregressive Transfer Function Error is a suitable measure for a comparison of estimated models in simulations. It can be calculated in the time and in the frequency domain. It shows that models with an accurate spectral description give at the same time a small prediction error. This measure is not sensitive for the position of estimated poles; they may be inside, on or outside the unit circle in the complex plane; all three situations occur in LSFB estimates.
The combination of a WPC definition for subsets and the strategy Stepwise Directed Search yields good-fitting subset models with less parameters than other estimated models. It is useful to restrict the greatest lag in subsets to the selected WPC order. It makes the selected subset independent of an arbitrarily chosen upper boundary of candidate lags. Moreover, the NATFE of subsets below the WPC order is smaller than the NATFE that would be obtained without this limitation of lags.

SUBSET SELECTION GIVES BETTER MODELS.

REFERENCES

[1] McClave, J.T., Subset Autoregression. Technometrics 17, (1975), 213-220.
[2] McClave, J.T., Estimating the Order of Autoregressive Models: The Max Chi-squared Method. J.Am. Statist. Assoc. 73, (1978), 122-128.
[3] Broersen, P.M.T., Selecting the Order of Autoregressive Models from Small Samples. IEEE Trans. Acoust., Speech, Signal Processing ASSP-33, (1985), 874-879.
[4] Broersen, P.M.T., Subset Regression with Stepwise Directed Search. Appl. Stat., 35, no. 2. (1986).
[5] Parzen, E., Some Recent Advances in Time Series Modeling. IEEE Trans. Automat. Contr. AC-19, (1974), 723-730.
[6] Marple, L., A New Autoregressive Spectrum Analysis Algorithm. IEEE Trans. Acoust., Speech, Signal Processing ASSP-28, (1980), 441-454.
[7] Broersen, P.M.T., The Predicition Error and the Weak Parameter Criterion in Autoregressive Estimation (1986), submitted for publication.
[8] Broersen, P.M.T., Subsets of Reflection Coefficients. Preprints ICASSP Conference, (1986), Tokyo.
[9] Akaike, H., A New Look at the Statistical Model Identification. IEEE Trans. Automat. Contr. AC-19, (1974), 716-723.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N2.2

KNOWN INPUT POWER SPECTRUM IN ADAPTIVE L.M.S. AND A.G. ALGORITHMS

Gregori Vázquez, Antoni Gasull and Miguel A. Lagunas.

E.T.S. Ingenieros de Telecomunicación. U.P.C.
C/ Jordi Girona Salgado, s/n. 08034 Barcelona - Spain.

ABSTRACT. This work deals with the use of previous or colateral information to improve the behaviour of adaptive algorithms. The study is made on gradient-based methods due to the relatively simple and good performances that they use to exhibit.

This paper shows that the complete knowledge of the data at the input of the adaptive filter (and in consequence of its autocorrelation matrix and its inverse) can be used to modify the classic L.M.S. algorithm leading to new expressions for the gradient and for the optimum 'step size', alternative, in some cases, to the Powell expression.

Finally, the description is completed with the comparison between the variation ranges and VLSI implementation cost for this two optimum 'step size' values and a natural generalization set of parameter is obtained.

1. INTRODUCTION

For the sake of simplicity, let us focuss the classic problem of Wiener filtering. Two possible alternatives can be adopted. The first one is the direct use of the optimum Wiener equation and the other is that an adaptive approach could be better under actual situations, where finite arithmetics and no-stationary conditions are used to be imposed.

If we adopt the second possibility, the question is how to use all the previous or colateral information available in a given adaptive algorithm.

From our point of view, there would be two possible choices to reflect these additional information in an adaptive squeme with a quadratic objective. They are the following:

a) To use it to estimate better the parameters or associated functions involved in the adaptive algorithm.

b) To include the colateral information as constrains or just in the minimization process.

This work is driven in both senses. The first one is the most obvious and will be used only to improve the gradient estimate.

On the other hand, the second one is not so direct as the previous, and, in general, it will try to modify the whole structure into the adaptive scheme to satisfy the constrains or the pursued error minimization criterion.

Thus, althrough it seems an atractive possibility, the designer will have to pay attention because, as a matter of fact, often the structure obtained will need a very intensive computation. In our case, only an optimum value for the step size will be searched, keeping the usual adaptive scheme.

2. REVIEW OF THE MINIMUM M.S.E. LINEAR FILTERING: [1], [2], [3], [4].

Let's consider the general scheme given in the figure 1. The objective is to minimize the mean square error (m.s.e.) between a reference signal $y(n)$ and an estimate of this signal at the output of a Q order F.I.R. filter defined by the coefficient vector $\underline{W}$. Thus, given an input data signal $x(n)$, we dispose of a data vector $\underline{X}_n$ and the desired estimation:

$$\hat{y}(n) = \underline{X}_n^T.\underline{W} = \underline{W}^T.\underline{X}_n \qquad (2.1)$$

where:

$$\underline{X}_n^T = (x(n),x(n-1),\ldots,x(n-Q+1))$$

$$\underline{W}^T = (w(0),w(1),\ldots,w(Q-1))$$

The weight vector will be chosen so that the M.S.E. is minimum:

$$\varepsilon^2 = E((y(n)-\hat{y}(n))^2)=E((y(n)-\underline{X}_n^T.\underline{W})^2) \qquad (2.2)$$

and developing the expression:

$$\varepsilon^2 = E(y^2(n))-E(y(n)\underline{X}_n^T)\underline{W} - \\ - \underline{W}^T E(y(n)\underline{X}_n)+\underline{W}^T R_{xx}\underline{W} \qquad (2.3)$$

The minimization of this expression leads to the well known optimal Wiener solution:

$$\underline{W}_{opt} = R_{xx}^{-1}.\underline{P} \qquad (2.4)$$

Where $R_{xx}=E(\underline{X}_n.\underline{X}_n^T)$ is the QxQ autocorrelation matrix of the data sequence x(n) and $\underline{P}=E(y(n).\underline{X}_n)$ is a cross-correlation vector between the data vectors $\underline{X}_n$ and the reference samples y(n) .

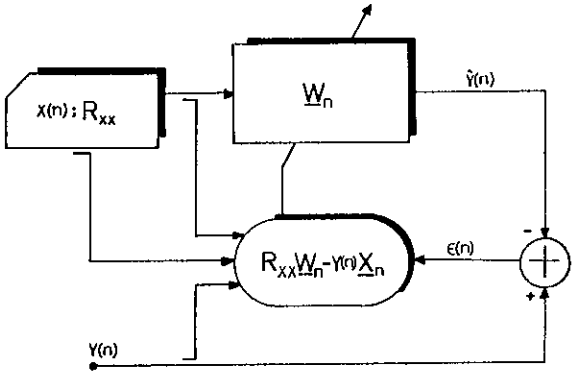


Figure 1. General diagram.

In general, without additional information about R_{xx} and $\underline{P}$, the designer has to use estimates of both R_{xx} and $\underline{P}$. The most familiar approach is the Steepest Descent Method based in the gradient of the error given in (1.2):

$$\underline{W}_{n+1} = \underline{W}_n - \mu\underline{\nabla}_n \qquad (2.5)$$

with $\underline{\nabla}_n = R_{xx}\underline{W}_n - \underline{P}$ the exact gradient vector of the M.S.E. and μ the step size.

But, the knowledge about R_{xx} and $\underline{P}$ is partial and thus, the exact gradient is substituted by an estimate ($\hat{\underline{\nabla}}_n$):

$$\underline{W}_{n+1} = \underline{W}_n - \mu\hat{\underline{\nabla}}_n \qquad (2.6)$$

Widrow proposed as gradient $\hat{\underline{\nabla}}_n = e(n).\underline{X}_n$ (i.e. the so-called instantaneous gradient) [3], and L.J. Griffiths to $y(n)\underline{X}_n - \underline{P}$ [2] in an adaptive array context. As it is seen, the Widrow approach is a complete instantaneous estimate and it presents a very computation simplicity. For this reasons, this proposal will be used in one of our final algorithms.

3. DESCRIPTION OF THE METHOD:

In some cases, the data sequence {x(n)} is exactly known (i.e. active sonar, time delay estimation, ...), and in consequence, its autocorrelation matrix and its inverse matrix can be considered as data. Thus, under this condition, it seems that the best alternative is directly the Wiener equation for the filter coefficient estimation, that is:

$$\underline{W}_n = R_{xx}^{-1}.\hat{\underline{P}}_n \qquad (3.1)$$

where: $\hat{\underline{P}}_n$ is a vector cross-correlation estimate, for instance, an instantaneous one:

$$\hat{\underline{P}}_n = y(n).\underline{X}_n \qquad (3.2)$$

but it is clear that this method doesn't make enough use of the past information. Among many alternatives, a possible one could be the following smoothed version:

$$\underline{W}_n = \alpha.\underline{W}_{n-1} + (1-\alpha)R_{xx}^{-1}.\hat{\underline{P}}_n \qquad (3.3)$$

where α is a constant such that $0<\alpha<1$. However, in actual situations, the evaluation of the expressions will be made with finite arithmetics and between them, basically, in fixed point. On the other hand, the election of parameter α is a difficult issue because it will affect strongly to the convergence speed and it is not clear.

Thus, even in this case, it seems to be reasonable the election of adaptive expressions based on the gradient of the mean square error.

Under the conditions of the problem, our proposal for the gradient estimate is:

$$\hat{\underline{\nabla}}_n = R_{xx}.\underline{W}_n - y(n).\underline{X}_n \qquad (3.4)$$

instead of the reported by Widrow and Griffiths in other contexts. If the prior knowledge of R_{xx} is not complete, an adaptive actualitation could be made by the successfully employed expression:

$$\hat{R}_{xx}(n) = (1-\beta)\hat{R}_{xx}(n-1) + \beta\underline{X}_n^*\underline{X}_n^T;\ 0<\beta<1 \qquad (3.5)$$

Where: * denotes complex conjugate, expression that has been verified, recently, into the Window Methods of Spectral Estimation [5], [6], giving a good physical sense to its ordinary use.

At this moment, it is necessary to describe the 'step size' (μ) actualitation in each iteration. As it is well known, the election of this parameter will affect to the convergence time, the stability and many other parameters of the algorithm. In this sense, for the L.M.S. method, it is shown that a sufficient condition to guarantee stability is that $0<\mu<1/\lambda_{max}$, where λ_{max} is the maximum eigenvalue of R_{xx}.

Another strategy is the so-named Accelerated Gradient (A.G.) approachs. In this way, Powell argued that a suitable election of μ is such that the M.S.E. (2.3) is minimized in each iteration. The obtained expression is:

$$\mu_n = \frac{\nabla_n^T \nabla_n}{\nabla_n^T R_{xx} \nabla_n} \qquad (3.6)$$

Thus, the optimum is just the invers of the Rayleigh quotient, which it is ensured to be bounded by the inverses of the maximum and minimum eigenvalues of R_{xx}:

$$\frac{1}{\lambda_{max}} < \mu_n < \frac{1}{\lambda_{min}} \qquad (3.7)$$

Our proposal differs in the minimization objective. Assuming R_{xx} exactly known, we consider that the best error to be minimized is the M.S.E. associated to the weights, given by:

$$\varepsilon'^2 = E((\underline{W}-\underline{W}_{opt})^T(\underline{W}-\underline{W}_{opt})) \qquad (3.8)$$

For this new objective, the optimum 'step size' (μ_n') is found to be:

$$\mu_n' = \frac{\nabla_n^T R_{xx}^{-1} \nabla_n}{\nabla_n^T \nabla_n} \qquad (3.9)$$

relation that can be expressed in many other equivalent forms through the use of the various gradient estimates.

The analysis of the new quotient (3.9) shows that its variation range is bounded by the same values (3.7) that for Powell's relation (3.6). As it is seen, this optimum needs the knowledge of the inverse of R_{xx}, and thus, it will be a usefull relation only in the problem under study.

The gradient in (3.9) can be evaluated by the proposed relation (3.4), but for the sake of simplicity, other possibility is the use of the Wiener's gradient, leading to an equivalent approach such that:

$$\mu_n' = \frac{\underline{X}_n^T R_{xx}^{-1} \underline{X}_n}{\underline{X}_n^T \underline{X}_n} \qquad (3.10)$$

that represent the most simplified estimate.

The comparison between the Powell's step size (3.6) and the new quotient suggests the following natural generalitation:

$$\mu_n^K = \frac{\nabla_n^T R_{xx}^K \nabla_n}{\nabla_n^T R_{xx}^{K+1} \nabla_n} \qquad (3.11)$$

This complete set of possible values are shown to satisfy the bound conditions (3.7) for each K, too, and particularly leads to (3.6) and (3.9) for K=0 and K=-1, respectively.

Again for (3.11) any approach could be adopted for the gradient, suplying different estimates of (3.11) like in (3.10).

Under our approach, the gradient noise, coefficients error, missadjustment error and convergence rate have shown a better performance with respect other algorithms. Besides, the A.G. propossal presents the same computation cost as for Powell's one, and the same structure.

Finally, a complete study of (3.11) for different K-values will be presented in further papers.

4. ALGORITHMS AND IMPLEMENTATIONS

As it has been described, the main proposed scheme consists in the iterative evaluation of (2.6), where the gradient is given by (3.4) and the step size by (3.9).

It is known that the A.G. Methods present a hard computation effort, nevertheless, using VLSI techniques, it is not a trouble. Keeping it in mind, our scheme can be modified to achieve maximum simplicity. For intance, the gradient (3.4) could be substituted by the simpler one given by Widrow, leading to relation (3.10). The analysis of this way shows that many systolic realizations can be adopted, and among them, the CORDIC one seems to be the best under low sampling rate assumption.

One possible implementation is shown in fig. 2, making use of only two of the six basic CORDIC elements. Note in fig. 2, that normalizations needed to ensure convergence in CORDIC elements can be applied jointly to the reference signal y(n). So, it is just needed a constant factor, and it yields a very compact realization.

The network to evaluate the value (3.10), that is not in fig. 2, will only requiere a normalized matrix-vector and vector-vector product and no-particular comment will be necessary.

5. CONCLUSIONS

This paper has considered the use of prior information into adaptive algorithms. It has shown that the complete knowledge of the input data autocorrelation matrix or power spectrum density function can be used to improve the gradient estimate and defining a better criterion that Powells' to optimize the step size in each iteration. As a consecuence of the found relation a generalization of the step size computation has been presented. The proposal has shown an improvement in the main parameters of the adaptive system.

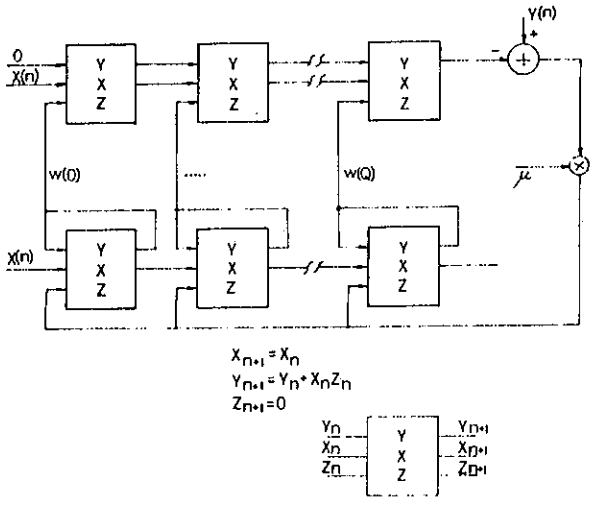


Figure 2. Systolic Implementation.

REFERENCES

|1| R.A. Monzingo and T.W. Miller. "Gradient-Based Algorithms". John Wiley & Sons, Inc. 1980.

|2| L.J. Griffiths. "A Simple Adaptive Algorithm for Real-Time Processing in Antenna Arrays". Proc. of the IEEE, Vol. 57, No. 10. Oct. 1969.

|3| B. Widrow et al. "Adaptive Antenna Systems". Proc. IEEE, Vol. 55, pp.2143-2159. Dec. 1967.

|4| B. Widrow et al. "A Comparison of Adaptive Algorithms Based on the Methods of Steepest Descent and Random Search". IEEE Trans. Antennas and Prop., Vol. AP-24, No. 5, Sep. 1976.

|5| M. Bertran & C. Nadeu. "On the Inclusion of Prior Information in the Window Method of Spectral Estimation". Proc. of MELECON'85/Vol. II, pp.59-61. Oct. 1985. Madrid.

|6| R.A. Monzingo and T.W. Miller. "Recursive Methods for Adaptive Array Processing". John Wiley & Sons, Inc. 1980.

|7| J.S. Walther. "A Unified Algorithm for Elementary Functions". 1971, Spring JCC, pp. 379-385.

|8| H.M. Ahmed et al. "A VLSI Speech Analysis Chip Set Based on Square-Root Normalized Ladder Forms". ICASSP 1981, pp.648-653.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N2.4

LEAST-SQUARES RECURSIVE SEQUENTIAL DETECTION OF A SIGNAL WITH UNKNOWN POWER

Pierre-Yves ARQUÈS

G.E.R.D.S.M., D.C.A.N. TOULON,
Le Brusc, 83140 SIX-FOURS-LES-PLAGES (France).

We present an extension of the detection model with observation distance criterion to the sequential case. A particular case is the weighted least squares sequential test. The later is emphasized in the case of the detection of a vectorial signal with unknown amplitude ; this test can be put under recursive form.

1. INTRODUCTION

The "static" detection problem deals with a threshold receiver in which the only decision is taken at a single instant. This decision is binary, "signal present in the observation" or "noise only". In opposition, the sequential detection procedure has a "dynamic" idea. The final decision is taken as soon as possible, according to its performance, on a temporal decision set on which the observer can acquire more observation samples, and thus more information [1].

Usual models of sequential detection can be obtained either by non-structured optimization methods (Bayes, Wald, maximum a posteriori probability criteria), or by structured processings. These models lead to the following test structure : at each decision instant, the likelihood ratio, calculated from the observation at this instant, is compared with two thresholds. Depending on the chosen method, these thresholds are constant or time varying. Under a hypothesis of independence, the observation likelihood ratio can be written in recursive form. These models require a rich a priori information and can be enlarged in some cases where the signal depends on unknown parametres. In this last case, the recursivity of the structure is not necessarily preserved.

The detection model with observation distance criterion [1,2] can be extended to the sequential case in a more accurate manner than in a previous work [3]. Noting that the least-squares are a particular distance criterion, we address the problem of detection of a vectorial signal with unknown amplitude, using discrete-time observations. No particular assumption on the noise probability density function is made, so it can be non-Gaussian : the structure we obtain requires only fairly poor information.

More important is the fact that this weighted least-squares test can be put under recursive form : the receiver output at time t_k is obtained from the output at time t_{k-1} and from the observation at time t_k .

Considering the discrete case, we note t_k , $k \in \mathbb{N}^+$, the observation and decision instants. At t_k , the available observation $x_k = \{ v_1 , v_2 , \ldots , v_k \}$ is a realization of a random $n_e k$ - variable X_k , derived from a random n_e -vectorial process V_k (v_k is a realisation of V_k). The observation, in a time independent manner, corresponds either to noise alone (H_o hypothesis) or to signal and noise (H_1 hypothesis). At every t_k , the choice is between the decisions $\delta_o(t_k)$ "choice of H_o at t_k" , $\delta_1(t_k)$ "choice of H_1 at t_k" , $\delta_d(t_k)$ "acquisition of an observation sample at t_k".

2. SEQUENTIAL TEST WITH DISTANCE CRITERION

We suppose simple hypothesis H_o and H_1 .At every t_k , we consider two distances d_{ok} and d_{1k} in the space $\mathcal{X}(t_k)$ of the observations x_k ,and two references $\bar{x}_{ok}$ and $\bar{x}_{1k}$ of the observation x_k ($\bar{x}_{ok} \in \mathcal{X}(t_k)$, $\bar{x}_{1k} \in \mathcal{X}(t_k)$). At every x_k ,we associate the two numerical values $d_{ok}(x_k , \bar{x}_{ok})$ and $d_{1k}(x_k , \bar{x}_{1k})$, representing the distances from x_k to its reference for each possible hypothesis.

The sequential test can be conceive in several manners, in fitting the situation by means of additive weightings and choice of distances and references. For example we can differ the final decision as long as the values of the distances are neighbouring or/and the observation is too far from one of the references.

A general form of the sequential test with observation distance criterion is (a being an exponent and $\varkappa$, $\varkappa'$, $\varkappa''$ being additive weights):

$$
\left\lfloor
\begin{array}{l}
\forall k \in \mathbb{N}^+ , \\
\bullet \ \delta_o(t_k) : \\
d^a_{ok}(x_k, \bar{x}_{ok}) + \gamma_{ok} < \\
\quad \inf(d^a_{1k}(x_k, \bar{x}_{1k}) + \gamma''_{1k}, \gamma'_{ok}) ; \\
\bullet \ \delta_1(t_k) : \\
d^a_{1k}(x_k, \bar{x}_{1k}) + \gamma_{1k} < \\
\quad \inf(d^a_{ok}(x_k, \bar{x}_{ok}) + \gamma''_{ok}, \gamma'_{1k}) ; \\
\bullet \ \delta_d(t_k) : \text{otherwise} .
\end{array}
\right. \quad (1)
$$

We suppose a simple hypothesis H_o and, with a signal depending of a parameter θ, a composite hypothesis H_1 ; the latter is formed by several simple hypothesis H_θ. We impose to come back to the case of a simple hypothesis $H_{\hat{\theta}}$, by estimation of θ, conditionaly to the signal presence, from the observation x_k ; we use for that a special distance criterion estimator ([1] p.91) :

$$
(2) \quad \hat{\theta}_k(x_k) : \forall x_k , \ \inf_\theta \{ d_{\theta k}(x_k, \bar{x}_{\theta k}) \} ,
$$

where $d_{\theta k}$ and $\bar{x}_{\theta k}$ are the distance and the x_k-reference connected to the hypothesis H_θ. Finally we use the distance $d_{\hat{\theta} k}$ in place of the distance d_{1k}. Thus the sequential test is given by (1) in which $d_{\hat{\theta} k}(x_k, \bar{x}_{\hat{\theta} k})$ is substituted to $d_{1k}(x_k, \bar{x}_{1k})$.

3. WEIGHTED LEAST SQUARES SEQUENTIAL TEST

We suppose that the V_k, X_k, $\bar{x}_{jk}$ are column matrices. Calling A^T the transpose of A, we have :

$$
X_k^T = \left[V_1^T \,|\, V_2^T \,|\, \ldots \,|\, V_k^T \right] .
$$

The sequential test with distance criterion becomes a weighted least-squares sequential test under the four following conditions :

- we use distances which are derived from euclidian norms, weighted by symmetric and (strictly) positive definite ($n_e k$, $n_e k$) - matrices G_{ok}, G_{1k}, $G_{\theta k}$;
- we choice a = 2 ;
- we choice the first conditionnal (to hypothesis) observation moments as observation references ;
- we choice the inverses of the conditionnal (to hypothesis) zero-mean observation covariance matrices as weighting matrices. The test is derived from (1) in using :

$$
\left\lfloor
\begin{array}{l}
\forall j \in \{0, 1, \theta\} , \\
d^a_{jk} = (x_k^T - \bar{x}_{jk}^T) \, G_{jk} \, (x_k - \bar{x}_{jk}) , \\
\bar{x}_{jk} = E_{V|H_j}\{X_k\} , \\
G_{jk} = (E_{V|H_j}\{X_k X_k^T\} - E_{V|H_j}\{X_k\} E_{V|H_j}\{X_k^T\})^{-1} .
\end{array}
\right. \quad (3)
$$

4. SEQUENTIAL WLS-TEST FOR A SIGNAL WITH UNKNOWN POWER

We restrict to the sequential weighted least-squares test in the case of the detection, in the n_e - vectorial noise B_k, of the n_e - vectorial signal S_k of which the vectorial amplitude is unknown ; we look for a recursive form. The signal is :

$$
S_k = A_{\mathcal{D}} \, Z_k = Z_{\mathcal{D}k} \, A ;
$$

and we suppose that :

- the signal and the noise are additive ;
- the first noise moment $E\{B_k\}$ is known;
- the n_e - vectorial signal Z_k is or a known determinist one or a random one of known first moment $E\{Z_k\}$;
- the n_e - vectorial amplitude A is constant and unknown ; its elements are non-negative and are or identical ($A = a I_c$, $I_c = [1, .., 1]$) or not identical ($A^T = [a_1, .., a_{n_e}]$) ;
- Z_k and A being column matrices, $Z_{\mathcal{D}k}$ and $A_{\mathcal{D}}$ are diagonal (n_e, n_e) - matrices of which elements (i,i) are elements i of Z_k and A.

The input vectorial process is :

- under H_o :

$$
(4) \quad V_k = B_k ,
$$

with

. mean value $E\{V_k\} = E\{B_k\}$,

. centered covariance matrix :

$$
C_{o,kj} = E\{V_k V_j^T\} - E\{B_k\} E\{B_j^T\} .
$$

- under H_1 :

$$
(5) \quad V_k = B_k + A_{\mathcal{D}} \, Z_k ,
$$

with

. conditional (to the amplitude) mean value :

$$
E_{V|A}\{V_k\} = E\{B_k\} + E\{Z_{\mathcal{D}k}\} \, A ,
$$

. conditional (to the amplitude) centered covariance matrix :

$$
C_{1|A,kj} = E\{V_k V_j^T\} - (E\{B_k\} + E\{Z_{\mathcal{D}k}\} A) \, (E\{B_k^T\} + A^T E\{Z_{\mathcal{D}k}\}) .
$$

(In the determinist case, $E\{Z_k\} = Z_k$, $E\{Z_{\mathcal{D}k}\} = Z_{\mathcal{D}k}$).

We take as observation references :

$$
\left\lfloor
\begin{array}{l}
\bar{x}_{ok} = \{ E\{B_i\}, \ i \in \{1, \ldots, k\} \} , \\
\bar{x}_{\hat{A}k} = \{ E\{B_i\} + E\{Z_{\mathcal{D}i}\} \, \hat{A}_k , \\
\qquad\qquad i \in \{1, .., k\} \} ;
\end{array}
\right. \quad (6)
$$

the reference $\bar{x}_{\hat{A}k}$ use the weighted least - squares estimation $\hat{A}_k$ (made in t_k from x_k) of the n_e - parameter A ; the weighting $G_{\hat{A}k}$ used for the estimation is also used for the detection in t_k (under H_1). We have or $\hat{A}_k = \hat{a}_k I_c$ or $\hat{A}_k^T = [\hat{a}_{1k}, \ldots, \hat{a}_{n_e k}]$; we

call $\hat{A}_{Dk}$ the diagonal matrix derived from $\hat{A}_k$.

We build, $\forall\, j \in \{0, \hat{A}\}$, diagonal partitioned (kn_e , kn_e) - matrices G_{jk}, with the (n_e, n_e) - submatrices $g_{j1}, \ldots, g_{jk}$ on the main diagonal and null submatrices elsewhere. We choice :

. for a known second order :

$$(7)\quad \begin{cases} g_{oi} = (E\{B_i B_i^T\} - E\{B_i\} E\{B_i^T\})^{-1}, \\ g_{\hat{A}i} = (E\{(B_i + \hat{A}_{D\,i-1} Z_i)(B_i^T + Z_i^T \hat{A}_{D\,i-1})\} \\ \quad -(E\{B_i\} + \hat{A}_{D\,i-1} E\{Z_i\})(E\{B_i^T\} + E\{Z_i^T \hat{A}_{D\,i-1}\}))^{-1}. \end{cases}$$

. for an unknown second order, with stationnarity of centered covariances :

$$(8)\quad \begin{cases} g_{oi} = (\frac{1}{i} \sum_{j=1}^{i} (V_j V_j^T - E\{B_j\} E\{B_j^T\}))^{-1}, \\ g_{\hat{A}i} = (\frac{1}{i} \sum_{j=1}^{i} (V_j V_j^T - (E\{B_j\} + \hat{A}_{D\,j-1} E\{Z_j\}) \\ \quad (E\{B_j^T\} + E\{Z_j^T\} \hat{A}_{D\,j-1}))^{-1}. \end{cases}$$

The estimation $\hat{A}_k$ of A in t_k is such as ([2] eq.(4 - 35 a') and (8 - 34)) :

$$(9)\quad \begin{cases} \hat{A}_k : \inf_A \{ \sum_{i=1}^{k} (V_i^T - E_{V|A}\{V_i^T\}) g_{\hat{A}i} (V_i - E_{V|A}\{V_i\}) \}, \\ \hat{A}_k : \sup_A \{ \sum_{i=1}^{k} E_{V|A}\{V_i^T\} g_{\hat{A}i} (2V_i - E_{V|A}\{V_i\}) \}, \\ \hat{A}_k : \sup_A \{ \sum_{i=1}^{k} A^T Z_{Di} g_{\hat{A}i} (2V_i - 2E\{B_i\} - Z_{Di} A) \}. \end{cases}$$

It can be written :

$$(10)\quad \begin{cases} d^a_{ok} = \sum_{i=1}^{k} (V_i^T - E\{B_i^T\}) g_{oi} (V_i - E\{B_i\}) \\ \qquad = d^a_{oBk}, \\ d^a_{1k} = \sum_{i=1}^{k} (V_i^T - E\{B_i^T\} - \hat{A}_k^T Z_{Di}) g_{\hat{A}i} \\ \quad (V_i - E\{B_i\} - Z_{Di} \hat{A}_k) \\ \qquad = d^a_{1Bk} + d^a_{1Zk}, \end{cases}$$

with :

$$(11)\quad \begin{cases} d^a_{1Bk} = \sum_{i=1}^{k} (V_i^T - E\{B_i^T\}) g_{\hat{A}i} (V_i - E\{B_i\}), \\ d^a_{1Zk} = -2 \sum_{i=1}^{k} \hat{A}_k^T Z_{Di} g_{\hat{A}i} (V_i - E\{B_i\} \\ \qquad - \frac{1}{2} Z_{Di} \hat{A}_k) \\ = -2\hat{A}_k^T \sum_{i=1}^{k} Z_{Di} g_{\hat{A}i} (V_i - E\{B_i\}) \\ \quad + \hat{A}_k^T (\sum_{i=1}^{k} Z_{Di} g_{\hat{A}i} Z_{Di}) \hat{A}_k. \end{cases}$$

5. RECURSIVE FORM OF THE SEQUENTIAL WLS-TEST

The previous sequential weighted least-squares test can be written in recursive form in putting the d^a_{jk} in recursive form and using the recursive least-squares estimator for $\hat{A}$. It derives from (1) by :

$$(12)\quad \begin{cases} d^a_{ok} = d^a_{oBk}, \quad d^a_{1k} = d^a_{1Bk} + d^a_{1Zk}, \\ d^a_{jBk} = d^a_{jB\,k-1} + (V_k^T - E\{B_k^T\}) g_{jk} \\ \quad (V_k - E\{B_k\}), \ \forall\, j \in \{0, \hat{A}\}, \\ d^a_{1Zk} = -2\hat{A}_k^T W_k + \hat{A}_k^T F_k \hat{A}_k ; \end{cases}$$

with :

$$(13)\quad \begin{cases} d^a_{jB0} = 0, \ \forall\, j \in \{0, \hat{A}\}, \\ W_k = W_{k-1} + Z_{Dk} g_{\hat{A}k} (V_k - E\{B_k\}), \\ W_0 = 0, \\ F_k = F_{k-1} + Z_{Dk} g_{\hat{A}k} Z_{Dk}, \\ F_0 = 0 ; \end{cases}$$

with ([2] eq. (6 - 49)) :

$$(14)\quad \begin{cases} \hat{A}_k = \hat{A}_{k-1} + H_k (V_k - Z_{Dk} \hat{A}_{k-1} - E\{B_k\}), \\ H_k = P_{k-1} Z_{Dk} (Z_{Dk} P_{k-1} Z_{Dk} + g^{-1}_{\hat{A}k})^{-1}, \\ P_k = (P^{-1}_{k-1} + Z_{Dk} g_{\hat{A}k} Z_{Dk})^{-1}, \\ P_o^{-1} = 0 ; \end{cases}$$

and with g_{oi} and $g_{\hat{A}i}$ given by (7) or (for unknown second order), in place of (8), by :

$$(15)\quad \begin{cases} g_{oi} = (\frac{i-1}{i} g^{-1}_{o\,i-1} + \frac{1}{i}(V_i V_i^T - E\{B_i\} E\{B_i^T\}))^{-1}, \\ g_{\hat{A}i} = (\frac{i-1}{i} g^{-1}_{\hat{A}\,i-1} + \frac{1}{i}(V_i V_i^T - \\ (E\{B_i\} + \hat{A}_{D\,i-1} E\{Z_i\}) \\ (E\{B_i^T\} + E\{Z_i^T\} \hat{A}_{D\,i-1}))^{-1}. \end{cases}$$

REFERENCES

[1] P.Y. ARQUÈS : Application en détection d'un critère de minimalisation de distance d'observation .Actes du Colloque National sur le Traitement du Signal et ses Applications, NICE, Juin 1975 (GRETSI), pp.447-453.

[2] P.Y. ARQUÈS : Décisions en traitement du signal, Masson, PARIS, 1979, 1982.

[3] P.Y. ARQUÈS : Unitary Introduction of Concepts and Methods in the Detection of Signals in Noise. Signal Processing : Theories and Applications (EUSIPCO 80, First European Signal Processing Conference, LAUSANNE, Septembre 1980). North Holland Publishing Company, 1980, pp.309-314.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N2.5

JOINT ESTIMATION OF CLOSE DELAYS AND APPLICATION TO UNDERWATER ACOUSTICS

M. A. PALLAS and G. JOURDAIN

CEPHAG, INPG/IEG, UA346, BP46, 38402 ST MARTIN D'HERES CEDEX, FRANCE

Using estimation theory, joint maximum likelihood delay estimation, occuring for example in two-path active propagation, is developed. Optimal structure and estimate performances are both defined. Two particular cases of decorrelated estimates are studied. These results are then applied to a real data underwater acoustics experiment.

1. INTRODUCTION

Communication performances are strongly dependent on the propagation channel characteristics. Whatever the area to be studied may be, the first step is to consider the propagation channel and to get some precise knowledge of it. Supposing the channel to be a linear deterministic filter leads to achieve filter identification and to design, for instance, impulse response estimation. This step is referred to as a non parametric method. In the case where some modelisation of the channel, using a set of parameters, is available, the system identification becomes parametric and comes down to model parameter estimation. [1]

In underwater acoustics, and also in various other physical channels, sound emitted at one point propagates along separate paths, depending on the sound celerity profile (sound celerity versus depth) and the emitter depth. Each propagation path may be regarded as a linear filter consisting, in the simplest modelisation case, of an attenuation and a delay term. And the whole propagation filter may be considered as the sum of these elementary linear filters. We are interested here in estimating the *set of delay parameters*.

In the first part of this paper, we survey the principles and classical results of one-delay estimation, considering both passive and active proceedings. Afterwards, we will deal only with active procedures. In a second part, we consider a two-path propagation channel and study the two-delay estimation case, establishing the structure of the optimal receiver and its performances, pointing out some border-line cases. The third part is devoted to a real data experiment in the sea, which involves both separate and close paths. We discuss the applicetion of different kinds of methods on these date in order to study the propagation delays.

2. TIME DELAY ESTIMATION. ACTIVE AND PASSIVE APPROACHES. [2-1]

2.1. Active procedure

In this case, the emitted signal is a real deterministic, completely known signal s(t). For a single path propagation, we modelise the received signal: $r(t)=\alpha\, s(t-\tau) + b(t)$, where b(t) is supposed to be an additive white gaussian noise of power spectral density (psd) γ_0, α is the attenuation, and τ is the delay to be estimated. We only recall here the main results of classical parameter estimation theory. The maximum likelihood (ML) delay estimate $\hat{\tau}$ is given by:

$$\int r(t)\,\frac{\partial s(t-\tau)}{\partial \tau}\,dt\,\Big|_{\tau=\hat{\tau}} = 0 \qquad (1)$$

leading to the optimal structure of figure 1, also consisting of the channel input-output intercorrelation and a maximum detection. This estimator is asymptotically unbiased.

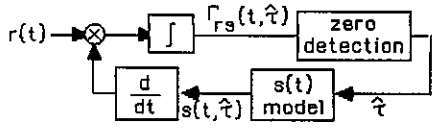


Figure 1

We use the Cramer-Rao lower bound to define the limit precision of the estimator:

$$\mathrm{Var}\,\hat{\tau} \geq \frac{1}{R\,.\,\beta_s^2} \qquad (2)$$

where R is the signal-to-noise ratio, measured at the intercorrelation output:

$$R=\frac{\alpha^2\,E_s}{\gamma_0} \qquad (3)$$

E_s is the signal energy, S(ν) is the Fourier Transform of s(t). β_s is the effective bandwidth:

$$\beta_s=\frac{1}{E_s}\int \nu^2\,|S(\nu)|^2\,d\nu \;;\quad E_s=\int s^2(t)\,dt \qquad (4)$$

which is a root mean square measure of spectral spreading of the signal [3].

2.2. Passive procedure

As the emitted signal is unknown, we need two observation points for time delay estimation:
$r_1(t)= s(t) + b_1(t)$
$r_2(t)= \alpha\, s(t-\tau) + b_2(t)$
where $b_1(t)$ and $b_2(t)$ are independent identically

distributed gaussian white noises. τ is now the propagation delay between the two observation points. Although the estimate structure seems like that of the active case (intercorrelation between $r_1(t)$ and $r_2(t)$, and maximum detection), these two procedures are basically different, since the reference $r_1(t)$ is now corrupted by noise. Quazi has shown in [2] that for bandpass signals active procedure leads to more precise estimates than the passive one, by a rate of $1/(SNR)^{1/2}$ where SNR is the signal power to noise power ratio (corresponding in the active case to the signal to noise ratio measured in the signal bandwidth *before* the intercorrelation operation).
In the following, we only treat active systems.

3. TWO-PATH PROPAGATION

In an active context, we modelise the signal received after propagation in a two-path channel:

$r(t)= \alpha_1 s(t-\tau_1) +\alpha_2 s(t-\tau_2) +b(t)$, $t\in(T)$, where $b(t)$ is still supposed white and gaussian with psd γ_0. $s(t)$ is the known deterministic signal. τ_1 and τ_2 are both delays to be estimated. α_1 and α_2 are known real attenuations.

3.1. Joint maximum likelihood estimator. Optimal structure

Writing in the above model:
$Y(t,\tau_1,\tau_2)=\alpha_1 s(t-\tau_1)+\alpha_2 s(t-\tau_2)$, the ML estimates $(\hat\tau_1,\hat\tau_2)$ must verify the set of likelihood equations [4]:

$$\begin{cases} \displaystyle\int_{(T)} \frac{\partial Y(t,\tau_1,\tau_2)}{\partial \tau_1}\,[r(t)-Y(t,\tau_1,\tau_2)]dt \Big|_{\substack{\tau_1=\hat\tau_1\\ \tau_2=\hat\tau_2}} =0 \\ \displaystyle\int_{(T)} \frac{\partial Y(t,\tau_1,\tau_2)}{\partial \tau_2}\,[r(t)-Y(t,\tau_1,\tau_2)]dt \Big|_{\substack{\tau_1=\hat\tau_1\\ \tau_2=\hat\tau_2}} =0 \end{cases} \qquad (5)$$

This leads to the following system:
(with the notation: $x'(t)= \frac{dx(t)}{dt}$)

$$\begin{cases} \Gamma'_{rs}(\hat\tau_1) = \alpha_1 \Gamma'_{ss}(\hat\tau_1-\hat\tau_2) \\ \Gamma'_{rs}(\hat\tau_2) = \alpha_2 \Gamma'_{ss}(\hat\tau_1-\hat\tau_2) \end{cases} \qquad (6)$$

The estimator structure is symmetrical vs τ_1 and τ_2 but exhibits a coupling expressed by $\Gamma'_{ss}(\tau_1-\tau_2)$.

3.2. $(\hat\tau_1,\hat\tau_2)$ estimator performances

As in the one-path case, we have no explicit expression of the estimate, and we can only write bounds for the estimator performances. First of all, the estimates are asymptotically unbiased ($R\to\infty$). The estimates variances will be characterized by the expressions of the Cramer-Rao bounds.
Let I be the Fisher information matrix, whose elements are:

$$I_{ij} = \frac{1}{\gamma_0}\int_{(T)} \frac{\partial Y(t,\tau_1,\tau_2)}{\partial \tau_i}\cdot\frac{\partial Y(t,\tau_1,\tau_2)}{\partial \tau_j}dt \qquad (7)$$

The matrix is easily calculated:

$$\mathbf{I} = \begin{bmatrix} -\frac{\alpha_1^2}{\gamma_0}\Gamma''_{ss}(0) & -\frac{\alpha_1\alpha_2}{\gamma_0}\Gamma''_{ss}(\tau_1-\tau_2) \\ -\frac{\alpha_1\alpha_2}{\gamma_0}\Gamma''_{ss}(\tau_1-\tau_2) & -\frac{\alpha_2^2}{\gamma_0}\Gamma''_{ss}(0) \end{bmatrix}$$

In the case of unbiased estimators, the Cramer-Rao bounds matrix is given by $\mathbf{I}^{-1}$. We detail here the final expressions of the variance:

$$\mathrm{Var}\,\hat\tau_i \geq \frac{1}{R_i\cdot\beta_s^2}\cdot\frac{1}{1-\rho^2(\tau_2-\tau_1)} \quad i=1,2 \qquad (8)$$

with: $\rho(\tau_2-\tau_1)\triangleq \dfrac{\Gamma''_{ss}(\tau_2-\tau_1)}{\Gamma''_{ss}(0)}$

$$\text{or } \mathrm{Var}\,\hat\tau_i \geq \frac{1}{\mathfrak{R}}\cdot\frac{\mathfrak{R}}{R_i}\cdot\frac{1}{\beta_s^2\cdot[1-\rho^2(\tau_2-\tau_1)]} \triangleq \frac{1}{\mathfrak{R}}\cdot\frac{1}{B_T^2} \qquad (9)$$

R_i is the signal-to-noise ratio for the path i, at the matched filter (or intercorrelation) output, and $\mathfrak{R}$ is the total signal-to-noise ratio at the matched filter output, given by:

$$R_i = \frac{\alpha_i^2\,\Gamma_{ss}(0)}{\gamma_0}\ ,\ \mathfrak{R} = R_1+R_2+2\,\frac{\alpha_1\alpha_2\,\Gamma_{ss}(\tau_2-\tau_1)}{\gamma_0} \qquad (10)$$

β_s is the effective bandwidth defined in (4).
We don't give here the expression of the two other terms of **I**. In fact, the matrix **I** is not diagonal in general, and the estimates $\hat\tau_1$ and $\hat\tau_2$ are *not decorrelated*. But when $|\tau_1-\tau_2|$ *is large with respect to the correlation duration of s(t)* , **I** becomes diagonal, which means that the estimates $\hat\tau_1$ and $\hat\tau_2$ are asymtotically decorrelated. In this particular case, the system (6) becomes:

$$\begin{cases} \Gamma'_{rs}(\hat\tau_1)=0 \\ \Gamma'_{rs}(\hat\tau_2)=0 \end{cases} \qquad (11)$$

The optimal structure reduces to a structure similar to figure 1 for each delay. The Cramer-Rao bound simplifies into:

$$\mathrm{Var}\,\hat\tau_i \geq \frac{1}{R_i\cdot\beta_s^2}$$

Briefly, when the propagation delays are well separated, we can estimate each delay as if it were alone, and apply directly the results of §2.

3.3. Time difference of arrival $(\hat\tau_2-\hat\tau_1)$

Instead of the couple (τ_1,τ_2), we may consider the new couple (σ,δ) where $\sigma=\tau_1+\tau_2$ and $\delta=\tau_1-\tau_2$. Then we rewrite the model as: $r(t)=Y(t,\sigma,\delta)+b(t)$ where:

$$Y(t,\sigma,\delta)= \alpha_1 s\left(t-\frac{\sigma+\delta}{2}\right) + \alpha_2 s\left(t-\frac{\sigma-\delta}{2}\right)$$

Optimal structure: The ML estimator must now fulfill the set of equations:

$$\begin{cases} \alpha_1 \Gamma_{rs}'\left(\frac{\hat{\sigma}+\hat{\delta}}{2}\right)+\alpha_2 \Gamma_{rs}'\left(\frac{\hat{\sigma}-\hat{\delta}}{2}\right)=0 \\ -\alpha_1 \Gamma_{rs}'\left(\frac{\hat{\sigma}+\hat{\delta}}{2}\right)+\alpha_2 \Gamma_{rs}'\left(\frac{\hat{\sigma}-\hat{\delta}}{2}\right)+2\alpha_1\alpha_2 \Gamma_{ss}'(\hat{\delta})=0 \end{cases}$$

Assuming that $\delta \ll \sigma$, we may develop Γ_{rs}' about $\hat{\sigma}/2$, at the first order. Then we get:

$$\begin{cases} \hat{\delta}\, \Gamma_{rs}''\left(\frac{\hat{\sigma}}{2}\right) - (\alpha_1+\alpha_2)\, \Gamma_{ss}'(\hat{\delta}) = 0 \\ 2\, \Gamma_{rs}'\left(\frac{\hat{\sigma}}{2}\right) + (\alpha_1-\alpha_2)\, \Gamma_{ss}'(\hat{\delta}) = 0 \end{cases} \qquad (12)$$

Estimate performances: As in the previous paragraph, we calculate the Fisher information matrix for the new couple of parameters (σ,δ). It leads to the following Cramer-Rao bounds (13), supposing the estimates to be unbiased:

$$\begin{aligned} \text{Var}\,\hat{\sigma} &\geq \frac{1}{\mathfrak{N}} \frac{E_Y}{E_s} \frac{(\alpha_1^2+\alpha_2^2)-2\alpha_1\alpha_2\,\rho(\delta)}{\alpha_1^2\,\alpha_2^2\,\beta_s^2\,[1-\rho^2(\delta)]} \triangleq \frac{1}{\mathfrak{N}} \frac{1}{B_\sigma^2} \\ \text{Var}\,\hat{\delta} &\geq \frac{1}{\mathfrak{N}} \frac{E_Y}{E_s} \frac{(\alpha_1^2+\alpha_2^2)+2\alpha_1\alpha_2\,\rho(\delta)}{\alpha_1^2\,\alpha_2^2\,\beta_s^2\,[1-\rho^2(\delta)]} \triangleq \frac{1}{\mathfrak{N}} \frac{1}{B_\delta^2} \end{aligned} \qquad (13)$$

$\mathfrak{N}$, E_Y, E_s, $\rho(\ )$ and β_s are described in (9), (8) and (4) respectively.
In the general case, both estimates $\hat{\sigma}$ and $\hat{\delta}$ are correlated.

Particular case: $\alpha_1=\alpha_2=\alpha$ $(R_1=R_2=R)$ When the attenuations are the same on each path, then the matrix I becomes diagonal i.e. the ML estimates $\hat{\sigma}$ (summation of the delays) and $\hat{\delta}$ (difference between the delays) are asymptotically decorrelated. In this case, the ML equations (12) are:

$$\begin{cases} \hat{\delta}\, \Gamma_{rs}''\left(\frac{\hat{\sigma}}{2}\right) - 2\,\alpha\, \Gamma_{ss}'(\hat{\delta}) = 0 \\ \Gamma_{rs}'\left(\frac{\hat{\sigma}}{2}\right) = 0 \end{cases} \qquad (14)$$

This means that the ML structure consists of first estimating the mean position of the pair of delays and then separating both delays symmetrically from the mean position estimate.
The estimate variances are bounded by:

$$\begin{cases} \text{Var}\,\hat{\sigma} \geq \frac{2}{R\,.\,\beta_s^2} \frac{1}{1+\rho(\delta)} \\ \text{Var}\,\hat{\delta} \geq \frac{2}{R\,.\,\beta_s^2} \frac{1}{1-\rho(\delta)} \end{cases} \qquad (15)$$

where $R=R_1=R_2$, which are the same formulae as those given by Munier [5]. The position $\hat{\sigma}$ is naturally always better estimated than the difference $\hat{\delta}$ when $\hat{\delta}$ is small.

As a conclusion to part 3, we can say that we have analysed the general active case of a two-path propagation, presenting the optimal estimates structure and their statistical performances. We pointed out two different particular cases leading to decorrelated estimates.

4. APPLICATION TO AN UNDERWATER ACOUSTICS EXPERIMENT

4.1. Description of the experiment

This active middle range experiment sets two-stopped boats into action, as described in figure 2. A low frequency large bandwidth signal is periodically emitted from one boat and propagates to the reception hydrophone 140 km away. According to a primary propagation knowledge, there may be several propagation paths. The emitted signal is a biphase modulated 60 Hz carrier. The modulation is achieved by a 127-maximal-length-binary-sequence (MLBS), of elementary binary digit θ=67ms. This kind of signal, called phase-shift-keying (PSK) signal, has a sharp correlation function, which is convenient in channel identification problems [1] [6].

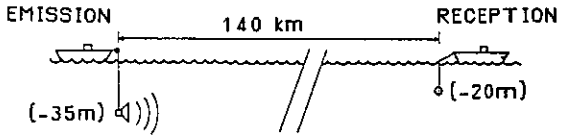


Figure 2

4.2. Performance limits

Assuming there are two paths, the expected performances of time delay estimates are bounded by formulae (9) and (13). We present in figure 3 the graph of $1/B_\tau$ versus the difference $(\tau_1-\tau_2)$, for different

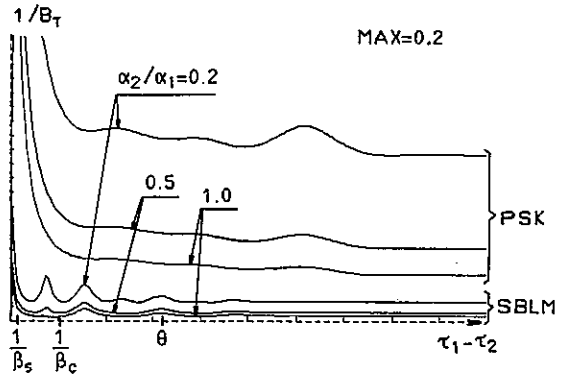


Figure 3

values of the ratio α_2/α_1, for the PSK signal and the MLBS respectively. We notice immediately that using PSK signals leads to more accurate estimates than MLBS, since:
β_s (PSK effective bandwidth) $\gg$ β_c (MLBS effective bandwidth): $\beta_s^2 = \beta_c^2 + 4\,\pi^2\nu_0^2$ where ν_0 is the carrier frequency [1]. We have indicated in the graph the abscisse θ, usually defining the temporal resolution of the MLBS, and the abscissa $1/\beta_s$ and $1/\beta_c$

defining in a certain sense the limit resolution of ML methods with these signals [5]. From figure 3, we check that, when $\tau_1-\tau_2$ is large with respect to θ, the ordinate remains constant and equal to $[\Re/(R_1.\beta_s^2)]^{\frac{1}{2}}$ or $[\Re/(R_1.\beta_c^2)]^{\frac{1}{2}}$ for PSK and MLBS respectively (this corresponds to decorrelated estimates). We have drawn on figure 4 the graphs of $1/B_\sigma$ and $1/B_\delta$ versus $\delta=\tau_1-\tau_2$ for various values of α_2/α_1, in the case of PSK signals. We notice that the value of the ordinate oscillates on both sides of the asymptotic value. The case $\alpha_2/\alpha_1=1$ corresponds to decorrelated estimates $\hat{\sigma}$ and $\hat{\delta}$.

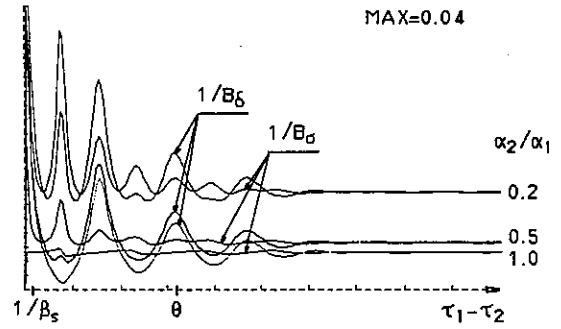


Figure 4

4.3. Experimental results

At each emission, we intercorrelate the received signal with a copy of the emitted signal. An envelope detection leads to figure 5, where we get an image of the square channel impulse response.

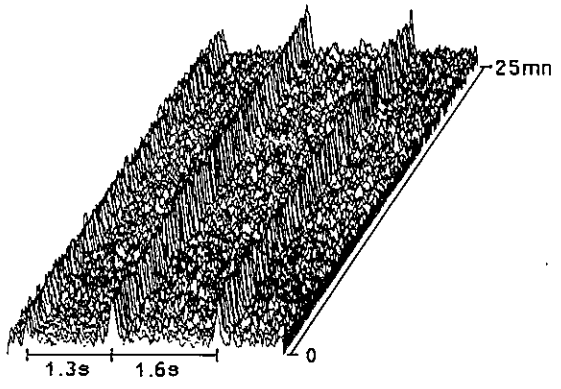


Figure 5

We notice three main peaks of rather constant amplitude. These peaks are far enough from each other to be treated independently. A priori knowledge of the propagation (such as ray-tracing) is in good agreement with these three paths; but it seems also possible that each group is composed of at least two close signal arrivals. As a matter of fact, working with these real data necessitates the introduction of more than two propagation paths in the model of §3. On the other hand, working directly with the received bandpass signal leads to introduce, in the propagation model, phase terms for each path [1]. Such a model would lead obviously to a very complex ML optimal structure. We think that a two-step approach is better adapted in this case: it would be possible to apply high resolution methods to each group of delays exhibited above.

5. CONCLUSION

Considering the case of a two-path propagation, we have described delay estimates in the ML sense. Although the optimal structure seems not very easy to implement in the most general case, we noticed nevertheless some particular cases leading to simplified decorrelated estimates. Considering a real data experiment, we apply the previous results in order to predict the limit precision of the delay estimates. But in a general propagation case, more than two paths are available, which leads to complicated ML structures. Yet other kinds of methods, such as AR modelisation, Pisarenko method or other high resolution methods, don't increase so much in complexity with the number of paths and may provide good estimates even if time delay difference is less than temporal signal resolution.

ACKNOWLEDGEMENTS

This work was supported by the Direction of the French Naval Constructions.

REFERENCES

[1] Jourdain, G. and Pallas, M.A., Multiple time delay estimation in underwater acoustic propagation, in: Stochastic Processes in Underwater Acoustics (Springer Verlag), in print.
[2] Quazi, A.H., An overview of the time delay estimate in active and passive systems for target localization, IEEE ASSP, n°3 (1981)
[3] Levine, B., Fondements théoriques de la radiotechnique statistique tome 2 (Editions MIR, Moscou, 1973)
[4] Sage, A.P. and Melsa, J., Estimation theory with applications to communication and control (Mac Graw Hill, 1971)
[5] Munier, J., Pouvoir séparateur en estimation non linéaire en présence de bruit faible, colloque GRETSI, Nice, 1977
[6] Henrioux, J.P., Génération de séquences binaires, rapport interne CEPHAG n°5/72, 1972

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N2.6

A NEW SIGNAL ESTIMATION USING A RECEPTION MODEL

P. COMON and J.L. LACOUME

CEPHAG, UA CNRS 346, INPG/IEG, BP 46, 38402 Saint Martin d'Hères Cedex, France

This talk designs a reception model for waves propagated through a deterministic medium. Therefore, the concept of 'propagation modes' is introduced, and the modes properties are emphasized. This very general approach leads to an observation model which is shown to be useful in many situations occuring in physics. The unknown parameters are on one hand deterministic, and on the other hand random. For this reason, a new optimization procedure is proposed allowing the estimation of a remote source. In the case of a totally polarized source, the estimate turns out to be very simple.

1. INTRODUCTION

Based on the physical laws of wave propagation, some linear models have been proposed for estimating the excitation source from the observations. The number of such papers is so great that we do not try here to give an exhaustive list of references. All these contributions are founded on the combination of a physical model and of an 'optimal' estimation procedure. Some light will be shed on these two closely connected problems in this paper.

We first show that, with general assumptions (linearity, stationarity, homogeneity), the laws of wave propagation in a deterministic medium lead to a general linear modeling of the observed signals. In this model, we can distinguish the random excitation factors issuing from the sources and the deterministic relations imposed by the medium of propagation.

We introduce then a new estimation criterion combining maximum a posteriori (for the stochastic parameters) and maximum likelihood (for the deterministic parameters).

2. A RECEPTION MODEL

In this section, we want to establish a model for the wavefield received on an array of sensors. Therefore, general considerations upon the propagation laws and the properties of the propagation medium are taken into account. This model is available in many physical situations, including electromagnetic, acoustic, or elastic waves in different kinds of media.

2.1. Physical laws governing propagation

The wave propagation in a physical medium is governed by integro-differential equations describing on one hand the propagation laws, and on the other hand the medium properties :

- The propagation relations are for example :
. Maxwell equations for electromagnetic waves.
. Mass and cinetic momentum conservation for elastic or acoustic waves.

- The relations linked to the medium are for example :
. Local relation between the current density and the electric field for electromagnetic waves.
. The stress and constraints tensorial relations in which the pressure enters, for the elastic waves in solids or for acoustic waves in fluids.
These relations describing the medium properties are not linear, but can be linearized by assuming small perturbations. In the following, we limit our investigation to the linear case.

2.2. Statistical assumptions

The physical laws can be summarized by :

$$\Lambda\left[\,\vec{V}(\vec{r},t)\,\right] = \vec{S}(\vec{r},t) \qquad (1)$$

where :

. Λ is an integro-differential linear operator.
. $\vec{V}(\vec{r},t)$ is a vector standing for the wave field.
. $\vec{S}(\vec{r},t)$ is the 'source term' standing for the excitation, that is the input of the system.

We shall consider the wave field $\vec{V}(\vec{r},t)$ and the source term $\vec{S}(\vec{r},t)$ as multivariate stochastic functions, since their properties are known only in mean, whereas the propagation medium will be assumed to be deterministic.

* This work has been supported by the Direction of the French Naval Constructions.

This kind of modeling is correct in a lot of encountered situations, namely the passive detection in submarine acoustics, the characterization of natural electromagnetic phenomena issuing from storm effects or induced telluric currents, earth environment electrical fluctuations, sun, planet and stars radiations. This model is also well fitted to most natural seismic phenomena resulting from the natural movements of the earth...

2.3. Physical assumptions

In order to go further, we have to introduce 3 additional assumptions, not very restrictive :

(A1) The sources are located in a domain Δ_s that can be clearly separated from the measure domain Δ_m containing the array of sensors.

(A2) The medium of propagation is invariant versus space (homogeneity of Λ) and time (stationarity of Λ).

(A3) The sources are stationary random variables, and the steady state is reached.

The set of above assumptions leads to two main consequences :

First, from (A2), the linear operator Λ commutes with space and time translations. Thus, the Monochromatic Plane Waves (MPW) defined as

$$\vec{\phi}_{\vec{k},w}(\vec{r},t) = \vec{A}(\vec{k},w)\, e^{j(\vec{k}^T\vec{r}-wt)} \qquad (2)$$

are eigenfunctions of operator Λ . In this expression, k denotes the wave vector describing the spatial (periodic) structure of the wave, and w denotes the pulsation giving its time (periodic) variations.

Secondly, in the measurement domain Δ_m, which is free of sources (A1), equation (1) can be rewritten :

$$\Lambda[\ \vec{V}(\vec{r},t)\ ] = \vec{0} \qquad (3)$$

Assumptions (A2) and (A3) allow to take the space-time Fourier Transform of equation (3). Operator Λ splits then into an infinite set, parametrized by wave number $\vec{k}$ and pulsation w, of finite rank operators $L(\vec{k},w)$. The compactness of linear operator Λ , which is of practical value, yields among other things that the MPW form a complete orthogonal basis. Furthermore, any vector $\vec{V}(\vec{r},t)$ can be split onto this basis and its components can be denoted $\vec{V}(\vec{k},w)$:

$$L(\vec{k},w)\ \vec{V}(k;w) = \vec{0} \quad ; \quad \forall\ \vec{k},w \qquad (4)$$

where : $\underline{L}$ is a n x n deterministic matrix
$\vec{V}$ is a n x 1 random vector

The two main results involved by this equation are the dispersion relation and the existence of eigenmodes.

2.4. Dispersion relation

In order for the linear relation (4) to have a solution, the relation below must be satisfied :

$$\text{Det}\quad L(\vec{k},w) = 0 \qquad (5)$$

In the 4-dimensional $\{\vec{k},w\}$ space, this relation defines a 3-dimensional hypersurface family ξ_D that we call the dispersion surface. Note that ξ_D can contain 1 to n sheets. A necessary condition for a MPW to be a solution of (4) is that its values k and w satisfy relation (5). This is, presented here in a general form, the well-known property giving the velocity of propagation versus direction and frequency.

2.5. Eigenmodes of propagation, polarization states

When the dispersion relation is satisfied, the pair $(\vec{k},w)$ belongs to the surface ξ_D and the solutions of (4) are eigenvectors of $L(\vec{k},w)$ associated with the null eigenvalue. This provides a family of particular MPW, that we index by p, called eigenmodes of propagation :

$$\vec{\phi}_{p,\vec{k},w}(\vec{r},t) = \vec{A}_p(\vec{k},w)\, e^{j(\vec{k}^T\vec{r}-wt)} \qquad (6)$$

with : $(\vec{k},w) \in \xi_D$
$\vec{A}_p(\vec{k},w)$: eigenvector of $L(\vec{k},w)$ associated with eigenvalue zero.

These eigenmodes form an orthonormal basis of operator Λ's kernel. Any wave field solution can be expressed in this basis.

Let us now look at the polarization property relating the wave field components. The polarization state of the wave field associated with each eigenmode depends on the multiplicity of the null eigenvalue, that is of the root of (5). We distinguish :

- The pure state : a total polarization is associated with a simple root of (5) ; in this case, the space spanned by the single eigenmode is 1-dimensional. Furthermore, the corresponding components of the wave field can be written :

$$\vec{U}_{\vec{k},w}(\vec{r},t) = \alpha\ (\vec{k},w)\ \vec{\phi}_{1,\vec{k},w}(\vec{r},t) \qquad (7)$$

α is a random amplitude fixed by the source excitation and $\phi_1(r,t)$ is the single eigenmode associated with the zero eigenvalue. The wave field is a random vector, but there exists a deterministic almost sure relation between its n components.

- Partial polarization : when the multiplicity of the zero eigenvalue of $L(\vec{k},w)$ is $q(\vec{k},w) > 1$, the solutions of (5) span a q-dimensional space and the polarization becomes partial.

Finally, the general solution is a sum over the different modes p, and over the values of $(\vec{k},w)$ belonging to the dispersion surface sheets ξ_D^p.

It can be written, taking the time Fourier Transform :

$$\vec{V}(\vec{r},\nu) = \sum_{p=1}^{Q} \int_{(k,\nu)\in\xi_D^p} \alpha_p(\vec{k},\nu)\, \vec{A}_p(\vec{k},\nu)\, e^{j\vec{k}^T\vec{r}}\, d\lambda(\vec{k}) \quad (8)$$

Q denoting the maximum number of distinct modes ($Q < n$) and $\alpha_p(\vec{k},w)$ denoting the random excitation factors fixed by the power and the geometry of the sources.

In order to conclude this section by stating the observation model used in the following section, let us look more precisely at the form of the solution $\vec{V}(\vec{r},t)$.

2.6. Modeling of the received wave

The model is based on the fact that the sources have a fixed (that is deterministic) geometry, whereas the time variation of the signals emitted by each source is stochastic.

The excitation factor entering the source expression can be written as :

$$\alpha_p(\vec{k},w) = \sum_{i\in\Delta_s} X_{p,i}(w)\, e_{p,i}(\vec{k}) \quad (9)$$

where i labels the different sources, $X_{p,i}(w)$ characterize the time (random) part of the variations, and $e_{p,i}(\vec{k})$ describe the (deterministic) geometry of the sources.

It can be emphasized that the variables w and k turn out to be separated in the expression of the excitation. Then this yields the following general form for the solutions $\vec{V}(\vec{r},\nu)$:

$$\vec{V}(\vec{r},\nu) = \sum_{i} \sum_{p=1}^{Q} X_{p,i}(\nu) \cdot \vec{W}_{p,i}(\vec{r},\nu)$$

where

$$\vec{W}_{p,i}(\vec{r},\nu) = \int_{(\vec{k},\nu)\in\xi_D^p} e_{p,i}(\vec{k})\, \vec{A}_p(\vec{k},\nu)\, e^{j\vec{k}^T\vec{r}}\, d\lambda(\vec{k}) \quad (10)$$

This kind of model has been already used to identify ground anomalies from natural electromagnetic field measurements [2]. This general result simplifies if the case of a single source and a single mode is considered :

$$\vec{V}(r,\nu) = X(\nu)\, \vec{W}(r,\nu) \quad (11)$$

In this formula, $\vec{W}$ is a deterministic n x 1 vector and describes the relations between the components of $\vec{V}$, that is the polarization, involved here only by the propagation properties and the source geometry. Moreover, let us emphasize that if the sources are stationary, the excitation factors $X_{p,i}(\nu)$ are necessary uncorrelated for two different values of the pulsation ν.

3. ESTIMATION PROCEDURE

3.1. Observation model

Suppose we wish to isolate a source from the others, by estimating its variables X(w) and $\vec{W}(\vec{r},w)$. As pointed out before, the model that we should deal with in the case of totally polarized waves is :

$$\vec{V}(\vec{r},w) = X(w)\, \vec{W}(\vec{r},w) + \vec{z}(\vec{r},w) \quad (12)$$

In the following, variables $\vec{r}$ and w are sometimes omitted for more convenience. The noise $\vec{z}$ independent of the considered source X includes the contribution of all the other sources, and also eventually an extraneous background noise.

In order to carry out a Maximum A Posteriori (MAP) estimation, we need a conditional probability density function. With that object, let :

- X be uniformly distributed
- $\vec{z}$ be zero-mean complex normal with covariance matrix Γ_z ; we denote $z \sim N_n^c(o, \Gamma_z)$.

The justification of these assumptions lies on two remarks :
i) We have few a priori statistical informations about the amplitude of the sources. Thus, the most suitable way to proceed is to suppose that the sources are uniformly distributed.
ii) The noise is the sum of many independent sources. From the central limit theorem, the noise is then asymptotically normal complex. Furthermore, to introduce some uncertainty in the knowledge of the noise, it is supposed that its covariance matrix can be written as :

$$\Gamma_z(\vec{r},w) = \gamma(r,w) \cdot G(r,w) \quad (13)$$

where G is a known n x n definite positive hermitian matrix and γ is an unknown real scalar.

3.2. An optimization criterion

The unknowns to be estimated are deterministic (like W) and random (like X). So, we propose a joint Maximum A Posteriori & Maximum Likekihood (MAP & ML) procedure already introduced in [4]. The MAP estimation of two random variables X and Y from a set of observations V is defined by

$$(X_{MAP},Y_{MAP}) = \text{Arg} \max_{X,Y} p(X,Y/V)$$

or differently by

$$(X_{MAP},Y_{MAP}) = \text{Arg} \max_{X,Y} p(X,Y,V)$$

If Y is uniformly distributed, this can be rewritten as

$$(X_{MAP},Y_{MAP}) = \text{Arg} \max_{X,Y} p(V,X/Y) \quad (14)$$

Continuing, if Y is considered as a deterministic variable, the maximization performed in (14) is a ML estimation of Y. This is well known. So, based on the result (14), we define the joint MAP & ML estimation of the pair (X,W) :

$$(X_{MAP},W_{ML}) = \text{Arg} \underset{X,\vec{W}}{\text{MAX}}\ p(V,X/W) \qquad (15)$$

Thus, the solution (X_{MAP},W_{ML}) is obtained by maximizing a single functional.

3.3. An estimate of the source

Given any space-pulsation pair $(\vec{r},w)$, we attempt to estimate jointly the following unknowns :
- random scalar variable X
- deterministic nxl vector W
- coefficient Y of the noise covariance matrix.

It can be noticed that if the pair $\{X,\vec{W}\}$ provides a representation of the source in (12), so it is with the pair $\{X/\beta, \beta\vec{W}\}$ where β is any deterministic complex scalar. So, we can lay down the constrainst to $G^{-1/2}W$ to be unitary, without loss of generality, to describe the source :

$$\vec{W}^T\, G^{-1}\, \vec{W} = 1 \qquad (16)$$

$\vec{W}^T$ denoting the conjugated transposed vector of W.

The uncertainty is now reduced to any complex β of modulus equal to 1.

In order to make the computation of $\vec{W}$ and Y, we shall use M independent observations $\vec{V}^\mu$:

$$\vec{V}^\mu = X^\mu \,.\, \vec{W} + \vec{z}^\mu \qquad (17)$$

index μ distinguishing the different independent realizations, so $1 \leq \mu \leq M$. Vector $\vec{W}$ in expression (15) does not depend on μ because it is a deterministic variable.

The unknowns are computed by maximizing the a posteriori PDF for each value of $(\vec{r},w)$.

Replacing the PDF by its value and maximizing its logarithm yields :

$$(X_{MAP},\vec{W}_{ML},Y_{ML}) = \text{Arg} \underset{X^\mu,\vec{W},Y}{\text{MAX}} \{-M\ell n \mid \det(\pi Y\, G) \mid + \sum_{\mu=1}^{M} (\vec{V}^\mu - X^\mu\vec{W})^T\, G^{-1} (\vec{V}^\mu - X^\mu\vec{W})/Y\} \qquad (18)$$

A necessary condition to access to a maximum is (taking (16) into account) that the following system is satisfied [3,4].

$$\begin{aligned} X_{MAP}^{\mu} &= \vec{W}_{ML}^T\, G^{-1}\, \vec{V}^\mu \\ S_v\, G^{-1}\, \vec{W}_{ML} &= \lambda\, \vec{W}_{ML} \\ Y_{ML} &= \frac{1}{n} \sum_{i=2}^{n} \lambda_i \ ; \ |\lambda_1| \leq \cdots \leq |\lambda_n| \end{aligned} \qquad (19)$$

where S_v denotes the sample covariance matrix of the observation :

$$S_v = \frac{1}{M} \sum_{\mu=1}^{M} \vec{V}^\mu\, \vec{V}^{\mu T} \qquad (20)$$

and where the λ_i denote the eigenvalues of $S_v\, G^{-1}$ and $\lambda \in \{\lambda_1, \ldots, \lambda_n\}$. This shows that W_{ML} is an eigenvector of $S_v\, G^{-1}$. In oder for the functional (18) to be maximized, λ must be the greatest eigenvalue λ_1 of the matrix $S_v G^{-1}$, and W_{ML} is then the associated eigenvector ω_1.

In the present case, where the solution that we are searching for is a pure state, Y_{ML} does not enter in the expression of the best estimate of the source wave field which is :

$$[X^\mu \,.\, \vec{W}]_{MAP\&ML} = \vec{\omega}_1^{\,T}\, G^{-1}\, \vec{V}^\mu \,.\, \vec{\omega}_1 \qquad (21)$$

This kind of result has been presented by Samson [5] in an other context with less general tools. Our approach allows directly the joint estimation of random and deterministic parameters, namely covariance matrices, and gives among other things theoretical foundations to the so-called ML spectral or spatial analysis introduced by Capon [1]; this latter point is being developed elsewhere.

4. CONCLUDING REMARKS

It has been shown that the general observation model (12) V = XW + z is appropriate in a great variety of situations often encountered in physics. It has been stated by taking only into account the properties of linearity, homogeneity and stationarity of the propagation medium.

In order to identify the parameters entering this model, we cope with the problem of estimating simultaneously random and deterministic parameters. Therefore, a new estimation criterion is introduced which allows giving sound foundations to many well known estimators, actually based on rather heuristic considerations.

REFERENCES

[1] Capon J., High Resolution Frequency Wavenumber Spectrum Analysis. Proc. IEEE, 57, n° 8, 1408-1418, Aug. 1969.
[2] Comon P. and F. Planson, Ground Response to Electromagnetic Natural Excitation, IASTED Internat. Symp., Paris, June 1985.
[3] Comon P., Estimation multiviarable complexe, Revue Traitement du Signal, Vol. 3, n° 2, 1986.
[4] Comon P., Traitement de signaux magnétiques multivariables, Doct. Thesis, INPG, Grenoble, Dec. 9, 1985.
[5] Samson J.C., Pure States, Polarized Waves & Principal Components in the Spectra of Multiple Geophysical Time Series. Geophys.J.R.Astr.Soc., Vol. 72, n° 3, 647-664, 1983.

A METHOD FOR DETECTING MODAL CHANGES IN SHARP SPECTRUM PLUS NOISE SIGNALS.

E. Daymier & F. Castanie
Ecole Nationale Supérieur
Lab. Traitement du Signal
Toulouse Cedex
France

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EFFICIENT ALGORITHMS FOR LINEAR PHASE STRUCTURES WITH APPLICATIONS IN SIGNAL MODELING

An-Loong Kok
Dimitris Manolakis

Department of Electrical and Computer Engineering
Northeastern University
Boston, MA 02115, USA

This paper deals with fast algorithms for the efficient solution of some Toeplitz systems of linear equations, which appear in block adaptive filtering using FIR filters with linear phase, linear prediction using FIR predictors with linear phase and speech modeling using the line spectrum pair model. Some known algorithms as well as new algorithms, are presented in a unified framework and their computational complexities are compared and analysed.

1. INTRODUCTION

An FIR filter with linear phase, i.e. constant phase and group delays or only constant group delay, is characterized by a symmetric or antisymmetric impulse response, respectively. Consider an FIR of order p with the following input-output relationship

$$y(n) = -\underline{c}_p^t \underline{x}_p(n) \tag{1}$$

where

$$\underline{c}_p = [c_1 \;\; c_2 \;\; \cdots \;\; c_p]^t \tag{2}$$

and

$$\underline{x}_p(n) = [x(n) \; x(n-1) \; \cdots \; x(n-p+1)]^t \tag{3}$$

Let J be the p×p exchange matrix having ones at the antidiagonal and zeros elsewhere. Then, the FIR filter (1) has linear phase or just constant group delay, if it satisfies the conditions [1], [2]

$$\underline{c}_p^L = J\underline{c}_p^L \tag{4}$$

$$\underline{c}_p^G = -J\underline{c}_p^G \tag{5}$$

respectively.

If z(n) is a desired response signal and x(n), z(n) are jointly stationary, then the optimum FIR filter in the MSE sense (Wiener filter), is specified by the following set of normal equations

$$R_p \underline{c}_p = -\underline{d}_p \tag{6}$$

where R_p is a p×p symmetric Toeplitz matrix obtained from the first p samples of the autocorrelation r_ℓ of x(n), and $\underline{d}_p$ is a p-vector containing the first p samples of the crosscorrelation between x(n) and z(n) [2], [3].

If we impose the constraints (4) or (5), it turns out that the resulting Wiener filters with linear phase are determined by [2]

$$R_p \underline{c}_p^L = -\frac{1}{2}(\underline{d}_p + J\underline{d}_p) \tag{7}$$

$$R_p \underline{c}_p^G = -\frac{1}{2}(\underline{d}_p - J\underline{d}_p) \tag{8}$$

Suppose now that we want to model a signal using an all-pole model with system function

$$H(z) = \frac{1}{1 + \sum_{k=1}^{p} a_k z^{-k}} \tag{9}$$

If we use the autocorrelation method of linear prediction, the model parameters are obtained by solving the following set of equations

$$R_p \underline{a}_p = -\underline{r}_p \tag{10}$$

where

$$\underline{a}_p = [a_1 \; a_2 \; \cdots \; a_p]^t \tag{11}$$

and

$$\underline{r}_p = [r_1 \; r_2 \; \cdots \; r_p]^t \tag{12}$$

However, if we impose the phase linearity constraints, (4) or (5), the corresponding models are determined by [3]

$$R_p \underline{a}_p^L = -\frac{1}{2}(\underline{r}_p + J\underline{r}_p) \tag{13}$$

$$R_p \underline{a}_p^G = -\frac{1}{2}(\underline{r}_p - J\underline{r}_p) \tag{14}$$

Let us now turn our attention to the efficient solution of the above Toeplitz linear systems of equations. The linear system (6) can be efficiently solved using the Levinson algorithm, which requires $2p^2$ MAD (Multiplications and Divisions). In contrast, (10) can be solved using the Durbin algorithm with p^2 MAD. Although the Levinson algorithm can be applied to (7), (8), (12) and (13), more efficient solutions are possible, at least for (7) and (12), using an algorithm introduced at [4]. This scheme requires approximately $1.25p^2$ MAD. In case both $\underline{c}_p^L$ and $\underline{c}_p^G$ or $\underline{a}_p^L$ and $\underline{a}_p^G$ are required, a more efficient solution can be obtained using the relations

$$\underline{c}_p^L = \frac{1}{2}(\underline{c}_p + J\underline{c}_p) \quad , \quad \underline{c}_p^G = \frac{1}{2}(\underline{c}_p - J\underline{c}_p) \tag{15}$$

$$\underline{a}_p^L = \frac{1}{2}(\underline{a}_p + J\underline{a}_p) \quad , \quad \underline{a}_p^G = \frac{1}{2}(\underline{a}_p - J\underline{a}_p) \tag{16}$$

developed in [2], [5] and the Levinson or Durbin algorithm respectively.

It is interesting to be noticed at this point that the linear phase all-pole models specified by (16) are identical within a constant to the singular predictors used in the line spectrum pair modeling of speech, introduced by Itakura [6], [7].

Indeed, the singular linear predictors introduced by Itakura are specified by

$$R_p \, \underline{a}_p^+ = -(\underline{r}_p + J\underline{r}_p) \tag{17}$$

$$R_p \, \underline{a}_p^- = -(\underline{r}_p - J\underline{r}_p) \tag{18}$$

Hence

$$\underline{a}_p^+ = 2\underline{a}_p^L \tag{19}$$

$$\underline{a}_p^- = 2\underline{a}_p^G \tag{20}$$

The objective of this paper is to introduce efficient algorithms for the efficient solution of (6), (7), (8), (13) and (14). This work extents results presented in [8], in the context of stability tests, and in [9], for the efficient solution of (10).

2. LINEAR PHASE PREDICTION AND HARMONIC PAIR MODELING

The efficient order recursive algorithms for the solution of (17) or (18), to be described below, exploit the fact that the symmetric Toeplitz matrix R_p is persymmetric, i.e. $JR_p = R_pJ$, and the right-hand side vector is either symmetric or antisymmetric. A proof of these algorithms using a polynomial approach is given in [8], [9], whereas a matrix based proof is available in [10].

The algorithms for the recursive computation of the singular predictors $\underline{a}_p^+$ and $\underline{a}_p^-$, and hence of $\underline{a}_p^L$ and $\underline{a}_p^G$ are given in the following tables.

ORDER RECURSIVE ALGORITHM FOR $\underline{a}_p^+$

Input data: $r_0, r_1, \cdots, r_p$

Initialization: $\gamma_1^+ = r_0 + r_1$, $a_1^+ = -2r_1/r_0$ (21)

For m=1 step 1 to p-1 do

$$\gamma_{m+1}^+ = r_0 + \underline{r}_m^t \, \underline{a}_m^+ + r_{m+1} \tag{22}$$

$$1+K_{m+1}^+ = \gamma_{m+1}^+ / \gamma_m^+ \tag{23}$$

$$\underline{a}_{m+1}^+ = \begin{bmatrix} \underline{a}_m^+ \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{a}_m^+ \end{bmatrix} - (1+K_{m+1}^+) \begin{bmatrix} 1 \\ \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} \tag{24}$$

End of m loop

ORDER RECURSIVE ALGORITHM FOR $\underline{a}_p^-$

Input data: $r_0, r_1, \cdots, r_p$

Initialization: $\gamma_1^- = r_0 - r_1$, $a_1^- = 0$ (25)

For m=1 step 1 to p-1 do

$$\gamma_{m+1}^- = r_0 + \underline{r}_m^t \, \underline{a}_m^- - r_{m+1} \tag{26}$$

$$1+K_{m+1}^- = \gamma_{m+1}^- / \gamma_m^- \tag{27}$$

$$\underline{a}_{m+1}^- = \begin{bmatrix} \underline{a}_m^- \\ -1 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{a}_m^- \end{bmatrix} - (1+K_{m+1}^-) \begin{bmatrix} 1 \\ \underline{a}_{m-1}^- \\ -1 \end{bmatrix} \tag{28}$$

End of m loop

Due to the symmetry and antisymmetry of $\underline{a}_m^+$ and $\underline{a}_m^-$, each of the recursions (22), (24), (26) and (28) requires m/2 multiplications instead of m. This results to a computational complexity of $0.5p^2+0.5p$ and $0.5p^2-0.5p$ for the computation of $\underline{a}_p^+$ and $\underline{a}_p^-$, respectively.

The computation of the total errors, if needed, can be carried out using the formulars

$$\alpha_m^+ = r_0 + \underline{r}_m^t \, \underline{a}_m^L = \frac{1}{2}(\gamma_{m+1}^+ + r_0 - r_{m+1}) \tag{29}$$

$$\alpha_m^- = r_0 + \underline{r}_m^t \, \underline{a}_m^G = \frac{1}{2}(\gamma_{m+1}^- + r_0 + r_{m+1}) \tag{30}$$

3. LINEAR PHASE FIR WIENER FILTERING

In this section we present two new order recursive algorithms for the efficient computation of FIR Wiener filter with linear phase. To simplify the notation, instead of (7) and (8) we consider the solution of the following linear systems of equations

$$R_p \, \underline{c}_p^+ = -(\underline{d}_p + J\underline{d}_p) \tag{31}$$

$$R_p \, \underline{c}_p^- = -(\underline{d}_p - J\underline{d}_p) \tag{32}$$

The developed algorithms are summarized in the following tables. A complete derivation is given in [10].

ORDER RECURSIVE ALGORITHM FOR $\underline{c}_p^+$

Input data: $r_0, r_1, \cdots, r_p$
$d_1, d_2, \cdots, d_p$

Initialization: $a_{-1}^+ = 2$, $\gamma_0^+ = r_0$ (33)

$c_1^+ = -2d_1/r_0$, $\gamma_1^L = d_1$ (34)

For m=1 step 1 to p-1 do

$$\gamma_m^+ = r_0 + \underline{r}_{m-1}^t \underline{a}_{m-1}^+ + r_m \tag{35}$$

$$1+K_m^+ = \gamma_m^+ / \gamma_{m-1}^+ \tag{36}$$

$$\underline{a}_m^+ = \begin{bmatrix} \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ a_{m-1}^+ \end{bmatrix} - (1+K_m^+) \begin{bmatrix} 1 \\ \underline{a}_{m-2}^+ \\ 1 \end{bmatrix} \tag{37}$$

$$\gamma_{m+1}^L = \underline{r}_m^t \underline{c}_m^+ + d_{m+1} \tag{38}$$

$$K_{m+1}^L = (\gamma_{m+1}^L - \gamma_m^L) / \gamma_m^+ \tag{39}$$

$$\underline{c}_{m+1}^+ = \begin{bmatrix} \underline{c}_m^+ \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \underline{c}_m^+ \end{bmatrix} - \begin{bmatrix} 0 \\ \underline{c}_{m-1}^+ \\ 0 \end{bmatrix} - K_{m+1}^L \begin{bmatrix} 1 \\ \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} \tag{40}$$

End of m loop

ORDER RECURSIVE ALGORITHM FOR $\underline{c}_p^-$
Input data: $r_0, r_1, \cdots, r_p$
$d_1, d_2, \cdots, d_p$

Initialization: $\gamma_m^- = r_0$, $c_1^- = 0$, $\gamma_1^G = -d_1$ (41)

For m=1 step 1 to p-1 do

$$\gamma_m^- = r_0 + \underline{r}_{m-1}^t \underline{a}_{m-1}^- - r_m \tag{42}$$

$$1+K_m^- = \gamma_m^- / \gamma_{m-1}^- \tag{43}$$

$$\underline{a}_m^- = \begin{bmatrix} \underline{a}_{m-1}^- \\ -1 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{a}_{m-1}^- \end{bmatrix} - (1+K_m^-) \begin{bmatrix} 1 \\ \underline{a}_{m-2}^- \\ -1 \end{bmatrix} \tag{44}$$

$$\gamma_{m+1}^G = \underline{r}_m^t \underline{c}_m^- - d_{m+1} \tag{45}$$

$$K_{m+1}^G = (\gamma_{m+1}^G - \gamma_m^G) / \gamma_m^- \tag{46}$$

$$\underline{c}_{m+1}^- = \begin{bmatrix} \underline{c}_m^- \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \underline{c}_m^- \end{bmatrix} - \begin{bmatrix} 0 \\ \underline{c}_{m-1}^- \\ 0 \end{bmatrix} - K_{m+1}^G \begin{bmatrix} 1 \\ \underline{a}_{m-1}^- \\ -1 \end{bmatrix} \tag{47}$$

End of m loop

Due to the symmetry and antisymmetry of $\underline{c}_m^+$ and $\underline{c}_m^-$, each of the recursions (35), (37), (38), (40), (42), (44), (45), and (47) requires m/2 multiplications instead of m. This results to a computational complexity of p^2 and p^2-2p approximately for the computation of $\underline{c}_p^+$ and $\underline{c}_p^-$, respectively.

The computation of the total errors, if needed, can be carried out using the formulas

$$\alpha_m^L = r_{zz}(0) + \frac{1}{2} \underline{d}_m^t \underline{c}_m^+ \tag{48}$$

$$\alpha_m^G = r_{zz}(0) + \frac{1}{2} \underline{d}_m^t \underline{c}_m^- \tag{49}$$

where $$r_{zz}(0) = \epsilon\{z^2(n)\} \tag{50}$$

ϵ denotes the mathematical expectation and z(n) is the desire output of the filter.

4. LINEAR PREDICTION

As it has been shown in [9], the linear predictor $\underline{a}_p$, specified by (10), can be recovered from $\underline{a}_m^+$ using the following formula

$$\begin{bmatrix} \underline{a}_{p-1} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ \underline{a}_{p-1} \end{bmatrix} + \begin{bmatrix} \underline{a}_p^+ \\ 1 \end{bmatrix} - (1+K_p) \begin{bmatrix} 1 \\ \underline{a}_{p-1}^+ \\ 1 \end{bmatrix} \tag{51}$$

where the reflection coefficient K_p is computed by

$$1+K_p = (2+\sum_{n=1}^{p} a_{p,n}^+) / (2+\sum_{n=1}^{p-1} a_{p-1,n}^+) \tag{52}$$

where $$\underline{a}_p^+ = [a_{p,1}^+ \; a_{p,2}^+ \; \cdots \; a_{p,p}^+]^t \tag{53}$$

This process requires one division and p/2 multiplications. However, if we want a truly order recursive algorithm which provides $\underline{a}_m$ for m=1,2,···,p, recursion (51) should be carried out for all values of m. In this case we can compute the reflection coefficients k_m by

$$K_m = 1 - \frac{1+K_m^+}{1-K_{m-1}} \tag{54}$$

This algorithm, which provides the same information with the Durbin algorithm requires $0.75p^2$ multiplications, 2p divisions and $2p^2$ additions/subtractions.

This order recursive algorithm is given in the following table.

ORDER RECURSIVE ALGORITHM FOR $\underline{a}_p$
Input data: $r_0, r_1, \cdots, r_p$

Initialization: $a_1^+ = -2r_1/r_0$, $\gamma_1^+ = r_0 + r_1$ (55)

$$K_1 = -r_1/r_0, \quad \alpha_1 = r_0 - 2r_1^2/r_0 \tag{56}$$

For m=1 step 1 to p-1 do

$$\gamma_{m+1}^+ = r_0 + \underline{r}_m^t \underline{a}_m^+ + r_{m+1} \tag{57}$$

$$1+K_{m+1}^+ = \gamma_{m+1}^+ / \gamma_m^+ \tag{58}$$

$$K_{m+1} = 1 - (1+K_{m+1}^+) / (1+K_m) \tag{59}$$

$$\alpha_{m+1} = \alpha_m (1 - K_m^2) \tag{60}$$

$$\underline{a}_{m+1}^+ = \begin{bmatrix} \underline{a}_m^+ \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{a}_m^+ \end{bmatrix} - (1+K_{m+1}^+) \begin{bmatrix} 1 \\ \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} \tag{61}$$

$$\begin{bmatrix} \underline{a}_{m+1} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ \underline{a}_{m+1} \end{bmatrix} + \begin{bmatrix} \underline{a}_{m+1}^+ \\ 1 \end{bmatrix} - (1+K_{m+1}^+) \begin{bmatrix} 1 \\ \underline{a}_m^+ \\ 1 \end{bmatrix} \tag{62}$$

End of m loop

A similar algorithm based on the antisymmetric predictors $\underline{a}_m^-$ can be also derived. This scheme is discussed in [9], [10].

5. FIR WIENER FILTERING

In the case of FIR Wiener filtering, the optimum filter $\underline{c}_p$, defined by (6), can be obtained from either $\underline{a}_m^+$, $\underline{c}_m^+$ (or $\underline{a}_m^-$, $\underline{c}_m^-$) using the new algorithm described in the following tables. The derivation of these algorithms is given in [10].

ORDER RECURSIVE ALGORITHM FOR $\underline{c}_p$ VIA $\underline{a}_m^+$ AND $\underline{c}_m^+$

Input data: $r_0, r_1, \cdots, r_p$

$r_{zz}(0), d_1, d_2, \cdots, d_p$

Initialization:

$$a_{-1}^+ = 2 \ , \quad \gamma_1^+ = r_0 + r_1 \tag{63}$$

$$K_1 = -r_1/r_0 \ , \quad K_1^c = -d_1/r_0 \tag{64}$$

$$c_1^+ = 2K_1^c \ , \quad \alpha_1 = r_0 + 2K_1 r_1 \tag{65}$$

$$\gamma_1^L = d_1 \ , \quad \alpha_1^c = r_{zz}(0) + K_1^c d_1 \tag{66}$$

For m=1 step 1 to p-1 do

$$\underline{a}_m^+ = \begin{bmatrix} \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ \underline{a}_{m-1}^+ \end{bmatrix} - (1+K_m^+) \begin{bmatrix} 1 \\ \underline{a}_{m-2}^+ \\ 1 \end{bmatrix} \tag{67}$$

$$\gamma_{m+1}^+ = r_0 + \underline{r}_m^t \, \underline{a}_m^+ + r_{m+1} \tag{68}$$

$$1+K_{m+1}^+ = \gamma_{m+1}^+ / \gamma_m^+ \tag{69}$$

$$K_{m+1} = 1 - (1+K_{m+1}^+) / (1 + K_m) \tag{70}$$

$$\gamma_{m+1}^L = d_{m+1} + \underline{r}_m^t \, \underline{c}_m^+ \tag{71}$$

$$K_{m+1}^L = (\gamma_{m+1}^L - \gamma_m^L) / \gamma_m^+ \tag{72}$$

$$\underline{c}_{m+1}^+ = \begin{bmatrix} \underline{c}_m^+ \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \underline{c}_m^+ \end{bmatrix} - \begin{bmatrix} 0 \\ \underline{c}_{m-1}^+ \\ 0 \end{bmatrix} - K_{m+1}^L \begin{bmatrix} 1 \\ \underline{a}_{m-1}^+ \\ 1 \end{bmatrix} \tag{73}$$

$$K_{m+1}^c = (K_m^c - K_{m+1}^L) / (1 + K_m) \tag{74}$$

$$\alpha_{m+1} = \alpha_m (1 - K_{m+1}^2) \tag{75}$$

$$\alpha_{m+1}^c = \alpha_m^c - (K_{m+1}^c)^2 \alpha_{m+1} \tag{76}$$

$$\begin{bmatrix} \underline{c}_{m+1} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \underline{c}_{m+1} \end{bmatrix} + \begin{bmatrix} \underline{c}_{m+1}^+ \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ \underline{c}_m^+ \\ 0 \end{bmatrix} - K_{m+1}^c \begin{bmatrix} 1 \\ \underline{a}_m^+ \\ 1 \end{bmatrix} \tag{77}$$

End of m loop

This order recursive algorithm requires approximately $1.25p^2$ multiplications, $2p$ divisions, and $3.25p^2$ additions/subtractions. If not all the information given by this algorithm are needed, this algorithm can be modified accordingly and save some computations.

A similar algorithm based on the antisymmetric counterparts can also be derived. This scheme is discussed in [10].

6. CONCLUSIONS

This paper has dealt with efficient algorithms for the estimation of linear predictors and FIR Wiener filters having either linear or arbitrary phase characteristics. Both existing and new algorithms were discussed. By necessity the discussion was short and limited to serial algorithms. More details as well as additional issues are investigated in a forthcoming paper[10].

ACKNOWLEDGEMENT

This work was partially supported by NSF Grant ECS-8507430.

REFERENCES

[1] L. R. Rabiner and B. Gold, Theory and Application of Digital Signal processing, Prentice-Hall, New Jersey, 1975.

[2] D. Manolakis, G. Carayannis, and N. Kalouptsidis, IEEE Trans. on CAS, Vol. 31, No. 11, pp. 974-978, Nov. 1984.

[3] B. Friedlander and M. Morf, IEEE Trans. on ASSP, Vol. 30, No.3, pp.381-390, June 1982.

[4] D.C. Farden, ACM Trans. Math. Software, Vol. 3, pp. 159-163, June 1977.

[5] D. Manolakis, N. Kalouptsidis and G. Carayannis, Eletronics Letters, Vol. 18, No. 10, pp. 429-431, 13th May 1982.

[6] F. Itakura and N. Sugamura, Proceedings of Speech Study Group of the Acousticcal Society of Japan, S79-46, NOv. 1979.

[7] F.K. Soong and B.H. Juang, in Proc. 1984 IEEE ICASSP, paper 1.10, San Diego.

[8] Y. Bistritz, IEEE Trans. Circuits and Systems, Vol. CAS-30, pp. 917-919, 1983.

[9] P. Delsarte and Yves Genin, "The split Levinson algorithm", IEEE Trans. on ASSP, 1986.

[10] An-Loong Kok and D. Manolakis, "Serial and parallel algorithms for the solution of special Toeplitz systems", in preparation.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

APPLICATION OF OS ESTIMATORS TO SONAR SIGNAL DETECTION

Prof. K.M. Wong and S. Chen

Department of Electrical and Computer Engineering
McMaster University
Hamilton, Ontario, Canada
L8S 4L7

The detection of a narrowband signal in a sonar environment with spatially uncorrelated white noise depends very much on the accuracy in estimating the noise power so that a threshold can be correctly fixed. The conventional MA method presents a large bias when operated in the environment where interfering signals exist in the neighbourhood. Various types of nonlinear methods based on the use of order statistics are introduced and analyzed and are found to be very much more robust.

1. INTRODUCTION

A common arrangement of the sensors in a passive sonar system is in the form of a linear towed array. Often, the signals received by the sensors in the array are then passed through an FFT analyzer and beamformer which determine the frequency contents as well as directional features of the signals. In the passive sonar system considered here, there are P sensors uniformly spaced along a straight line. Let the output signal of the pth sensor be denoted by $x_p(nT)$, then because of the delay involved in receiving the signal from sensor to sensor, the information of the arrival angle θ of the signal is contained in the beam which is formed by summing the signals from different sensors together. Furthermore, the power spectrum associated with the signal can be estimated by partitioning $x_p(nT)$ into M segments each denoted by $x_{pm}(nT)$, and then averaging the magnitude square of the Fourier coefficients over M segments. M is sometimes referred to as the "time-bandwidth product". Thus, the output of the FFT spectrum analyzer and beamformer can be written as

$$Z_x(k\Omega, \theta, M) = \frac{1}{M} \sum_{m=1}^{M} \sum_{p=1}^{P} \sum_{q=1}^{P} X_{pm}(k\Omega) \cdot X^*_{qm}(k\Omega) \cdot e^{-j(p-q)\frac{k\Omega}{c} d \sin\theta} \qquad (1)$$

where $X_{pm}(k\Omega)$ represents the DFT of $x_{pm}(nT)$, c is the propagation velocity of sound in water, and d is the separation between sensors. If the signals from the sensors contain only zero-mean Gaussian noise v(nT) which are uncorrelated from sensor to sensor and are of unity noise power, then it can readily be shown that the output Z_v has a probability density function (PDF) given by [1,2]

$$f_{Z_v}(z) = \begin{cases} \dfrac{M^M}{(M-1)!} z^{M-1} e^{-Mz} & \text{for } z \geq 0 \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

On the other hand, if a narrowband Gaussian signal mixed with uncorrelated Gaussian noise is received, the PDF of the output Z_x can be shown to be [1,2]

$$f_{Z_x}(z) = \begin{cases} \dfrac{\{M/(1+\rho)\}^M}{(M-1)!} z^{M-1} e^{-Mz/(1+\rho)} & \text{for } z \geq 0 \\ 0 & \text{otherwise} \end{cases} \qquad (3)$$

where ρ is the signal-to-noise ratio (SNR).

One of the functions of a sonar signal processing system is to detect the presence of a narrowband signal in a particular frequency bin and a particular beam in the output Z_x. To carry out the detection process, a threhold level Z_T is chosen such that

$$Z_T = (1 + r)\hat{Z}_v \qquad (4)$$

where $\hat{Z}_v$ is the estimate of the mean power of the noise in the frequency bin and r is a constant ratio. To decide if a signal is present, we examine the sample $Z_x(k\Omega)$ along a beam and compare it to this threshold. Using Eq. (4) we arrived at the decision rule [3],

$$\frac{Z_x(k\Omega)}{\hat{Z}_v} \underset{H_0}{\overset{H_1}{\lessgtr}} (1+r) \qquad (5)$$

where H_0 denotes the hypothesis that only noise is present and H_1 denotes that a signal is present. Since $\hat{Z}_v$ is an estimate, it is a random variable which depends on the data as well as on the method of estimation. The choice of the ratio r thus depends on what the specified values of probability of false alarm P_{FA} and the probability of detection P_D are. The average values of these probabilities are given by [1,2]

$$E[P_{FA}] = \int_0^\infty \left[\int_{(1+r)\hat{z}_v}^\infty f(z|H_0)\, dz \right] f_{\hat{Z}_v}(\hat{z}_v)\, d\hat{z}_v \qquad (6)$$

and

$$E[P_D] = \int_0^\infty \left[\int_{(1+r)\hat{z}_v}^\infty f(z|H_1)\, dz \right] f_{\hat{Z}_v}(\hat{z}_v)\, d\hat{z}_v \qquad (7)$$

where $f(z|H_0)$ and $f(z|H_1)$ in the system we considered are given by Eqs. (2) and (3), respectively, $f_{\hat{Z}_v}(\hat{z}_v)$ is the PDF of the mean noise power estimate and depends on the data and

the estimator used. In general, we fix a specified maximum value of $E[P_{FA}]$ so that the ratio r is calculated from Eq. (6). Using this threshold in Eq. (7), with a particular SNR ρ, the value $E[P_D]$ can be evaluated.

The most common methods of estimating the mean noise power utilize the principle of averaging, which is efficient and unbiased if the data contain noise only. However, if signals are present in the neighbouring frequency bins, the method of averaging suffers severe bias. To improve the estimation of mean noise power, a family of nonlinear estimators based on the use of order statistics (OS) are considered here. Their use in noise estimation with or without interfering signals is discussed and the PDF of their estimates are developed and compared.

2. METHODS OF NOISE POWER ESTIMATION

In this section, various methods of estimating noise power are introduced and their essential properties are discussed. Each method is operated along the frequency axis of the output $Z_x(k\Omega, \theta)$ so that an estimate of the noise $\hat{Z}_v$ is obtained.

2.1 Moving Average Filter

The moving average (MA) filter is a simple linear processor the input-output relationship of which is given by

$$\hat{Z}_X(k) = \sum_{\ell=-L}^{L} a_\ell Z_X(k+\ell) \qquad k = \ldots, -1, 0, 1, \ldots \tag{8}$$

where $Z_x(k)$ and $\hat{Z}_x(k)$ are the input and output sequences respectively; (2L+1), L being an integer, is called the window size; and a_ℓ are constants usually chosen such that $a_{-L} = a_{-L+1} = \ldots = a_{L-1} = a_L = 1/(2L+1)$. If the input samples contain noise only, the MA filter is optimum in the sense of unbiasness in estimating the mean noise power. However, in detecting a signal in a sonar environment, the input samples often include signal data, and, when these signal samples fall within the window of the MA filter, the noise power estimation will be biased.

In order to improve on the estimation of the noise power, the split window moving average (SWMA) filter is often employed in place of the MA filter. This method assumes that the frequency bins in the neighbourhood of the bin of interest contain noise only. Thus, the contents of the L-sample windows on either side of kΩ are averaged. The input-output relationship of the SWMA filter is the same as in Eq. (8). The constants a_ℓ are normally chosen such that

$$a_{-L} = a_{-L+1} = a_{-1} = a_1 = \ldots = a_L = \frac{1}{2L}$$
$$a_0 = 0. \tag{9}$$

The use of SWMA filter in the estimation of noise power in the frequency bin of kΩ is, assuming the neighbouring frequency bin contain noise only, in effect, taking the average of 2L random samples each assumed to have a chi-square distribution of 2M degrees of freedom as given by Eq. (2). The result is a random variable $\hat{Z}_v$ which is of a chi-square distribution of 4 LM degrees of freedom [4], i.e.

$$f_{\hat{Z}_v}(\hat{z}_v, 0) = \begin{cases} \dfrac{(2LM)^{2LM}}{(2LM-1)!}\, \hat{z}_v^{2LM}\, e^{-2LM\hat{z}_v} & \hat{z}_v \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

The notation we use here is such that $f^0_{\hat{Z}_v}(\hat{z}_v, 0)$ denotes the probability density function of $\hat{Z}_v$ when there is noise only. The zero indicates that there is no signal in the neighbouring 2L samples.

The SWMA filter offers an improvement on MA filter since if there is a signal in the frequency bin kΩ, it will not be taken into account ($a_0 = 0$) for the averaging. The probability density function of the estimated noise power will remain the same as given by Eq. (10) and therefore maintains an unbiased estimation of the average noise power as long as the neighbouring 2L frequency bins contain noise only. However, when other interfering signals fall within the neighbouring 2L frequency bins, the SWMA estimate of the mean noise power will be biased due to the presence of these signals. The PDF under the condition of having interfering signal is complicated since it depends not only on the number of interfering signals but also on the SNR of the signals. Therefore we employ the Edgeworth expansion [14] to approximate the PDF of the mean noise power estimate $f_{\hat{Z}_v}(\hat{z}_v)$.

2.2 Median Filter

Median (MD) filtering is a nonlinear signal processing technique first suggested by Tukey [5]. It possesses certain properties which often render it more attractive to use than a linear filter [6-12]. The median of 2L + 1 samples denoted by $\mathcal{M}_{2L+1}[\{Z_x(k+\ell);\ -L \leq \ell \leq L\}]$, is the (L + 1)th largest number of the set $\{Z_x(k+\ell);\ -L \leq \ell \leq L\}$. If a median filter of window size 2L + 1 is applied to estimate the noise power in the frequency bin of kΩ, then the result is given by

$$\hat{Z}_v(k\Omega) = \mathcal{M}_{2L+1}[\{Z_x((k+\ell)\Omega);\ -L \leq \ell \leq L\}] \tag{11}$$

If all the 2L + 1 frequency bins inside the window contain noise samples only, then the use of a median filter to estimate the noise power in the frequency bin of kΩ amounts to taking the median of 2L + 1 identically distributed random variables the probability density function $f_{Z_v}(z)$ of each is given by Eq. (2). The probability density function of the estimated power Z_v thus is given by [8,13]

$$f_{\hat{Z}_v}(\hat{z}_v, 0) = \frac{(2L+1)!}{L!\,L!} \left\{ f_{Z_v}(\hat{z}_v)\, F^L_{Z_v}(\hat{z}_v) [1 - F_{Z_v}(\hat{z}_v)]^L \right\} \tag{12}$$

where $F_{Z_v}(\hat{z}_v)$ is the cumulative distribution of Z_v.

When n signals fall within the filter window, the PDF of the estimate can be expressed as a "permanent" [8,14]. But, in general, the Edgeworth expansion is a very close approximation to the PDF.

2.3 Delta Trimmed (ΔT) Filter

The median filter is a special case of a more general class called order statistic (OS) filters [15]. The output of an OS filter of length 2L + 1 operating on a sequence $\{x_k\}$ is given by

$$y_k = \mathcal{O}\{\{x_\ell\}, k - L \le \ell \le k + L\}$$

$$= \sum_{i=1}^{2L+1} a_i x_{(i)}(k) \qquad (13)$$

where $\{x_{(i)}(k)\}$ is the ordered sequence of the set $\{x_\ell\}$ such that

$$x_{(1)}(k) \le x_{(2)}(k) \le \ldots \le x_{(2L+1)}(k) \qquad (14)$$

and a_i are constants that may be chosen for a particular application.

An important special case of the OS filter is the delta-trimmed (ΔT) filter [16] in which the constants a_i in Eq. (13) are chosen such that

$$a_i = \begin{cases} 1/\{2L+1-\lfloor\Delta_1(2L+1)\rfloor-\lfloor\Delta_2(2L+1)\rfloor\}; \\ 0; \end{cases}$$

$$\lfloor\Delta_1(2L+1)\rfloor+1 \le i \le 2L+1-\lfloor\Delta_2(2L+1)\rfloor \qquad (15)$$

otherwise

where $\lfloor u \rfloor$ denotes the greatest integer smaller than or equal to u, Δ_1 and Δ_2 are the fractions of the window size trimmed from the low and high ends, repsectively, of the ordered set. In our study, we choose $0 \le \Delta_1, \Delta_2 \le 0.5$. When $\Delta_1 = \Delta_2 = 0.5$, we have a median filter, whereas when $\Delta_1 = \Delta_2 = 0$, an MA filter results. In applying the ΔT filter in our case of estimating noise power, we normally choose Δ_2 so that the filter can remove the number of interfering signals that are likely to fall within the filter window

If the ΔT filter is applied to estimate noise power when only noise samples are present, then it is equivalent to taking the average of a reduced population of samples. The size of the sample population is given by

$$L_T = 2L+1 - \lfloor\Delta_1(2L+1)\rfloor - \lfloor\Delta_2(2L+1)\rfloor \qquad (16)$$

Each of the sample has a probability density function given by Eq. (2). However, since the samples are ordered, they are not independent any more. In our particular application, we choose $\Delta_1 = \Delta_2 = \Delta$.

The PDF of the estimate produced by a ΔT filter with or without interfering signals can again be approximated by the Edgeworth expansion.

2.4 Generalized Residual (GR) Filters

Although median and ΔT filtering can provide a stable noise estimate even when interfering signals fall into the filter window, in the case that only noise samples are present, the variance of the estimated power is greater than that obtained by a moving average filter. Furthermore, a bias term would also be present in the estimated noise power of the two OS filters. In order to provide adequate smoothing for noise power estimation while maintaining the desirable immunity to the presence of outliers, a linear estimator is appended to a median estimator resulting in an MD-MA filter. This idea was first proposed by Tukey [5] and later extended to a two-staged structure by Rabiner et al. [6]. Here we introduce a more general two-staged configuration which includes the use of a ΔT filter as well as an MD filter as shown in Fig. 1. In addition, a scaler C has been inserted at the output of the second stage. Note that when C = 0, this structure is reduced to a single-staged OS-MA filter. Such a structure, as shown in Fig. 1, is designated a generalized residual (GR) filter When a median filter is used as an OS filter in Fig. 1, we call the structure a Tukey filter; whereas when a ΔT filter is used in the place of an OS filter, we call it a ΔT-Tukey filter. The reasoning behind using a Tukey filter or a ΔT-Tukey filter for the estimation of noise power can be explained qualitatively as follows:

The input noise power samples $Z_v(k)$ contain the average noise power term as well as a random term. After passing $Z_v(k)$ through the first stage of OS-MA combination we obtain $\hat{Z}_{v1}(k)$ which still contains the average noise power information, whereas the random component has been greatly reduced in its variance but in addition, a bias component has been introduced. When $\hat{Z}_{v1}(k)$ is subtracted from the original sequence $Z_v(k)$, the resulting sequence Y(k) consists of largely the negative of this bias component as well as a random component which contains predominantly the original random component in $Z_v(k)$. When this sequence Y(k) is passed through the second stage of OS-MA combination yielding $\hat{Y}_1(k)$, the negative bias component is further biased and the random component in $\hat{Y}_1(k)$ is rather similar to that contained in $\hat{Z}_{v1}(k)$. The multiplication by a constant C scales the squence $\hat{Y}_1(k)$ so that the bias component or the random component contained in the final output sequence $\hat{Z}_v(k)$ can be cancelled out. Now, since the bias component in $\hat{Y}_1(k)$ is opposite in sign to that contained in $\hat{Z}_{v1}(k)$, in order to cancel the bias at the final output, we should add $\hat{Y}_1(k)$ to the delayed version of $\hat{Z}_{v1}(k)$. On the other hand, since the random component contained in $\hat{Y}_1(k)$ is rather similar to that in $\hat{Z}_{v1}(k)$, we should subtract $\hat{Y}_1(k)$ from $\hat{Z}_{v1}(k)$ to cancel the random component. These two requirements are incompatible. In our application, we find that the random component is the predominant factor and thus a negative sign is chosen. This fact will be evident from the results presented later. Also, in our numerous simulation examples [2], we find that a suitable value for the scalar is C = 2 for best cancellation of the random component in the ouput. Thus in the ensuing discussions, we assume all GR filters to have the scalar C = 2, and a negative sign feeding the lower branch sequence to the output adder as shown in Fig. 1.

The exact expression of the probability density function for the output $\hat{Z}_v$ in the case of a GR filter is difficult to find due to the correlation of samples introduced by the OS filters in the two stages. However, using the Edgeworth expansion, the probability density function $f_{\hat{Z}_v}(\hat{z}_v, n)$ can be estimated.

Probability density functions for other complex structures of filters with or without signals within the filter windows are similarly obtained.

3. COMPARISON OF THE VARIOUS METHODS OF NOISE POWER ESTIMATION

To compare the various estimators, we first evaluate the probability density function $f_{\hat{Z}_v}(\hat{z}_v, 0)$ yielded by each of the estimators under the condition of noise only. Here, the "time-bandwidth product", M, is set to be 50 and the window size, 2L + 1, for all the OS and MA filter parts is set to be 5. The input noise samples to the estimators have unity mean and variance $\sigma_v^2 = 0.02$. This comparison of the estimators

is shown in the left half of Table I, where the mean, the bias, the variance, and the mean-square error of the various estimates are tabulated.

With $M = 50$, $2L + 1 = 5$, and one interfering signal (SNR $3.7 \leq \rho \leq 13.2$) falling within the window of the filters, the probability density functions $f_{\hat{Z}_v}(\hat{z}_v, 1)$ for the SWMA and the nonlinear estimators are evaluated. The parameters of comparison for the various estimators in this case are tabulated on the right half of Table I.

Noise Only				
Type of Estimator	Estimated Mean Z_v	Bias β	Variance σ^2	M.S.E. $\varepsilon^2 = \beta^2 + \sigma^2$
SWMA	1.0	0.0	4.0×10^{-3}	4.0×10^{-3}
MD	0.9953	.0047	5.6×10^{-3}	5.62×10^{-3}
ΔT	0.9953	.0047	4.2×10^{-3}	4.22×10^{-3}
MD-MA	0.9942	.0058	3.2×10^{-3}	3.23×10^{-3}
ΔT-MA	0.9953	.0047	2.8×10^{-3}	2.82×10^{-3}
Tukey	0.9935	.0065	2.3×10^{-3}	2.34×10^{-3}
ΔT-Tukey	0.9949	.0051	1.7×10^{-3}	1.73×10^{-3}

One Interfering Signal within the Window of the Filter				
Type of Estimator	Estimated Mean Z_v	Bias β	Variance σ^2	M.S.E. $\varepsilon^2 = \beta^2 + \sigma^2$
SWMA	1.9986	-0.9986	0.6698	1.667
MD	1.0359	-0.0359	-0.0359	0.0076
ΔT	1.0448	-0.0448	0.0060	0.0080
MD-MA	1.0359	-0.0359	0.0047	0.0060
ΔT-MA	1.0448	-0.0448	0.0041	0.0061
Tukey	1.0418	-0.0418	0.0036	0.0053
ΔT-Tukey	1.0510	-0.0510	0.0026	0.0052

Table I. Comparison of PDF of various estimators under the conditions of noise only and with one interfering signal.

From Table I, it can be observed that when there is noise only, all nonlinear filters introduce a small bias while the SWMA is unbiased. However, as far as the variance is concerned, SWMA filter is surpassed in performance by all nonlinear filters but the MD and the ΔT. To assess the performances better, we compare the mean-square errors (MSE) of the estimation yielded by the various filters. Since the bias introduced by the nonlinear filters are relatively small, the MSE is predominantly due to the variance. Here, the heuristic explanation of the operation of the GR filter presented earlier can be confirmed. The Tukey and the ΔT-Tukey filters in Table I employ a subtraction of the lower branch signal from the upper branch signal at their outputs. It can be observed that although the biases in these two structures have been increased from their corresponding one-staged structures, the variances have been reduced. Since the variance is the predominant factor in this case, the MSE yielded by these two GR filters are lower.

In the case of one signal falling within the window of the filters, the situation changes dramatically. The overall PDF of the output of the SWMA filter has a large bias. This bias depends on the strength of the interfering signal, and in this case when the signal-to-noise ratio is in the range of $3.7 \leq \rho_I \leq 13.2$, the average bias turns out to be almost as large as the mean noise power. Furthermore, the variance has increased by over a hundred folds. On the other hand, the biases and variances exhibited by the nonlinear filters, although increased, are still within reasonably low level. The MSE of the SWMA filter is, as a result of the interfering signal, almost two hundred times that of the MD filter and more than three hundred times that of the ΔT-Tukey filter. One other interesting observation of the performance of the nonlinear filters is that, as a result of an interfering signal, each bias has increased dramatically and is of the order of ten times the bias when there is noise only. Each variance also increases, but is only approximately 1.5 times the corresponding bias variance when there is no interfering signal. As a result the bias is no longer negligible and the variance is no longer the predominant factor in calculating MSE. This is even more pronounced in the case when there are more than one interfering signals falling within the filter window.

4. CONCLUSIONS

We have, in this paper, glanced through the problem of detecting a narrowband signal in a sonar system. We have seen that the estimation of noise power is an essential part to this process of detection. The problem of using a conventional MA estimator has been addressed. This problem is especially acute when the estimator is operating in the environment of having interfering signals in the neighbourhood of the desired frequency bin. A number of nonlinear estimators based on the use of order statistics have been introduced and applied to the estimation of noise power with and without the interfering signals in the neighbourhood. The PDF of the outputs of these estimators have been developed. Their biases, their variances and their mean square errors have been evaluated. It has been found that all these nonlinear estimators are very much more robust than the conventional MA estimators. Among these nonlinear devices, the best estimates come from the generalized recursive (GR) filters which give the smallest mean squared errors. The GR filters also provide us with the additional flexibility of choosing to cancel the bias or the variance, depending on which is anticipated to be the more dominant factor. Based on these observations, it can be inferred that these nonlinear estimators will yield a lower probability of false alarm and a higher probability of detection than the conventional MA filter. This inference is confirmed in another report [18].

REFERENCES

[1] Walker, R.S., "The Detection Performance of FFT Processor for Narrow-band Signals", DREA Tech. Memorandum, 82/8, Defence Research Establishment Atlantic, Dartmouth, NS, Feb. 1982.

[2] G.R. McMillen, and K.M. Wong, "Median Filter Data Normalization of Two-Dimensional Acoustics Data Analysis", McMaster University, CRL Internal Report Series, No. CRL-116, July 1983.

[3] Helstrom, C.W., Statistical Theory of Signal Detection (Pergamon Press, 1968).

[4] Pugachev, V.S., Probability Theory and Mathematical Statistics for Engineers (Pergamon Press, 1984).

[5] Tukey, J.W., "Nonlinear (nonsuperposable) Methods for Smoothing Data", Conf. Rec., 1974 EASCON, Washington, DC, p. 673.

[6] Rabiner, L.R., Sambur, M.R., and Schmidt, C.E., "Applications of a Nonlinear Smoothing Algorithm to Speech Processing", IEEE Trans. Acoustics, Speech, Signal Processing, Vol. ASSP-23, No. 6, pp. 552-557, Dec. 1975.

[7] Jayant, N.S., "Average- and Median-Based Smoothing Techniques for Improving Digital Speech Quality in the Presence of Transmission Errors", IEEE Trans. Comm., Vol. COM-24, No. 9, pp. 1043-1045, Sept. 1976.

[8] Ataman, E., Aatre, V.K., and Wong, K.M., "Some Statistical Properties of Median Filters", IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-29, pp. 1073-1075, Oct. 1981.

[9] Gallagher, Jr., N.C. and Wise, G.L., "A Theoretical Analysis of the Properties of Median Filters", IEEE Trans. Acoustics, Speech, Signal Processing, Vol. ASSP-29, o. 6, pp. 1136-1141, Dec, 1981.

[10] Kuhlmann, F. and Wise, G.L., "On Second Moment Properties of Median Filtered Sequence of Independent Data", IEEE Trans. Comm., Vol. COM-29, No. 9, Sept. 1981.

[11] Nodes, T.A., and Gallagher, Jr., N.C., "Median Filters: Some Modifications and Their Properties", IEEE Trans. Acoustics, Speech, Signal Processing, Vol. ASSP-30, No. 5, Oct. 1982.

[12] Hahn, K.J., and Wong, K.M., "Median Filtering of Cepstra", Proc., IEEE International Electrical and Electronic Conference, Toronto, pp. 352-355, Sept. 1983.

[13] Justusson, B.I., "Median Filtering: Statistical Properties", in "Two-Dimensional Digital Signal Processing II", Ed. T.S. Huang, Springer-Verlag, 1981.

[14] David, H.A., "Order Statistics", Wiley, New York, 1981.

[15] Bovik, A.C., Huang, T.S., and Munson, Jr., D.C., "A Generalization of Median Filtering Using Linear Combinations of Order Statistics", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-31, No. 6, Dec. 1983, pp. 1342-1350.

[16] J.B. Bednar, and T.L. Watt, "Alpha-Trimmed Means and their Relationship to Median Filters", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. ASSP-32, No. 1, Feb. 1984, pp. 145-153.

[17] Vaughan, R.J. and Venables, W.N., "Permanent Expressions for Order Statistic Densities", J. Royal Statistical Society, vol. 34, pp. 308-310, 1972.

[18] Wong, K.M. and S. Chen, "Detection of Narrowband Sonar Signals Using Order Statistical Filters", submitted to IEEE Trans. ASSP.

[19] diFranco, J.V. and Rubin, W.L., Radar Detection (Prentice-Hall, Englewood Cliffs, NJ, 1968).

[20] Whalen, A.D., Detection of Signal in Noise (Academic Press, New York, 1971).

[21] Fry, T.C., Probability and Its Engineering Uses, 2nd Ed. (VanNostrand, Princeton, New Jersey, 1965).

[22] Johnson, N.I. and Kotz, S., Continuous Univariate Distributions – 1 (Houghton Mifflin Co., Boston, 1970).

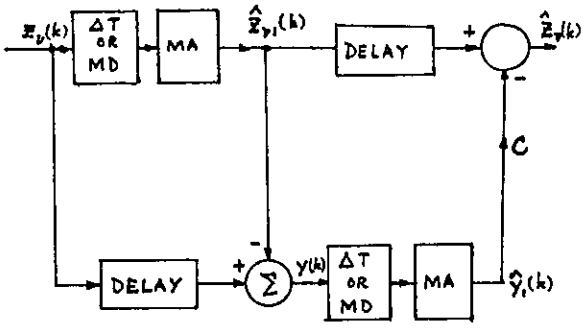


Fig. 1 The structure of a GR filter

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

NON-RECURSIVE METHODS FOR ON-LINE ESTIMATION OF THE FUNDAMENTAL WAVEFORMS OF SIGNALS USING KALMAN FILTER THEORY

Tadeusz ŁOBOS

Technical University of Wrocław, Poland*

Using Kalman filter theory, new nonrecursive algorithms for estimating the fundamental voltage and current waveforms from noise signals are developed and investigated. For non-harmonic random-noise, these procedures have better filter properties than Fourier algorithms especially with increase in filter order and sampling frequency. The use of the elaborated algorithms for protecting electric power systems has some advantages, owing to the convenient transient response time.

1. INTRODUCTION

In protection of electrical power systems serious problem arises to estimate the real fundamental components of voltages and currents from measured random signals with high noise levels. When a fault occurs, the power system is in transient state. A digital (computer) protection processes the input random signals, estimates real signals and detects a fault. Digital system must possibly faithfully reproduce the fundamental input signals, especially during transient states.
In recent years much research relating to on-line digital protection have been reported. Fast and efficient algorithms for on-line fault identification have been proposed. However, the known algorithms have some disadvantages. The most widely used Fourier technique [1...4] promises good accuracy only if it is assumed that the measured signals contain only harmonic components and are rather well suited for estimation under steady state conditions. In transient state, the fault-induced non-fundamental components of the signals depend on the location and type of faults which are random in nature. Owing to this reason the discrete Kalman filters will be more appropriate for these purposes. This filters have been widely used to estimate optimal states from noise measurements [5] . They are especially well suited to on-line digital computer implementation, because the noise measurements are processed recursively.
Several applications of Kalman filters in electrical power systems have been presented in recent years [6,7] . Under transient conditions following a fault occurence, Kalman filters are more suitable than Fourier ones, because they can estimate optimally the basic components of the voltage and current from the random noise signals. However, a Kalman filter used recursively has an infinite impulse response. The transient response time is therefore relatively long. This is one of the main difficulties which prevents wider application of these filters in power system relaying.

This paper introduces and investigates new non-recursive algorithms basic on the Kalman filter theory. The introduction bases on the two-state Kalman filter model for signals with periodical main components [8] and the three-state model for signals with exponential dc-component. Computer program for calculation these algorithms has been elaborated. The filter coefficients are calculated for the constant sampling frequency and sampling window.
For the examination of the filter properties of the designated methods the frequency charackteristics have been investigated. The transfer functions obtained from the new algorithms have been compared with the corresponding Fourier algorithms. An investigation program was carried out for various sampling windows up to two periods of the basic component, and for various sampling frequency up to 2 kHz.

2. KALMAN FILTER THEORY

The function of the Kalman filter can be described using the diagram in fig.1. The source generates a signal X(k) as a time function. It can be a voltage or current component of a transmission line. This signal is applied through the measurement instruments with the parameter C(k) to the signal channel and corrupted by an additive noise V(k). The Kalman filter is a computational method that processes measurement results to deduce a minimum error estimate of the state vector X(k).

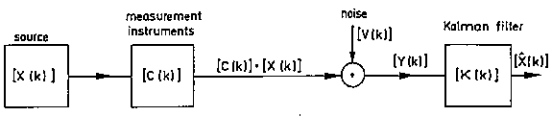


Fig.1. Signal estimation using a Kalman filter

* Present address: University of Erlangen-Nuremberg, Egerlandstr. 9, 8520 Erlangen, F.R.G.

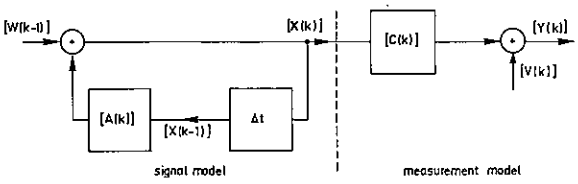


Fig.2. Model of the random signal process

In the derivation of a Kalman filter algorithm a signal and measurements model are assumed, as shown in fig.2. We assume that the random signal to be estimated can be modelled as a first -order recursive process driven by zero-mean white noise. Then we have

$$X(k) = A(k) \cdot X(k-1) + W(k-1) \qquad (1)$$

where
X(k) is the nx1 process state vector;
A(k) is the nxn state transition matrix;
W(k) is the nx1 noise vector corresponding to the noise in the real system.

The linear model of the measurement vector can be written as

$$Y(k) = C(k) \cdot X(k) + V(k) \qquad (2)$$

where
Y(k) is the mx1 measurement data vector;
C(k) is the mxn matrix representing the connection between the measurement and process state vectors, the so-called observation matrix;
V(k) is the additive noise or measurement error vector.

It is also assumed that the measurement noise and system noise are uncorrelated.
The system noise covariance matrix is given by
$Q(k) = E\{W(k) \cdot W(k)^T\}$
and the measurement noise covariance matrix by
$R(k) = E\{V(k) \cdot V(k)^T\}$
The general recursive Kalman filter equation is described as

$$\hat{X}(k) = A(k)\hat{X}(k-1) + K(k)\{Y(k) - C(k)A(k)\hat{X}(k-1)\} \qquad (3)$$

Note that here $\hat{X}(k)$ is the estimate of the signal X(k) at the time t_k, based on the previous estimate and only one data vector at time t_k has to be taken for calculation. The "best" estimate is specified as the estimate which minimizes the mean-square error of each signal component simultaneously.
The complete solution to the estimation problem is given by the equation for the Kalman filter gain in form of an nxm matrix.

$$K(k) = P_1(k)C(k)^T\{C(k)P_1(k)C(k)^T + R(k)\}^{-1} \qquad (4)$$

where

$$P_1(k) = A(k)P(k-1)A(k)^T + Q(k-1) \qquad (5)$$

The mean-square error covariance nxn matrix is given by

$$P(k) = P_1(k) - K(k)C(k)P_1(k) \qquad (6)$$

Using (4), (5) and (6) the alternative formula for the Kalman filter gain matrix, which will be used later, is obtained as

$$K(k) = P(k)C(k)^T R(k)^{-1} \qquad (7)$$

Substituting (7) in (6), the other formula for the error covariance matrix is obtained:

$$P(k)^{-1} = P(k-1)^{-1} + C(k)^T \cdot R(k)^{-1} \cdot C(k) \qquad (8)$$

3. OPTIMAL ESTIMATION OF THE FUNDAMENTAL WAVEFORMS OF SIGNALS

3.1. Two-state Model of the Signal

The waveforms of voltages and currents in electrical power systems may include a noise with a wide spectrum. The noise may be a combination of harmonics of the basic 50Hz waveforms and random noise due to faults and other disturbances.
Considering the Rayleigh distribution for the amplitude and the uniform distribution for the phase, the fundamental waveform of the signal may be described as

$$y(t) = X_a \sin(\omega_o t) + X_b \cos(\omega_o t) \qquad (9)$$

where X_a and X_b are independent, zero-mean, random variables with the noise variance δ^2.
The function of the Kalman filter is to estimate X_a and X_b in the presence of noise. For this purpose we need the two-state Kalman filter as an optimal estimator.The initial process state vector can be computed using two samples (at steps 0 and 1) of the waveform, which is assumed to be sinusoidal.For the further estimation the error covariance P(1) after the second sampling (step No.1) is assumed to be infinite.After the third sample we can determine the new estimated value for X_a and X_b.Generally, after the (k+1)th sampling step (from 0 to k) the measurement and state matrices are as follows:

$$C(k) = \begin{bmatrix} \sin[(k-1)\Delta] & \cos[(k-1)\Delta] \\ \sin(k\Delta) & \cos[(k\Delta)] \end{bmatrix} \qquad (10a)$$

$$Y(k) = \begin{bmatrix} y(k-1) \\ y(k) \end{bmatrix} \qquad (10b)$$

$$X(k) = \begin{bmatrix} X_a(k) \\ X_b(k) \end{bmatrix} \qquad (10c)$$

If we have (k+1) sampled values (from y(0) to y(k)) we can calculate the initial values $x_a(1)$ and $x_b(1)$ and next, after each sampling cycle, repeat the correction of the estimated values (k-1) times.
In this way we obtain the (k+1)th-order non-recursive algorithm for estimating the fundamental waveform from (k+1) sampling values of a noise signal.
Generally, this algorithms are as follows

$$X_a(k) = \sum_{i=0}^{k} K_{a,i}\, y(i) \qquad (11a)$$

$$X_b(k) = \sum_{i=0}^{k} K_{b,i}\, y(i) \qquad (11b)$$

The complex phasor of the basic waveform are given by

$$\underline{G} = X_b(k) - jX_a(k) = G_r + jG_i \qquad (12)$$

The time function can be found according to

$$y(t) = \mathrm{Re}\{\underline{G}\} \qquad (13)$$

3.2. Three-state Model of the Signal

In the practical investigations of short-circuit transients of currents, it was found that the exponential dc-component cannot be ignored. The component have to be included in the state model of the signal. The signal model may be described as

$$y(t) = X_a \sin(\omega_o t) + X_b \cos(\omega_o t) + X_c\, e^{-\frac{t}{T_g}} \qquad (14)$$

where T_g is the time constant of the exponential dc-component.
Such model leads to a three-state Kalman filter. The purpose of the Kalman filter will be to estimate X_a; X_b and eventually when needed X_c.
The time constant T_g is assumed to be known.
The initial process state vector can be computed using three samples (at steps 0, 1 and 2) of the waveform. Using (14) we getthe initial estimation of desired amplitudes X_a, X_b and X_c and they should be corrected in the next sampling steps (k=3,4...)
If we have (k+1) sampled values of the signal y(t) (from y(0) to y(k)) at first step we calculate initial estimated values $X_a(2)$, $X_b(2)$ and $X_c(2)$ and after each next sampling cycle k = 3,4,... we correct them (k-2) times. In this way we obtain the (k+1)th-order non-recursive algorithms. These algorithms are similar to two-state algorithms (see eqns 11...13) but the coefficients $K_{a,i}$ and $K_{b,i}$ take another values. In addition this model enables us to estimate initial value (for t = t_o) of the exponential dc-component according to formula

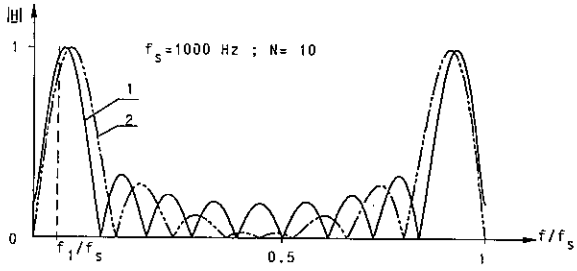
$$X_c(k) = \sum_{i=0}^{k} K_{c,i}\, y(i) \qquad (15)$$

4. INVESTIGATION OF THE ALGORITHMS

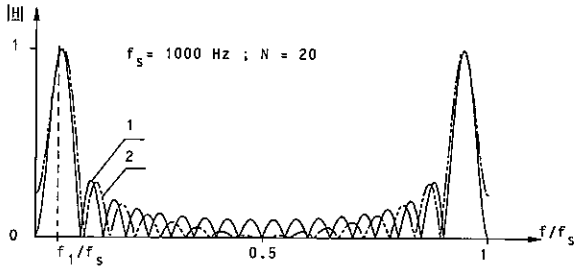
Using a computer program the coefficients $X_a(k)$ and $X_b(k)$ for two-state algorithms or respectively coefficients $X_a(k)$, $X_b(k)$ and $X_c(k)$ for three-states algorithms have been calculated.
In each case the sampling frequency and the sampling window are assumed to be constant.
For each algorithm the transfer function $\underline{H}(\omega)$ defined by relation

$$g_{out} = \underline{H}(\omega)\, g_{in}$$

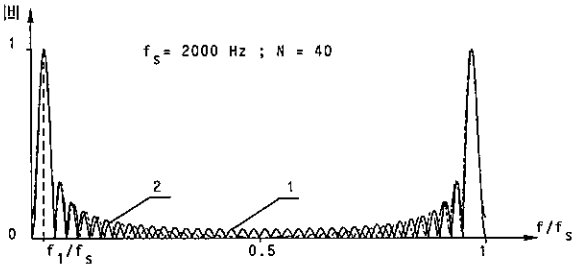
has been analysed. The transfer function obtained from the new non-recursive Kalman algorithms have been compared with the transfer function of the corresponding Fourier algorithm.
Fig.3a shows the magnitude spektrum for the sampling window equal to half a period of the 50Hz waveform. It is worth to note, that for f=0 the magnitude of this function is equal to zero. For the sampling frequency f_s = 1000Hz, the random signals of the frequency from 250Hz to 650Hz are suppressed better by Kalman algorithms than by Fourier ones.
Fig.3b and 3c shows the frequency characteristics for the sampling window equal to one periode and Fig.3d for two periods of the basic waveform. From these figures it follows, that the random signals of higher frequency are more suppressed by Kalman filter than by Fourier ones.The next Fig.4 shows the frequency characteristics of the three-state Kalman algorithms for estimating the fundamental waveforms. The coefficients of these algorithms have been calculated for the time constant T_g of the dc-component equal to 5; 10 and 25 ms. Owing to this reason the magnitudes of the



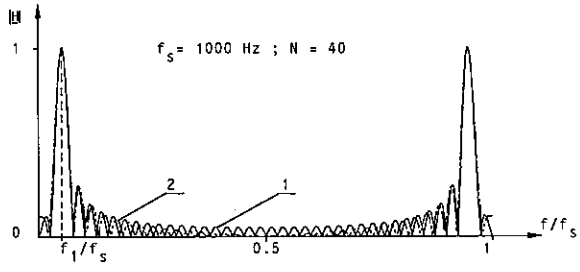
a)



b)



c)



d)

Fig.3 Magnitude spektrum of the transfer function. Curve 1, Fourier algorithm; 2, non-recursive Kalman algorithm. Sampling window equal to half a period (a), one period (b,c) and two period (d) of the 50Hz waveform. Sampling frequency f_s = 1000Hz (a,b,d) and f_s = 2000Hz (c).

transfer function for f=0 are greater than zero. It would be equal to zero for $T_g \to \infty$.
The transient response time of the Kalman algorithms is t shorter than that of the Fourier ones, because for all these algorithms K_o = 0

5. CONCLUSIONS

Until now, Kalman filters, as recursive optimal estimators have been used for estimating the fundamental waveform of voltage and current from the noise signal.
Using Kalman filter theory new nonrecursive algorithms are developed. The use of these algorithms for protecting electrical power systems has some advantages, because of their convenient transient response time. For non-harmonic random noise signals these procedures have much better filter properties than the Fourier algorithms, especially with an increase in filter order and sampling frequency. The properties of the new methods in the time domain are also better than those of the Fourier algorithms.

REFERENCES

[1] Ranjbar, A.M. and Cory, B.J., Algorithms for Distance Protection, IEE Conf.on Develop. in Power System Protection (London, 1975) pp.276-283.
[2] Phadke, A.G., Hlibka, T., Adamiak, M.G., Ibrahim, M. and Thorp, J.S., A Microcomputer Based Ultra-high-speed Distance Relay: Field Test, IEEE Trans., PAS-100, (1981) pp.2026-2036.
[3] Eichhorn, K.F. and Łobos, T., On-line Determination of Symmetrical Components by Sampling 8th Power Systems Computation Conference, (Helsinki 1984) pp.1155-1162.
[4] Hosemann, G. and Łobos, T., Ermittlung der symmetrischen Komponenten durch Abtastalgorithmen, Archiv f. Elektrotechnik 68 (1985) pp.1-16.
[5] Bozic, S.M., Digital and Kalman Filtering (Edward Arnold Ltd., London, 1979).
[6] Girgis, A.A. and Brown, R.G., Application of Kalman Filtering in Computer Relaying, IEEE Trans. PAS-100 (1981) pp.3387-3395.
[7] Dasgupta, K., Malik, O.P. and Hope, G.S., Digital Impedance Protection Using a Kalman Filter, 8th Power Systems Computation Conference, (Helsinki, 1984) pp.1135-1151.
[8] Łobos, T., Non-recursive Methods for On-line Estimation of Voltages or Currents and Symmetrical Components Using Kalman Filter Theory, Electric Power Systems Research 9 (1985) pp. 243-252.

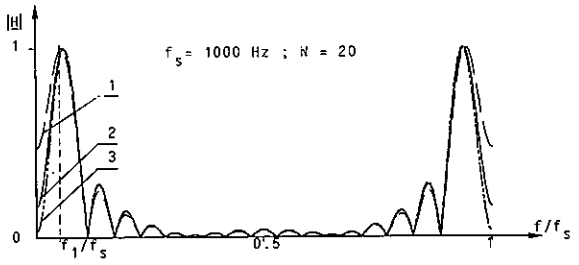


Fig.4 Magnitude spektrum of the transfer function of the three-state Kalman algorithm for estimating the basic waveform. Time constant T_g of the dc-component 1-5, 2-10 and 3-25ms. Sampling frequency f_s = 1000Hz. Sampling window equal to one period of the 50Hz waveform.

ACKNOWLEDGEMENTS

The author is especially grateful to Prof. G. Hosemann from the University of Erlangen-Nuremberg for his scientific support. The author also wishes to thank the Deutsche Forschungsgemeinschaft for the financial support of this project.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DISTORTION MEASUREMENTS IN S.C. CIRCUITS, USING L.M.S. FITTING ROUTINES.

Peter VAN PETEGHEM, Michiel STEYAERT, Willy SANSEN
Kath. Univ. Leuven, Dept. Elektrotechniek
Kard. Mercierlaan 94, B - 3030 Heverlee, BELGIUM

In this paper, a L.M.S. fitting technique is presented for the extraction of harmonic distortion. The estimates are derived from samples of the waveform. Therefore this method is better suited for discrete-time signal processing than standard distortion meters, which also take into account the transients between sampling times. The relationship is given between the A/D-Converter accuracy, and the accuracy of the T.H.D. estimates. Measurements on a CMOS low-distortion buffer, and a Switched-Capacitor filter are shown, allowing to compare the L.M.S. algorithm with standard distortion measurement techniques.

1. DISTORTION IN ANALOG CIRCUITS

Most analog signal processing circuits rely on building blocks which are **linear**. This means that the **response** y(t) is proportional to a given **excitation** x(t):

$$y(t) = C.x(t) \qquad (1)$$

However, no real building block is ever linear, i.e. the response is not exactly proportional to the excitation:

$$y = f(x) = a_1.x + a_2.x^2 + a_3.x^3 \ldots (2)$$

All phenomena which are associated with non-linear transfer characteristics, are contained in the notion **distortion**.

The most obvious effect of distortion is that the response of a pure sinusiod is not a pure sinusoid, but contains energy at integer multiples of the input frequency (called **harmonics**). In most applications, the amount of distortion is quantized by the energy at these harmonics.

The **distortion component HDN** is defined as the power at a frequency N times the input (fundamental) frequency, relative to the power at that fundamental frequency. The **Total Harmonic Distortion (T.H.D.)** is defined as the total R.M.S. value of the power in all harmonics, relative to the power in the fundamental. In general, both the values of HDN and T.H.D. are a function of the level and frequency of the signal.

If sampled-data analog circuits (like Switched Capacitor filters [6]), are compared with continuous-time circuits (like amplifiers or R.C. filters [5]), an essential difference is found. In **continuous-time** systems, the response of a system is valid for any time **t**, so distortion calculations should take into account any portion of the output waveform. In **sampled-data** circuits, the output is only valid at a given set of (equidistant) sampling points **t=n.T** (see fig.1). The output changes in steps; the way the output behaves in intermediate time intervals is irrelevant.

However, most commercially available distortion meters (like Hewlett-Packard's high-performance HP339A) do not make this difference: T.H.D. measurements of analog sampled-data circuits, using this type of equipment, include the transition between successive steps. **As a consequence, measured distortion is always too high.**
Therefore, it is an obvious strategy to deduce the T.H.D. figure of analog sampled-data circuits from a time-domain measurement: the set of data points {x(n)}, obtained by sampling at t=n.T.

A very commonly used approach for the determination of T.H.D. is looking for the dominant peaks in the **F.F.T.-transform** of that time series {x(n)}. Resolution of the harmonic components is limited by spectral

leakage due to windowing [7], and by a finite dynamic range in the sampling device (i.e. the number of bits) [4]. Resolution only can be enhanced by smoothing, which degrades speed, or which leads to more complex (i.e. more expensive) hardware.

Other techniques which have been mentioned in literature are **auto-regressive modelling**, and **adaptive filtering** [1]. Autoregressive modelling gives poor resolution of the harmonics, while adaptive Maximum-Likelihood or L.M.S. techniques yield very accurate harmonics estimates, but require a varying (and excessive) number of data points (3000 or more iterations for the retrieval of harmonics at the -50 dB level or lower).

2. LEAST-MEAN-SQUARES FITTING

Here a L.M.S. algorithm is presented [1] which acts on a fixed number of sampling points. The measured time series {x(n)} is matched in L.M.S. sense to a set of sinusoidal and cosinusoidal functions. The number of data points is N, the data have been sampled with B bits A/D-converter accuracy, and the number of distortion components to be retrieved is M.

$$y(n) = c + \sum_{m:1}^{M} [a(m).\sin(2.\pi.m.n.f_0.T_s + b(m).\cos(2.\pi.m.n.f_0.T_s)] \quad (3)$$

If we define the constant u:

$$u = 2.\pi.f_0.T_s \quad (4)$$

then, the objective function Q which has to be minimized is:

$$Q = \sum_{n:1}^{N} [x(n) - c - \sum_{m:1}^{M} a(m).\sin(n.m.u) - \sum_{m:1}^{M} b(m).\cos(n.m.u)]^2 \quad (5)$$

The weights c, a(m) and b(m) are found by differentiating Q to each of these parameters, and solving the resulting 2M+1 dimensional matrix equation:

$$\mathbf{R}.\mathbf{A} = \mathbf{Y} \quad (6)$$

In eq.(6) **R** is a square regular covariance matrix of order 2M+1. **A** is the estimated weights vector, and **Y** is the input covariance matrix. In [1] exact formulas are given for all matrix elements. The elements of **R** are only dependent on the parameters u, N and M, and **independent of the data itself**. It can be shown that **R** is regular and approximately diagonal. This means that eq.(6) is a numerically well-conditioned matrix equation. The values of the weights vector **A** can be found in an efficient way by solving eq.(6) using orthogonal techniques or triangular (i.e. LU) matrix decomposition techniques.

L.M.S. fitting leads to weight estimates that are unbiased and have minimum variance [3]. Quantization errors in the sampling device introduce estimation errors. The error on the component HDM is (for a 0.5 % confidence level):

$$e_{HDM} = 2,7313 \; . \; 2^{-B} \; . \; \sqrt{(M/N)} \quad (7)$$

In [1] it is shown that the presented L.M.S. fitting algorithm has a 15 to 20 dB better dynamic range than a F.F.T. peak-picking algorithm. Vice versa, this means that the L.M.S. technique allows to calculate reliable harmonics estimates with 2 to 3 bits lower A/D-converter resolution. On the other hand, the algorithm is sensitive to errors in the fundamental frequency f_0:

$$e_{HDM} = 0,25.(N+1).M.u.\Delta f_0/f_0 \quad (8)$$

It is advised to use equipment where fundamental and sampling frequency are derived from crystals.

In a lot of applications, T.H.D. is measured for a set of amplitude levels and a fixed f_0. Under these conditions the presented algorithm becomes extremely efficient. In an F.F.T. approach all calculations have to be redone for each measurement, which gives at least N*log(N) multiplications [3]. In the L.M.S. approach the matrix **R** will stay constant, so also its LU decomposition. Only **Y** has to be updated, requiring M*N multiplications. The weights vector **A** can be found by one single matrix multiplication, which requires only M*M multiplications. Therefore, for a low number of harmonics (M ≈ 3), the L.M.S. algorithm is faster than the F.F.T. algorithm.

The feasibility of the L.M.S. technique is shown by testing the algorithm with synthesized test signals (variables: distortion components HDi, and sampling accuracy B). The errors between estimated and input T.H.D. have been collected in fig.2 (black markers), together with the theoretically derived tolerance bound eq.(7) (solid line). It is seen that this bound is

relatively pessimistic, and that actual accuracy always is better. In fig.3 the F.F.T. of a test signal (using a Blackman-Harris window [7]) with 0.03 % T.H.D., sampled with a 10 bits A/D-converter, is shown. Harmonics are buried in the quantization noise. Nevertheless, the L.M.S. algorithm presented here still gives an adequate distortion estimate, as can be seen in Table 1.

3. DISCUSSION OF MEASUREMENTS

In this section the results of the L.M.S. estimation routine are compared with standard T.H.D. meters. All L.M.S. measurements are done with a Data General Data 6000 Digitizer (N=1024, B=14, M=3). Preliminary measurements showed that the distortion, contributed by the digitizer itself is 0.03 %.

In fig.4 and 5, measurements are shown on a low-distortion CMOS buffer. In fig.4, T.H.D. is given as a function of input level, and compared to a HP339A high-resolution Distortion Analyser. In fig.5, the components HD2 and HD3 are given as a function of frequency, and compared with measurements on a HP3562A Spectrum Analyser. In both figures the L.M.S. estimates are consistent with the control maesurements; this is obvious, as the buffer is a continuous-time circuit.

In fig.7, the components HD2 and HD3 (resulting from L.M.S. fitting) are shown for a 4th order elliptic Switched-Capacitor lowpass filter [2]. Its transfer function H(f) is given in fig.6. HD2 and HD3 have a bump in the transition range between passband and stopband. Measurements with the HP3562A Distortion Meter yield HD2=-46 dB and HD3=-60 dB in the passband. The L.M.S. algorithm yields for this sampled-data circuit HD2=-55 dB and HD3=-52 dB, which is, as been predicted, considerably lower. In Switched Capacitor circuits, the non-linearities that distort the sampled-data signal processing, have antisymmetrical characteristics [5,6]. This explains the higher HD3 in the L.M.S. measurements, than has been obtained by the HP3562A

4. CONCLUSIONS

A very flexible, low-cost, high-performance distortion algorithm has been presented, using the L.M.S. technique. This technique can be implemented very efficiently as a software packet on any desktop calculator incorporated in an automated test bench. No special-purpose expensive hardware equipment has to be installed. The presented technique is superior to existing software analysis methods as F.F.T. and autoregressive modelling methods, and is better adapted to discrete-time signal processing than standard T.H.D. measurement equipment.

REFERENCE LIST

[1] P. VAN PETEGHEM, P. VANDELOO, W. SANSEN, "Efficient L.M.S. Algorithm Performs Audio Distortion Analysis on Sampled Waveforms.", Pres. at the 75th Conv. of the A.E.S, 1984

[2] K. HALONEN, M. STEYAERT, W. SANSEN, "A Micropower 4th Order Elliptical Switched Capacitor Low-Pass Filter." Pres. at the IEEE C.I.C.C.conf., 1986

[3] M.B. PRIESTLEY, "Spectral Analysis and Time Series.", Academic Press, 1981

[4] A.V. OPPENHEIM, R.W. SCHAFER, "Digital Signal Processing.", Prentice-Hall, 1975

[5] P.R. GRAY, R.G. MEYER, " Analysis and Design of Analog Integrated Circuits.", Wiley, 1977

[6] R. GREGORIAN, K.W. MARTIN, G.C. TEMES, "Switched-Capacitor Circuit Design.", Proc. of IEEE, Vol.71(8), 1983

[7] F.J. HARRIS, "On the Use of Windows for Harmonic Analysis with the D.F.T." Proc. of IEEE, Vol.66(1), 1978

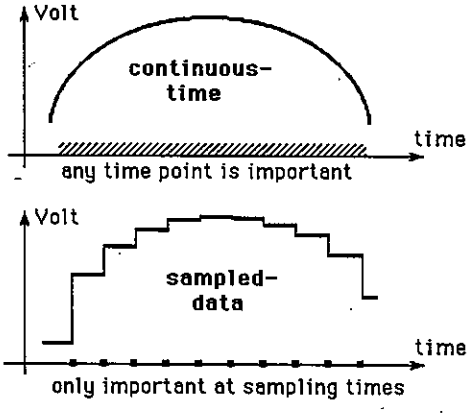


Fig.1: Comparison of continuous-time and sampled-data waveforms.

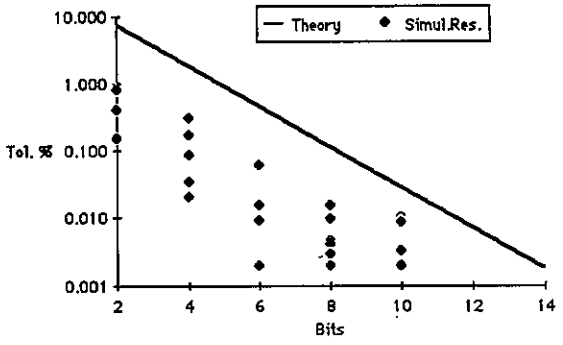


Fig.2: Test results of L.M.S. T.H.D. estimation algorithm, for synthesized test signals. (T.H.D.: 0.03 % to 10 %, B: 2 to 10 bits)

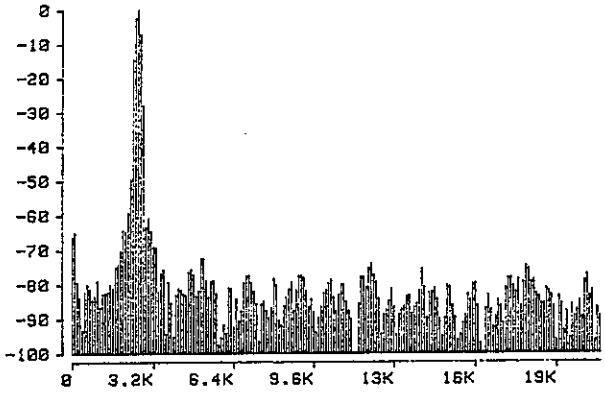


Fig.3: F.F.T. of test signal with T.H.D.=0.03% (B = 10, Blackman-Harris window)

i	a(i)	b(i)	HDi
1	0.031100	1.000000	0.00 dB
2	-0.000018	0.000257	-71.80 dB
3	0.000021	0.000038	-87.25 dB
4	0.000043	-0.000107	-78.78 dB
5	0.000034	-0.000059	-83.29 dB
6	-0.000022	-0.000055	-84.61 dB
Estimated T.H.D.: 0.0299 %			
Input T.H.D. : 0.0300 %			

Table 1: L.M.S. estimation results, from the same data as in fig.2.

Fig.6: Overall transfer function H(f) of 4th order elliptic Switched-Capacitor filter [2].

Fig.7: Measurement of HD2 and HD3 on S.C.filter in fig.6, as a function of freq..

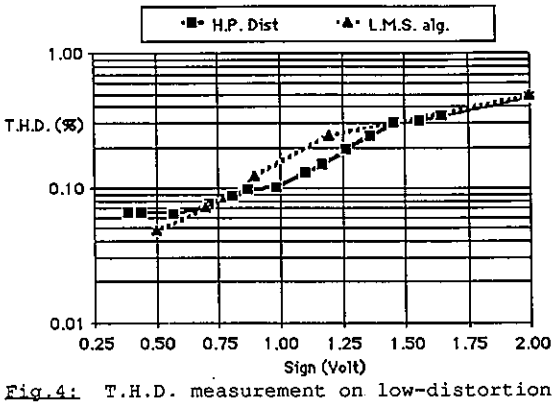


Fig.4: T.H.D. measurement on low-distortion CMOS buffer, as a function of input level. Comparison of L.M.S. algorithm with HP339A distortion analyzer.

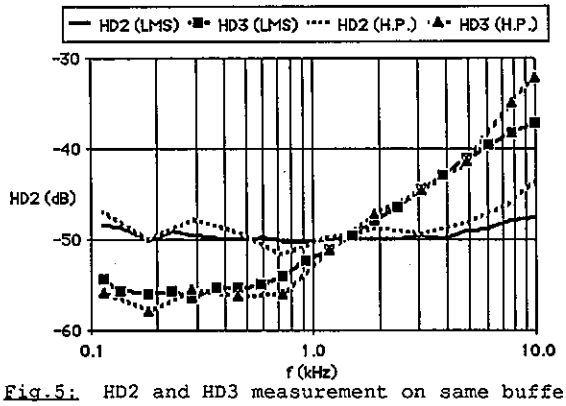
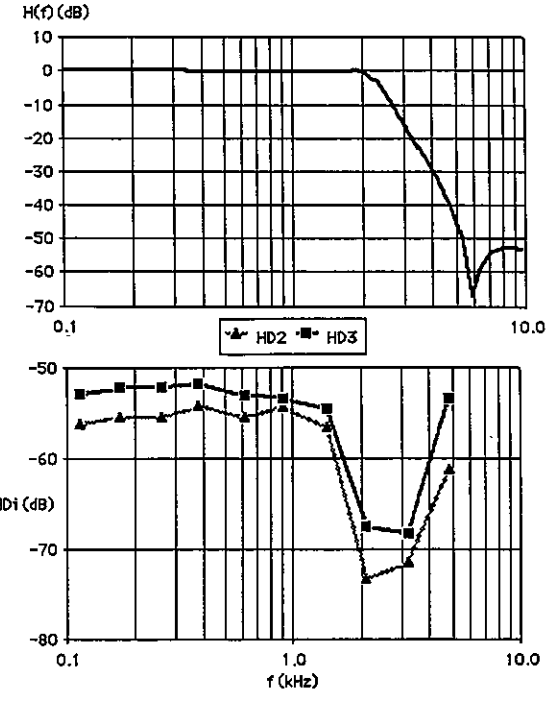


Fig.5: HD2 and HD3 measurement on same buffer as a function of frequency. Comparison of L.M.S. algorithm with HP3562A spectrum analyser. Input level: 2 V Ampl.



SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

FAST RECURSIVE/ITERATIVE TOEPLITZ EIGENSPACE DECOMPOSITION

A. A. (Louis) BEEX

Department of Electrical Engineering
Virginia Polytechnic Institute & State University
Blacksburg, VA 24061, USA

In narrowband array processing the information about the directions of the incoming signals can be extracted from the spatial correlation of the sensor signals. For an equi-spaced linear array of sensors this can be achieved by the eigenspace decomposition of the Hermitian Toeplitz correlation matrix. Here we describe a procedure to achieve that eigenspace decomposition recursively in order. This permits estimation of the angles of arrival for subsequent orders, facilitating early estimation of the number of sources. At each order a number of independent, structurally identical, nonlinear problems is solved in parallel, facilitating fast implementation.

1. INTRODUCTION

Recent approaches to array processing have concentrated heavily on eigenspace decomposition of the spatial correlation matrix of the sensor signals, as initiated by Schmidt [1] and Bienvenu [2]. For equi-spaced linear sensor arrays this correlation is often a Toeplitz matrix, so that it seems natural to try to exploit that special structure to achieve gains in the throughput of the resulting algorithm. We modify an algorithm reported by Gueguen [3], which had as its main shortcoming that it could possibly converge to the next higher eigenvalue rather than the minimal one. Our algorithm provides for the complete set of eigen values and eigen vectors of a Hermitian Toeplitz matrix, computed recursively in order, and consisting of a number of identical problems to be solved in parallel.

2. ARRAY PROCESSING

Assume a linear equi-spaced array of sensors with narrowband signals impinging on it. Under the assumption of nondispersive propagation, sensors without distortion, and envelope variations that are slow relative to the carrier frequencies of the narrowband signals, the received or measured vector of sensor signals is as follows:

$$(1)\quad x_t = s_t + n_t = s_t\, m_\theta + n_t$$

where s_t represents the narrowband signal, and the mode vector m_θ represents the delays with which the signal impinges on each of the sensors. The n-th element of the mode vector is

$$(2)\quad (m_\theta)_n = \exp(-j2\pi d \sin\theta/\Omega)$$

indicating that in general the mode vector is a nonlinear function of the signal arrival angle θ, center frequency Ω, array element spacing d, and sensor reponse. For reception of p simultaneous narrowband signals, the received vector becomes

$$(3)\quad x_t = \Sigma_{n\in[1,p]}\, s_{t,n}\, m_{\theta,n} + n_t$$

Under the assumption that the signals and the zero mean noise are uncorrelated, the spatial correlation of the measured sensor signals takes on the form of the matrix equation

$$(4)\quad R_x = M\, R_s\, M^H + \mu\, R_n$$

The matrix M has as its columns the vectors $m_{\theta,n}$, and as a result of (2) it is actually a Vandermonde matrix here. The signal covariance R_s is a diagonal matrix due to the assumption of uncorrelated sources, and the noise is assumed known, so that we might as well assume it to be white.

Research supported by the Center for Innovative Technology Grant #INF-85-020, Herndon VA, and Sperry Corporation, Reston VA.

For the problem considered here the matrices R_x, M R_s M^H, and μ R_n are Hermitian (Toeplitz). If the dimension of the correlation matrix is N, then the minimum eigenvalue of μ occurs N-p times, and the eigenspace approach is based on the fact that the space spanned by the minimal eigenvectors (the N-p dimensional noise subspace U_n) is orthogonal to the columns of M (the p dimensional signal subspace U_s). To find directions of arrival we need to find vectors of the form of m_θ which are orthogonal to the noise subspace, or alternatively, which are in the signal subspace. These methods correspond to minimizing and maximizing respectively the projection of m_θ onto the corresponding subspaces.

For our purposes then, we need to find the noise and signal subspaces, both spanned by eigenvectors, of the Hermitian Toeplitz matrix R_x.

3. EIGENSPACE DECOMPOSITION RELATIONS

Any Hermitian Toeplitz matrix can be written in the form of its spectral decomposition according to Franklin [4]

$$(5)\quad R_p = \Sigma_{n\in[0,p]}\ \mu_{n,p}u_{n,p}u^H_{n,p} = U_p\Sigma_pU^H_p$$

The n+1 st column of U_p is the n th normalized (unit norm) eigenvector $u_{n,p}$ of the Hermitian Toeplitz matrix with first row

$$[r_0\ r_1\ \ \dots\ \ r_p]$$

The matrix Σ_p is diagonal, and has as its elements the eigenvalues $\mu_{n,p}$, arranged in non decreasing order. This formulation results in a simple corresponding expression for the inverse

$$(6)\quad R^{-1}_p = U_p\Sigma^{-1}_pU^H_p$$

4. ORDER RECURSIVE EIGEN DECOMPOSITION

The question we like to answer is the following: Suppose we know the eigen decomposition for R_{p-1}, can we find from it the eigen decomposition for R_p? That is, find all eigenvalues and eigen vectors recursively in order.

To this end the following equation needs to be solved.

$$(7)\quad (R_p - \mu\ I)\ a = 0$$

Since we know the eigen decomposition for R_{p-1} equation (7) is rewritten as

$$\begin{bmatrix} r_0-\mu & \vdots & r^H \\ \cdots & \vdots & \cdots \\ r & \vdots & R_{p-1}-\mu I \end{bmatrix}\begin{bmatrix} 1 \\ \cdots \\ a^- \end{bmatrix} = 0$$

Consequently, the solution (μ,a^-) should be found to

$$(8a)\quad r_0-\mu + r^Ha^- = 0$$

$$(8b)\quad (R_{p-1}-\mu I)\ a^- = -r$$

The normalization of $a = [1\ a^{-H}]$ then gives a new order updated eigen value - eigen vector pair $(\mu_{n,p},u_{n,p})$.

First note that the matrix in (8b) is Hermitian Toeplitz, so that for any given μ the solution a^- can be found efficiently, using the Levinson algorithm. The remaining problem is to solve for all possible eigen values $\mu_{n,p}$ from (8a). To that end, substitute (8b) into (8a) to eliminate a^-.

$$(9)\quad r_0-\mu-r^H(R_{p-1}-\mu I)^{-1}r = 0$$

Using the known eigen decomposition of R_{p-1} leads to the equality

$$(10)\quad r_0-\mu=r^HU_{p-1}(\Sigma_{p-1}-\mu I)^{-1}U^H_{p-1}r = \Sigma_{n\in[0,p-1]}\ \beta^H_n\beta_n/(\mu_{n,p-1}-\mu)$$

where

$$\beta_n = r^H\ \mathrm{col}_{n+1}U_{p-1}$$

If we define the left hand side of (10) as $g(\mu)$, and the right hand side as $f(\mu)$, some interesting characteristics of these functions can be noted. The derivative $g'(\mu)$ equals -1. If μ approaches $\mu_{n,p-1}$ from below, then $f(\mu)$ approaches ∞; if μ approaches $\mu_{n,p-1}$ from above, then $f(\mu)$ approaches $-\infty$. If μ approaches $-\infty$ resp. ∞, then $f(\mu)$ approaches +0 resp. -0. The derivative of $f(\mu)$ is as follows

$$(11)\quad f'(\mu)=\Sigma_{n\in[0,p-1]}\ \beta^H\beta/(\mu_{n,p-1}-\mu)^2$$

which requires the same components as $f(\mu)$, and only one additional multiply, for its computation. From (11), note that $f'(\mu) > 0$ for $\mu\in(\mu_{n,p-1},\mu_{n+1,p-1})$, for all n. It is exactly in such intervals that we are looking for solutions, as a result of the inclusion principle [4] or interlacing property of eigen values.

$$(12)\quad \mu_{0,p}<\mu_{0,p-1}<\mu_{1,p}<\dots$$
$$\dots<\mu_{p-1,p}<\mu_{p-1,p-1}<\mu_{p,p}$$

Strict inequalities hold when the eigen values are distinct, whereas in general equalities may be possible.

Defining the function $F(\mu)$ as follows

$$(13)\quad F(\mu) = f(\mu) - g(\mu)$$

it is noted that from the previously iterated properties $F'(\mu) > 1$ for all μ in an interval $I_{n,p}=(\mu_{n-1,p-1},\mu_{n,p-1})$. Together with property (12), which says that there is exactly one solution $\mu_{n,p}$ that lies in interval $I_{n,p}$, this defuses any reason for not using Newton's method to solve $F(\mu) = 0$. With the definitions $I_{0,p}=(-\infty,\mu_{0,p-1})$, and $I_{p,p}=(\mu_{p-1,p-1},\infty)$ we have in general $\mu_{n,p}\in I_{n,p}$ for $n\in[0,p]$.

The full set of eigen values is then solved for in parallel, by solving

$$(14)\quad F(\mu)=0 \quad \mu\in I_{n,p} \quad n\in[0,p]$$

according to the following restricted Newton algorithm

$$(15)\quad \gamma_{k+1}=\mu_k-F(\mu_k)/F'(\mu_k)$$

$$\mu_{k+1}=\begin{cases}\gamma_{k+1} & \text{if } \gamma_{k+1}\in I_{n,p}\\ (\mu_k+\mu_{n,p})/2 & \text{if } \gamma_{k+1}>\mu_{n,p}\\ \quad \mu_k \text{ replaces } \mu_{n-1,p} & \\ (\mu_k+\mu_{n-1,p})/2 & \text{if } \gamma_{k+1}<\mu_{n-1,p}\\ \quad \mu_k \text{ replaces } \mu_{n,p} & \end{cases}$$

$$\text{stop if } |\mu_{k+1}-\mu_k|<\text{eps}$$

Note that when an iteration goes outside the restricted interval, since the gradient is larger than one, this indicates the direction in which the solution is to be found. Consequently the interval can be further restricted as indicated. Newton's method is hereby guaranteed to converge, and near the solution this convergence is quadratic. In the numerical example eps=10^{-4} was used, for which convergence occurred in about 8 iterations.

As part of the parallel problem, after finding a new eigen value $\mu_{n,p}$, the Levinson algorithm is used subsequently to solve (8b) for a^-. The normalization of a then results in having found the new eigen pair $(\mu_{n,p},u_{n,p})$. This eigen pair can be assessed by evaluating the norm of e.

$$(16)\quad e = (R_p-\mu_{n,p}I)u_{n,p}$$

The eigen pair assessment for the above algorithm was compared to results obtained from IMSL/EISPACK subroutines, which indicated that the performance of the recursive algorithm approached that of the batch algorithms in terms of numerical accuracy, orthogonality, etc. Since for the order recursive algorithm errors can accumulate, the tolerance parameter eps may have to be made smaller for higher order matrices. The latter would of course increase the number of iterations at each stage.

5. FAST PROJECTION EVALUATION

Finding directions of arrival in the context of our problem, requires evaluation of inner products

$$(17)\quad m_\theta{}^H u=\Sigma_{m\in[0,M-1]}\, u_{m+1}e^{-j\theta m}$$

$$=\text{DFT}(u_1,\ldots,u_M,0,\ldots,0)$$

if the angles of evaluation are chosen as $\theta_k=2\pi k/N$. The array dimension used is M, which is also the order of the correlation matrix decomposed. Efficient FFT or CZT algorithms can then be used, and the accuracy in resolving the maxima and/or minima of the projections onto signal and noise subspaces depends on the amount of zero padding. In our numerical example 1024 point FFT's were used, resulting in resolution of .35°.

6. PROBLEM AREAS

It becomes difficult to associate specific intervals with a particular eigenvalue in the case of multiple eigen values. This happens easily, for when the number of sensors exceeds the number of sources by more than one the minimal eigen value has multiplicity larger than one. Numerically however, this has not been a problem because the minimal eigen value is approximated to within the tolerance parameter eps. For subsequent stages this leads to a cluster of eigen values, closely spaced, and close to the true minimum eigen value. The real problem comes when solving for the associated eigen vectors using the Levinson algorithm. The first minimal eigen vector, which is in the noise subspace, is orthogonal to the signal subspace. The next minimal eigen vector is very close to the first one, and not orthogonal to it, as required. The orthogonality is evaluated, and the new vector is orthogonalized with respect to the signal subspace and already known vectors in the noise subspace. This procedure has led to very good results, even when using an ideal correlation matrix with known multiple minimal eigen values.

7. NUMERICAL EXAMPLE

An ideal correlation matrix was computed according to (4), with sources at 18° and 22°, for a signal to noise ratio of 10 dB. For matrix dimension 4 Figure 1a shows the maximization of the modevector projection onto the signal subspace $U_s=sp(u_2,u_3)$. In Figure 1b the minimization of the modevector projection onto the noise subspace $U_n=sp(u_0,u_1)$ is given. Figure 2 gives the signal subspace result for sources at 18° and 32°, 10 dB signal to noise ratio, and R_x estimated from 300 data points. In all these cases the source directions of arrival are found quite well.

8. CONCLUSIONS

An algorithm is presented that recursively in order finds the eigen decomposition for a Hermitian Toeplitz matrix. Convergence of the algorithm is guaranteed, and very reasonable results are obtained with fewer than 10 iterations of a restricted Newton algorithm for finding eigen values for the order increased matrix. Eigen vectors are found efficiently with the Levinson algorithm. The application to array sensor processing is illustrated, which uses FFT's for efficiency in finding directions of arrival from noise and/or signal subspace projections.

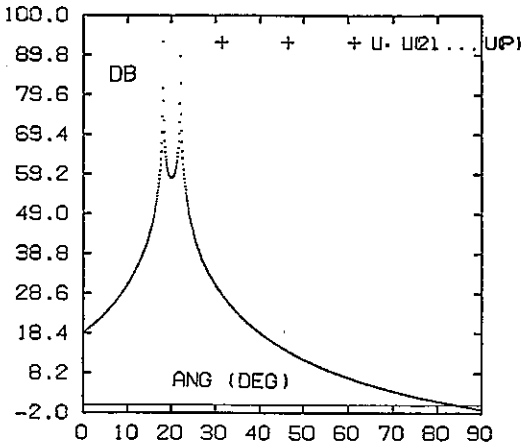


Fig. 1a. Signal Subspace Method for Known Correlation.

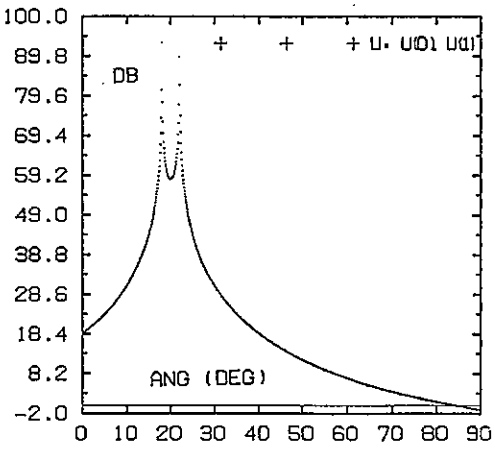


Fig. 1b. Noise Subspace Method for Known Correlation.

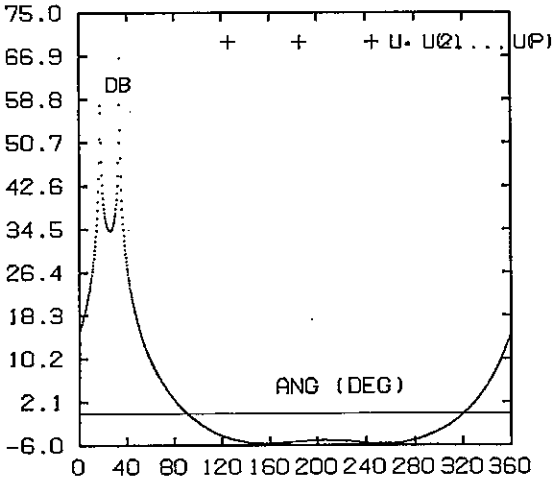


Fig. 2. Signal Subspace Method for Estimated Correlation from 300 Data.

REFERENCES

[1] Schmidt, R., Multiple Emitter location and Signal Parameter Estimation, RADC Spectrum Estimation Workshop, 1979.

[2] Bienvenu, G. and Kopp, L., Adaptivity to background noise spatial coherence for high resolution bearing estimation, ICASSP'80, Denver CO., 1980.

[3] Gianella, F. and Gueguen, C., Extraction des vecteurs propres de matrices de Toeplitz, GRETSI, Nice France, June 1981.

[4] Franklin, J.N., Matrix Theory, Prentice-Hall Inc., 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DYNAMIC BEHAVIOUR OF AN ADAPTIVE ARRAY ALGORITHM

C. Morisseau , C. Gallou *, F. Christophe

O.N.E.R.A. - BP 72 - 92322 Chatillon Cedex - France

This paper deals with the dynamic behaviour of an adaptive algorithm for optimizing the array output power. Firstly , a time decreasing loop gain providing improved convergence properties is proposed for stationary conditions. Then the influence of a non stationary environment (moving jammer, turbulent propagation medium) on the algorithm behaviour is analysed.

1. INTRODUCTION

An increasing number of communication or radar systems includes antenna array , a goal of which is to reduce the disturbances created by undesired signals. Many published references describe the performance which may be obtained in stationary situations . In this paper, we are interested in more realistic conditions.

Among the different adaptive techniques, the one we have chosen because of its simple implementation is based on a constraint gradient method. After a summary of these techniques we give a description of the retained algorithm and present some results obtained by simulation.

The following part of this study deals with the dynamic behaviour of a system using such an algorithm ; it is shown how, for stationary signals , a time-decreasing loop gain gives both reduced convergence time and high interference rejection . Then , simulations presented for sources moving with a given angular speed point out the different array behaviours depending on the loop gain.

Finally , a set of simulations illustrates the effects induced by scintillation phenomena that may be provoked by propagation through a turbulent medium.

2. RECALL OF THE ADAPTIVE TECHNIQUES

Progress in computers and new algorithmic methods make digital processing of real-time data for antenna arrays feasible. For instance high resolution methods give excellent results for discrimination of close sources in steady-state cases [1].

Adaptive techniques are also a subject of important interest and investigation because of their ability to respond to an unknown environment , in real-time , by steering nulls in the directions of the interference sources . This is achieved by adjusting weights so as to optimize a cost function under assumptions about desired and interference sources with sometimes a constraint on the weights . Practically , this optimization can be performed by two ways :

- direct methods which compute an estimate of the optimal weights from the covariance matrix ;
- iterative methods which use a gradient technique applied to the weights.

The well-known MSN (maximum signal-to-noise) algorithm of Applebaum [2] and LMS (least-mean-square) algorithm of Widrow [3] have both been developped with the second one . They have obtained very good results in steady-state situations . But there are applications such as moving sources or scanning radars, for which a rapid convergence is an essential requirement. So several works have studied the convergence properties of these methods and two limitations have been shown :

- their transient behaviour depends highly on the eigenvalues of signals covariance matrix which are related to the power of the sources [4] ;
- the convergence rate and so the loop gain are limited by the expected performance [3].

Some methods have then been proposed to improve these systems :

- the SMI (sample matrix inversion) algorithm [5] which gives directly an estimate of the optimal weights of the MSN algorithm, from an estimate of the covariance matrix using few samples . Its rate, independent of the noise environment, has been shown to be often faster than LMS and MSN one in terms of samples. But it is related to the number of array elements, and the covariance matrix estimate and inversion require an important number of complex multiplications;
- the SMG (sample matrix gradient) algorithm [6] which -based on a gradient technique applied to the LMS algorithm - uses a sample covariance matrix estimate as an input data. It provides faster and more stable convergence properties than the iterative techniques but involves a perceptible increase of the number of multiplications.

Other techniques, as reviewed in [7], have also been studied which propose a recursive estimate of the inverse of the covariance matrix , or include orthogonalization of the signals before adaptive processing.

In presence of rapidly changing radar applications , all these studies are able to bring significant ameliorations, but a compromise must still be adopted between a low computational load and improved convergence properties, depending on the dynamic range of the sources and the size of the array.

3. POWER MINIMIZATION ALGORITHM

The adaptive algorithm we have chosen is due to R.T. Compton [8]. Under the only hypothesis that the desired source is weaker than the interference sources, it minimizes the mean array output power :

* Presently at CEA - 91190 Gif/Yvette - France

$$\langle S.S^* \rangle = W^t \langle X.X^{t*} \rangle W^*$$

X : input signal vector, W : weights vector
(^{t} : transpose , * : complex conjugate)

with a quadratic constraint on the weights : $W^tW^* = 1$ (the output thermal noise is held constant). This method is appreciable in many applications because it does not require information about desired signal direction or waveform. It uses a gradient technique and the corresponding iterative equation is :

$$W(j+1) = W(j) - k.S(j).(X^*(j) - W(j).S^*(j))$$

with k : loop gain , j : sample index

We have studied this algorithm for large arrays ; as it needs only two multiplications per sample for each element , it is rather simple to implement.

Though the number of possible nulls in such arrays is most of the time bigger than the number of sources, the hypothesis on the sources power and the use of a gradient technique prevent the desired source from being rejected. So an adapted array pattern will just differ mainly from the initial one in proximity of the interference directions, the main lobe will remain unchanged even pointed towards a week source. Figs. 1, 2 illustrate this behaviour for two narrowband and uncorrelated signals which are incident on a circular array , with a signal-to-interference ratio SIR of 20 db and an output signal-to-noise ratio SNR of 30 db in absence of interference . The same power of uncorrelated gaussian noise is present on each element.

4. CONVERGENCE PROPERTIES IN STATIONARY ENVIRONMENT

The stability of this algorithm has been previously discussed by R.T. Compton [9]. Practically , as it has been already evoked, the choice of the appropriate loop gain is not easy :

- a low loop gain will give a satisfying final signal-to-noise-and-interference ratio SINR after a long transient period, whereas a high loop gain will converge rapidly providing a loss of performance ;
- the convergence rate is highly exemple-dependent .

So we have developped a time decreasing loop gain:

$$k(j) = C / (S.S^*)_0 . ((R-1) . \langle S.S^* \rangle_j / (S.S^*)_0 + 1)$$

with $(S.S^*)_0$: initial output power ; $\langle S.S^* \rangle_j$: mean output power estimated with the first j samples .

Initially $\langle S.S^* \rangle_j = (S.S^*)_0 \Rightarrow k(0) = C.R / (S.S^*)_0$.
After convergence $\langle S.S^* \rangle_j \ll (S.S^*)_0 \Rightarrow k(\infty) = C / (S.S^*)_0$.
So C gives the final loop gain and R is the initial-to-final loop gain ratio.

This loop gain was deduced from the following remarks :

-as $\Delta W = k.S.(X^* - W.S^*)$ is fairly proportional to the total received power , to introduce $1/(S.S^*)_0$ makes ΔW almost independent of the interference power. So a unique loop gain is available for a wide range of signal power ;

-the algorithm minimizes the output power. So the second factor of k(j) diminishes from R to 1. As shown in Fig. 3, k(j) gives, with appropriate values of C and R, the final SINR obtained with the low loop gain, for a convergence rate similar to the one of the high loop gain. The example conditions are identical to those of the first one and the SINR converges to the SNR of 30db.

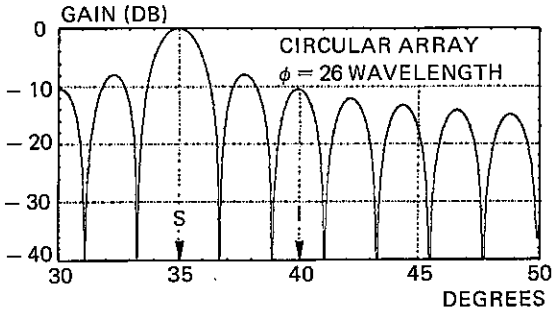


Fig. 1: Initial array pattern

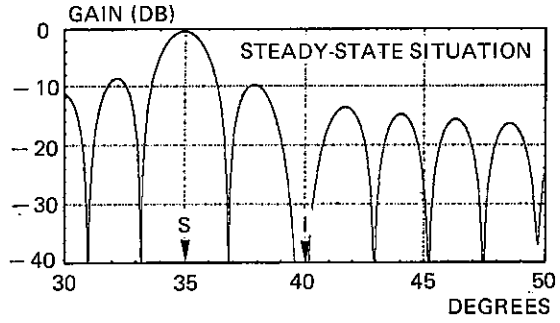


Fig.2 : Adapted array pattern

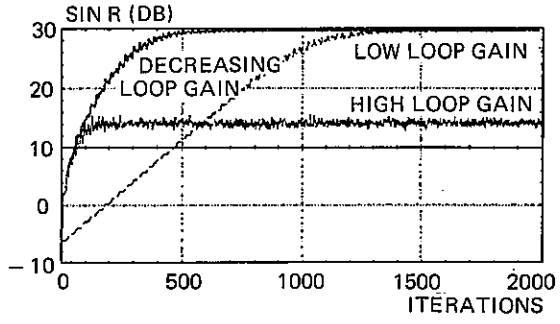


Fig.3 : Output SINR transient

5. NON STATIONARY APPLICATIONS

Since this adaptive system is not a linear one, it can not be entirely defined by its response to an an angular step, and characteristic situations will be described :

-interference sources with an apparent angular movement induced by a scanning antenna or the displacements of interference sources ;

-fluctuating propagation conditions .

5.1 Moving interference sources

In presence of moving interference sources, the directions the array must cancel evolve and the adaptive array processor must perform an "interference tracking" of which efficiency depends a priori on two parameters : interference source angular speed, algorithm loop gain.

To study the influence of the interference movement on the algorithm behaviour, for large arrays, simulations have been carried out, from which a description of the SINR evolution and of the recursive array pattern modifications has been deduced. Two basic angular movements have been analysed :

-an angular speed step ,

-a sinusoidal angular speed (such as obtained with an interference source describing a circle in the array far-field) .

We assume that, initially, the interference has a null angular speed and the array is optimized. The example conditions are identical to those of the preceeding one except for the interference speed.

5.1.1 Constant angular speed

As shown in Fig.4, the interference angular speed step is "followed" by the adaptive processor : the null created in the array pattern has moved from the initial (Fig. 2) to the present interference direction , whereas the main lobe is still pointed towards the desired signal, as result of the initial weighting .

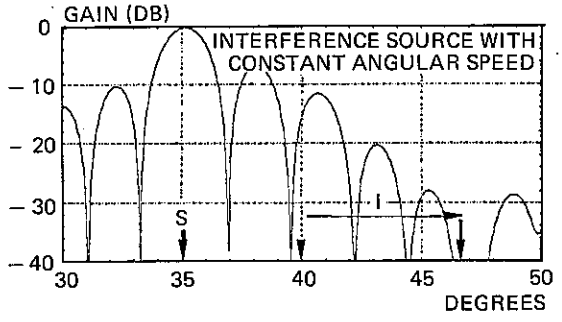


Fig. 4 : Adapted array pattern

But these simulations confirm the dependence of the tracking efficiency on the loop gain and the interference speed : they point out a transient period of the SINR (Fig. 5) , the shape of which can be related to the initial adapted array pattern swept by the interference source (Fig. 6) . However, after convergence the SINR curve reaches its initial value obtained with the same loop gain for the fixed interference source.

This transient period is characterized by its length and its peak value which increase with the interference speed or the inverse of the loop gain , as it can be seen on Fig.7

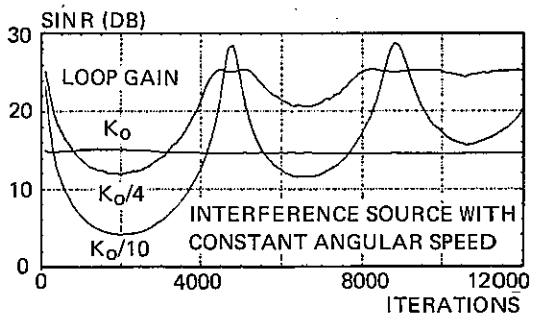


Fig. 5 : Output SINR transient

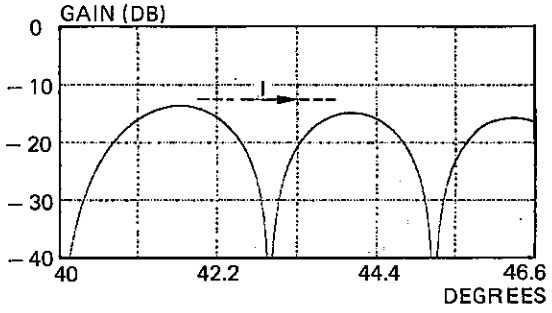


Fig. 6 : Initial array pattern swept by the interference source

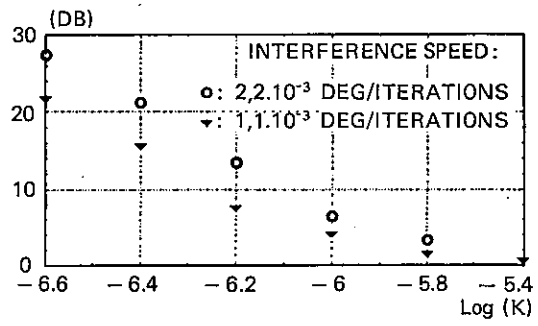


Fig. 7 : Maximum magnitude of the transient oscillations

5.1.2. Sinusoidal angular speed

These simulations give a description of the algorithm behaviour facing the repeated motions of an interference source in a given area , for large arrays. The interference angular speed is defined by its period and its peak value . As shown in Fig. 8 , we notice again the convergence of the algorithm illustrated by a transient period of the SINR which tends to the SNR value (the desired signal gain is maintained). But here, the SINR evolution is related to both the initial gain of the adapted array in the interference direction and the instant interference angular speed.

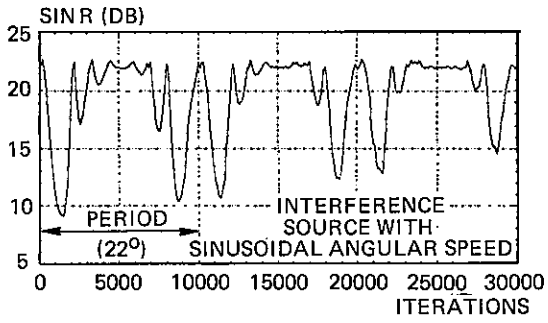


Fig.8 : Output SINR transient

5 . 2 . Turbulent propagation medium

In order to generate the required signals for subsequent Monte-Carlo simulations , angular fluctuations have been defined from a gaussian generator by their rms value and frequency spectrum . According with transionospheric phase-screen model [10] , the retained spectrum of the fluctuations is :

$$\Omega(f) = \Omega_0 \, (f^2 + f_0^2)^{-1.5} \, . \, \sin^2 (\pi \, f / f_1)$$

The numerical application led to $f_1 = 300 \, f_0$ and the sampling frequency for the algorithm was $f_s = 2 \, f_1$, corresponding to the pulse repetition frequency of a satellite tracking radar . For such values , the angular fluctuations are partially correlated from sample to sample . The assumed model does not introduce significant non linear wavefront distortion , and the amplitude scintillation has not been accounted for , in order to distinguish between the different effects .

The SINR evolutions presented on Figs. 9 and 10 correspond to a low and a medium loop gain , each for strong scintillation (0.15 beamwidth rms angular fluctuation), moderate (0.05 beamwidth rms angular fluctuation) , or without scintillation . For the low gain , a decrease in steady-state SINR with increasing scintillation is observed , whereas the higher gain results in nearly constant performance . This one is close to the level obtained at low gain , moderate scintillation , and that confirms the existence of an optimum loop gain at each scintillation level .

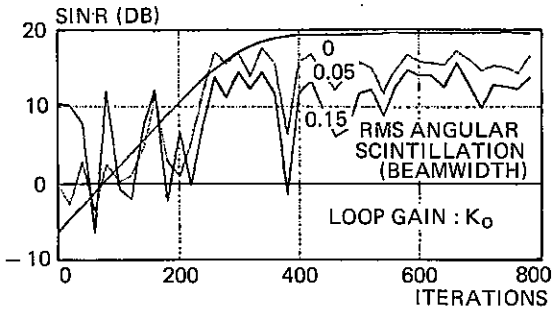


Fig. 9 : Output SINR transient

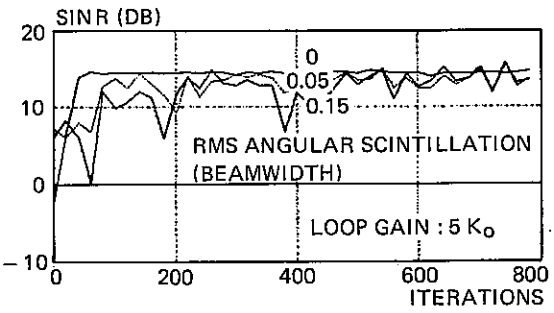


Fig. 10 : Output SINR transient

6 . CONCLUSION

The different simulations which have been presented , with the aim of approaching realistic non-stationary environments , show the adequateness of a gradient adaptive algorithm . The various trade-offs for defining an appropriate loop gain have been pointed out .

REFERENCES

[1] Bienvenu G., "High resolution properties of the space correlation matrix ", 9th. GRETSI Conf. Proc., Nice , May 1983, pp . 239-245 .

[2] Applebaum S.P., "Adaptive arrays", IEEE Trans. Ant.Prop., vol.24, no.5, p.585, Sept.1976 .

[3] Widrow B. , Mantey P. E. , Griffiths L. J. , Goode B. B. "Adaptive antenna systems", in Proc. IEEE vol. 55, no.12, pp. 2143-2159, Dec.1967.

[4] Compton R.T.,"Improved feedback loop for adaptive arrays", IEEE Trans. Aero.Elect.Syst., vol.16, no.2, Mar.1980 .

[5] Reed I.S., Brennan L.E., Mallett J.D.,"Rapid convergence in adaptive arrays", IEEE Trans. Aero. Elect. Syst., vol.10, no. 6 , pp. 853-863 , Nov1974 .

[6] Worms J., Krucker K., "On an improved gradient technique for adaptive array processing", IEEE int. radar. conf. , May 1985 Arlington (Virginia), pp. 39-44 .

[7] Gabriel W. J. , "Special issue on adaptive antennas" , IEEE Trans. Ant. Prop., vol. 24 no. 5 , pp 573 - 574 , Sept. 1976 .

[8] Compton R.T.,"Power optimization in adaptive arrays : a technique for interference protection", IEEE Trans. Ant.Prop., vol.28 , pp.79-85 , Jan.1980 .

[9] Compton R.T.,"A gain optimizing algorithm for adaptive arrays", IEEE Trans. Ant. Prop., vol.26, pp. 228-235 , Mar. 1978 .

[10] Rino C. L. , "A power law phase screen model for ionospheric scintillation " , Radio Science , Vol.14 , no. 6 , pp.1135 - 1155 , 1979 .

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SENSOR FAULT DETECTION BY MEANS OF JOINT LADDER ESTIMATION

U.Appel, W.Ptacek

Institut für Mathematik und Datenverarbeitung
Fakultät Elektrotechnik
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39
D-8014 Neubiberg, FRG

In a stochastic dynamical system monitored by multiple (dissimilar) sensors, changes of the output signal statistics of the sensors often indicate a sensor fault. For the problem of detection and identification of such faults, a numerically effective, approximative algorithm is presented. The algorithm calculates the output residual signals of the system making use of the analytical redundancy comprised in multiple sensor signals and decides upon a fault by means of the generalized likelihood ratio (GLR) approach. While the exact solution for estimating the residuals - based on a vector autoregressive model of the system - would make it necessary to estimate multivariate inverse filters, the presented algorithm uses only mutual pairs of sensor signals, estimating the residual signals by means of a joint process lattice filter. Though not optimal, this algorithm represents a good compromize between computational costs and performance.

1. INTRODUCTION

Physical systems are often subject to unexpected changes, such as component failures and variations in operating conditions, that tend to degrade overall system performance. We will refer to such changes as "failures," although they may not represent the failing of physical components. The problem of signal processing in this case is to detect such failures reliably and to discriminate between sensor failures and system parameter changes even under nonstationary conditions. For this task the basic approach is to use the redundancy comprised in multisensor signals as well as redundancy within each individual signal statistics.

Over the past decade numerous approaches to the problem of failure detection and identification (FDI) in dynamical systems have been developed; an excellent survey of these techniques was given by Willsky /4/. All of the analytical methods require that a model with a fixed parameter set is given a priori. In our approach, however, the only information available is that the system can be represented by a redundant model (Fig. 1.1), but no a priori knowledge about the process parameters is required as they are estimated on-line. In this model, a scalar statistic process is monitored via several sensors having different (and unknown) transfer functions. Using the analytical redundancy comprised in the linearly coupled sensor signals, it can be decided what type of system change - if any - has occurred, and which of the q sensors has failed.

The basic approach for solving this problem is to use a parameter estimation algorithm - e.g. based on a vector autoregressive (AR) model - in order to define inverse filters for the sensor signals, and to process the filter output signals (residuals) subsequently. In the absence of measurement noise and failure, these residuals are zero, while in the presence of failure residual power increases remarkably permitting a reliable fault detection.

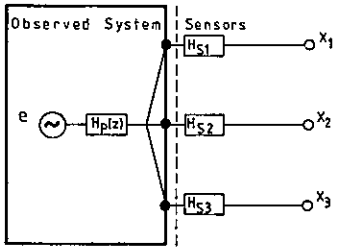


Fig. 1.1. Redundant model of a system under test (three-dimensional).

In general, however, the exact AR model estimation requires a substantial amount of calculations /2/. Also, with this approach the problem is "oversolved" by estimating the exact system parameters instead of only detecting changes in these parameters. Therefore, in this paper a simpler technique is proposed for the detection of parameter jumps using the joint process lattice structure. Though this procedure is optimal only under fairly restrictive conditions as stated in Section 2, it is advantagous due to its simplicity in more general applications too, where it leads to a slightly suboptimal solution only.

2. JOINT ESTIMATION

As stated before, the exact multivariate parameter estimation problem /2/ requires a high amount of numerical operations, making real-time processing with a single digital signal processor (DSP) impossible even for moderate sampling rates. For this reason, the complex task of multichannel filtering must be distributed to several parallel processors. The partitioning of the computational load can be performed with only a slight degradation of performance using the joint process calculation method.

2.1. Lattice for Vector Joint Esitmation

The vector joint process lattice is formed by a regular adaptive lattice filter for one of the q signals and q-1 weightening networks for the backward prediction error signals coming out of the lattice. The coefficients of the lattice are adjusted contiguously according to the order recursions /3/, which are based on the following generalized residual energies.

$$E_{m,i,j}(t) = \underline{e}_m^T(t-i)\ \underline{e}_m(t-j) \qquad (2.1a)$$

$$R_{m,i,j}(t) = \underline{r}_m^T(t-1-i)\ \underline{r}_m(t-1-j) \qquad (2.1b)$$

$$C_{m,i,j}(t) = \underline{e}_m^T(t-i)\ \underline{r}_m(t-1-j) \qquad (2.1c)$$

$0 \leq m \leq p;\ 0 \leq i,j \leq p-1$

The forward and backward prediction error vectors $\underline{e}_m(t)$ and $\underline{r}_m(t)$, respectively, are defined by

$$\underline{e}_m(t) = [e_m^t(t),\ e_m^{t-1}(t),\ldots,e_m^{t-v+1}(t)]^T \qquad (2.1d)$$ *)

and

$$\underline{r}_m(t) = [r_m^t(t),\ r_m^{t-1}(t),\ldots,r_m^{t-v+1}(t)]^T \qquad (2.1e)$$

where v is the period of observation (data window). The particular properties of the generalized residual energies lead to a simple algorithm /3/ which is roughly outlined.

At the start of the solution procedure the following equations give the initial conditions from the covariance matirx $\underline{\underline{\Theta}}(t)$ for m=0,

$$E_{0,i,j}(t) = \underline{x}^T(t-i)\underline{x}(t-j) = \Theta_{i,j}(t) \qquad (2.2a)$$

$$R_{0,i,j}(t) = \underline{x}^T(t-1-i)\underline{x}(t-1-j) = \Theta_{i+1,j+1}(t) \qquad (2.2b)$$

$$C_{0,i,j}(t) = \underline{x}^T(t-i)\underline{x}(t-1-j) = \Theta_{i,j+1}(t) \qquad (2.2c)$$

with the input timeseries $\{x(k) \in R,\ t \leq k \leq t-v+1\}$ which is represented by the vector $\underline{x}(t)$.

$$\underline{x}(t) = [x(t),\ x(t-1),\ \ldots,\ x(t-v+1)]^T \qquad (2.2d)$$

Thus the reflection coefficients for m=1 can be evaluated by (2.3a to 2.3b). Using these results the generalized residual energies with m=1 are given by (2.4a to 2.4d) which completes the initialization procedure.

Using the residual energies for the step m-1 the reflection coefficients for the step m are given by

$$K_m^f(t) = -\,C_{m-1,0,0}(t)/R_{m-1,0,0}(t) \qquad (2.3a)$$

$$K_m^b(t) = -\,C_{m-1,0,0}(t)/E_{m-1,0,0}(t) \qquad (2.3b)$$

with $1 \leq m \leq p$

which permit the updating of the residual energies by means of the following equations.

$$E_{m,0,j}(t) = E_{m-1,0,j}(t) + \qquad (2.4a)$$
$$+ K_m^f(t)C_{m-1,j,0}(t) + C_{m-1,0,j}(t)K_m^f(t-j) +$$
$$+ K_m^f(t)R_{m-1,0,j}(t)K_m^f(t-j)$$

$$R_{m,0,j}(t) = R_{m-1,1,j+1}(t) + \qquad (2.4b)$$
$$+ K_m^b(t-1)C_{m-1,1,j+1}(t) + C_{m-1,j+1,1}(t)K_m^b(t-1-j) +$$
$$+ K_m^b(t-1)E_{m-1,1,j+1}(t)K_m^b(t-1-j)$$

$$C_{m,0,j}(t) = C_{m-1,0,j+1}(t) + \qquad (2.4c)$$
$$+ K_m^f(t)R_{m-1,0,j+1}(t) + E_{m-1,0,j+1}(t)K_m^b(t-1-j) +$$
$$+ K_m^f(t)C_{m-1,j+1,0}(t)K_m^b(t-1-j)$$

$$C_{m,j,0}(t) = C_{m-1,j,1}(t) + \qquad (2.4d)$$
$$+ K_m^f(t-j)R_{m-1,1,j}(t) + E_{m-1,1,j}(t)K_m^b(t-1) +$$
$$+ K_m^f(t-j)C_{m-1,1,j}(t)K_m^b(t-1)$$

Once the reflection coefficients have been computed, the AR model can be realized by a so called canonical ladder form, shown in Fig. 2.1 and mathematically described by the following equations.

$$\underline{e}_m(t) = \underline{e}_{m-1}(t) + \underline{r}_{m-1}(t-1)\ K_m^f(t) \qquad (2.5a)$$

$$\underline{r}_m(t) = \underline{r}_{m-1}(t-1) + \underline{e}_{m-1}(t)\ K_m^b(t) \qquad (2.5b)$$

For the prediction of one process $\underline{y}_1(t)$ from measurements of a related process, e.g.,

$$\underline{y}_1(t) = \sum_{i=1}^{p} d_i \underline{x}(t-i) \qquad (2.6a)$$

we refer to /3/.

Using a similar approach, this algorithm was reformulated for q-1 joint channels

$$\underline{y}(t) = [\underline{y}_1(t),\ \underline{y}_2(t),\ \ldots,\ \underline{y}_{q-1}(t)] \qquad (2.6b)$$

$$\underline{y}_i(t) = [\,y_i(t),\ y_i(t-1),\ \ldots,\ y_i(t-v+1)]^T \qquad (2.6c)$$

$1 \leq i \leq q-1$

with additional calculation of the residual energies in each channel.

*) "T" denotes transpose

Therfore, the computation of the least squares prediction error for the joint processes can be performed by the lattice form (Fig. 2.1) which is mathematically described by

$$\underline{g}_m(t) = \underline{g}_{m-1}(t) + \underline{r}_{m-1}(t-1)\underline{H}_m^T(t). \qquad (2.7a)$$

The prediction error matrix $\underline{g}_m(t)$ and the adaptive weight vectors $\underline{H}_m(t)$, respectively, are defined by

$$\underline{g}_m(t)=[\underline{g}_{1,m}(t),\underline{g}_{2,m}(t),\ldots,\underline{g}_{q-1,m}(t)] \qquad (2.7b)$$

with (2.7c)

$$\underline{g}_{i,m}(t)=[g_{i,m}^{t}(t),g_{i,m}^{t-1}(t),\ldots,g_{i,m}^{t-v+1}(t)]^T$$

where $1\leq i\leq q-1$, $0\leq m\leq p$

and

$$\underline{H}_m(t)=[H_{1,m}(t),H_{2,m}(t),\ldots,H_{q-1,m}(t)]^T \qquad (2.7c)$$

with $1\leq m\leq p$.

Least squares prediction of $\underline{g}_m(t)$ is performed, if the weight vectors are calculated by

$$\underline{H}_m(t) = -\frac{\underline{g}_{m-1}^T(t)\ \underline{r}_{m-1}(t-1)}{\underline{r}_{m-1}^T(t-1)\ \underline{r}_{m-1}(t-1)}\ . \qquad (2.8)$$

To obtain covariance recursions for the inner products of g, r and e, the following auxiliary variables are introduced.

$$\underline{D}_{m,i,j}(t) = \underline{g}_m^T(t-i)\ \underline{r}_m(t-1-j) \qquad (2.9a)$$

$$\underline{A}_{m,i,j}(t) = \underline{g}_m^T(t-i)\ \underline{e}_m(t-j) \qquad (2.9b)$$

$$\underline{G}_{m,0,0}(t) = \underline{g}_m^T(t)\ \underline{g}_m(t) \qquad (2.9c)$$

Thus we obtain the order recursive update equations for the inner products in terms of the auxiliary variables.

$$\underline{D}_{m,i,j}(t) = \underline{D}_{m-1,i,j+1}(t) + \qquad (2.10a)$$
$$+K_m^b(t-1-j)\underline{A}_{m-1,i,j+1}(t) + \underline{H}_m(t-i)R_{m-1,i,j+1}(t)+$$
$$+ \underline{H}_m(t-i)C_{m-1,j+1,i}(t)K_m^b(t-1-j)$$

$$\underline{A}_{m,i,j}(t) = \underline{A}_{m-1,i,j}(t) + \qquad (2.10b)$$
$$+ K_m^f(t-j)\underline{D}_{m-1,i,j}(t) + \underline{H}_m(t-i)C_{m-1,j,i}(t) +$$
$$+ \underline{H}_m(t-i)R_{m-1,i,j}(t)K_m^f(t-j)$$

$$\underline{G}_{m,i,j}(t) = \underline{G}_{m-1,i,j}(t) + \qquad (2.10c)$$
$$+ \underline{H}_m(t-i)\underline{D}_{m-1,j,i}^T(t) + \underline{D}_{m-1,i,j}(t)\underline{H}_m^T(t-j) +$$
$$+ \underline{H}_m(t-i)R_{m-1,i,j}(t)\underline{H}_m^T(t-j)$$

For the determination of the weight vectors, it is easy to check that the calculations (2.10a to 2.10b) must be performed only for i=0 and $0\leq j\leq p-1$. Further, for the calcualtion of the joint process prediction error energy it is sufficient to compute (2.10c) with i=j=0. Consequently, the final joint recursions are given by

$$\underline{D}_{m,0,j}(t) = \underline{D}_{m-1,0,j+1}(t) + \qquad (2.11a)$$
$$+K_m^b(t-1-j)\underline{A}_{m-1,0,j+1}(t) + \underline{H}_m(t)R_{m-1,0,j+1}(t)+$$
$$+ \underline{H}_m(t)C_{m-1,j+1,0}(t)K_m^b(t-1-j)$$

$$\underline{A}_{m,0,j}(t) = \underline{A}_{m-1,0,j}(t) + \qquad (2.11b)$$
$$+ K_m^f(t-j)\underline{D}_{m-1,0,j}(t) + \underline{H}_m(t)C_{m-1,j,0}(t) +$$
$$+ \underline{H}_m(t)R_{m-1,0,j}(t)K_m^f(t-j)$$

$$\underline{G}_{m,0,0}(t) = \underline{G}_{m-1,0,0}(t) \qquad (2.11c)$$
$$+ \underline{H}_m(t)\underline{D}_{m-1,0,0}^T(t) + \underline{D}_{m-1,0,0}(t)\underline{H}_m^T(t) +$$
$$+ \underline{H}_m(t)R_{m-1,0,0}(t)\underline{H}_m^T(t)$$

and

$$\underline{H}_m(t) = -\underline{D}_{m-1,0,0}(t)/R_{m-1,0,0}(t) \qquad (2.11d)$$

The solution procedure of these recursions using the following initial conditions is equal to the lattice algorithm.

$$\underline{D}_{0,i,j}(t) = \underline{y}^T(t-i)\ \underline{x}(t-1-j) \qquad (2.12a)$$

$$\underline{A}_{0,i,j}(t) = \underline{y}^T(t-i)\ \underline{x}(t-j) \qquad (2.12b)$$

$$\underline{G}_{0,0,0}(t) = \underline{y}^T(t)\ \underline{y}(t) \qquad (2.12c)$$

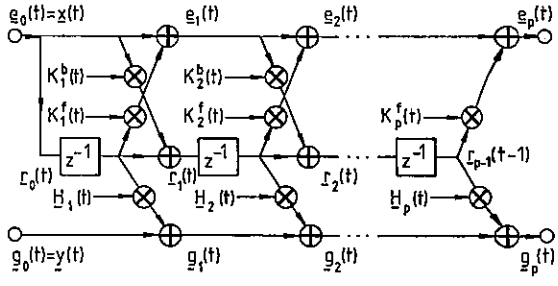


Fig. 2.1. Lattice for vector joint process prediction.

By calculation the equation (2.7a) of the prediction signals $\hat{y}(t)$ using the adaptive weight vector in (2.11d), the residual signals $\underline{g}_{m,i}(t)$ are calculated orderrecursively by subtracting from y(t) those portions which are correlated with the mutually orthogonal backward prediction error signals $\underline{r}_m(t-1)$. From these residual signals then the residual energies $\underline{g}_m(t)$ are computed, which are necessary for the calculation of failure decision statistics (generalized likelihood ratio - GLR).

2.2. Joint Process Calculation Method

In this approach, the redundant model of dimension q=m is substituted by an array of n basic

units of dimension q=2, each covering two sensors. (The basic unit is the discussed model of Fig. 2.1 but now reduced to dimension q=2). For an equal treatment of all sensors, each sensor signal has to be utilized once as input of a joint channel and once as main channel, i.e. m=n (see Fig. 2.2).

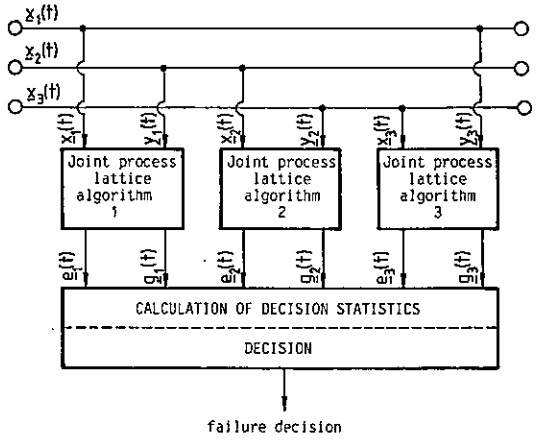


Fig. 2.2. Joint process calculation method (q=3).

A parameter jump due to a sensor fault leads to an increase in residual signal power only in the joint channel afflicted with the failure and the corresponding main channel, while abrupt variations in the observed process change the residual signal $\underline{e}_p(t)$ in all main pathes.

Therfore, a clear discrimination between variations in the basic process and changes of the sensor signals due to failure can be performed. For calculation of decision statistics, we refer to the GLR approach.

This procedure, which is based on a proposal of Willsky /5/ for use in failure detection applications, has proven to be an efficient and easy to use algorithm for the detection of parameter jumps in an autoregressive signal /1/. It is based on the comparison of signal segments defined by a growing reference window and a sliding test window. Within these two intervals as well as within a joint interval formed by concatenation of both, signal parameters are calculated, and by applying a suitable distance measure to the respective residual powers a decision can be made on whether the sigal parameters have been changed or not. Denoting these powers by $E^T_{p,0,0}(t)$ for the test window, $E^R_{p,0,0}(t)$ for the reference window and $E^P_{p,0,0}(t)$ for a "pooled" window formed by concatenation of both, a likelihood test can be performed as shown in detail in /1/ using these powers as well as the window length N_T, N_R and N_P, respectively, leading to a maximum likelihood ratio (2.13). This ratio λ compares the likelihood l_0 for the system of having the same parameters within the time boundaries defined by the two windows with the likelihood l_1 of having different parameters.

$$\lambda = l_0/l_1 = \frac{(E^R_{p,0,0})^{N_R}(E^T_{p,0,0})^{N_T}}{(E^P_{p,0,0})^{N_P}} \qquad (2.13)$$

In order to transform this ratio to a distance measure d (with d=0 for identical parameters), it is common practice to use the logarithm instead of λ. By comparing d with a reasonably choosen threshold, then, a decision can be made on a possible parameter jump of the system under test.

3. CONCLUSION

This paper has presented a joint process approach for the development of a FDI system. The only information used in this method was that the system is representable by means of a redundant model parameters of which may be unknown. The general AR approach should be used for such applications which can take advantage of the intercorrelations between several channels thus justifying the high computational costs. For time critical applications where employment of DSP units is indicated the joint process calculation method is the method by choice. Both approaches have been tested intensively in various simulations proving the expected properties of the proposed methods.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the support of the Deutsche Forschungsgemeinschaft, which made this work possible.

REFERENCES

/1/ Appel, U., A. v.Brandt (1983). Adaptive sequential segmentation of piecewise stationary time series. Information Sciences 29, pp. 27-56.

/2/ Ptacek, W., U. Appel (1986). Detection of sensor faults by means of multivariate calculation methods. Submitted to Proc. 2nd IFAC Workshop on Adaptive Systems in Control and Signal Processing, Lund, Sweden

/3/ Strobach, P. (1985). Schnelle adaptive Algorithmen zur ordnungsrekursiven Kleinste-Quadrate-Schätzung autoregressiver Parameter. Thesis, Univ. d. Bundeswehr, Neubiberg, FRG.

/4/ Willsky, A.S. (1976). A survey of design methods for failure detection in dynamic systems. Automatica, vol. 12, pp. 601-611.

/5/ Willsky, A.S., and H.L. Jones (1976). A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. IEEE Trans. Automat. Contr., vol. AC-21, pp. 108-112.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

PASSIVE ARRAY TREATMENT : DETECTION OF SIGNALS AND ESTIMATION OF THE SPECTRAL MATRIX OF THE NOISE

I. TAS AND C. LATOMBE.

Centre d'Etude des Phénomènes Aléatoires et Géophysiques ; UA 346
BP 46, 38402 Saint Martin d'Hères, FRANCE.

Four criteria used to detect how many uncorrelated excitations are reaching a passive array are recalled. They are compared on simulated signals and their results on real underwater acoustic signals are displayed. These criteria assume that records on each sensor of the passive array are corrupted by additive noises that are supposed to be uncorrelated and of equal variances. When noises are uncorrelated but do not have equal variances, the diagonal-type spectral matrix of the noises is estimated using principal component analysis. The straightforward amelioration brought in the localisation of the sources is proved.

1.ESTIMATION OF THE NUMBER OF IMPINGING SOURCES

In passive array treatment (radar, sonar, sismology), an unknown number P of point-like sources radiate P signals that propagate through a transmission medium and reach an array of N sensors. The recorded signals are built by the mixed impinging signals corrupted by additive noises. Although the a priori knowledge is very thin, the usual hypotheses on the mediun and on the source signals are :
i) The medium is linear, homogeneous and isotropic, meaning that each wave-front is spatially coherent.
ii) The P radiated signals and the noises are zero-mean stationary stochastic signals. Signals are uncorrelated and uncorrelated with the noises.

The N dimensional vector $\underline{R}(f)$, whose components are the Fourier transforms of the recorded signals is complex and supposed to be approximatly gaussian :

$$\underline{R}(f) = \sum_{i=1}^{P} A^i(f)\,\underline{s}^i(f) + \underline{B}(f) \qquad (1)$$

where $A^i(f)$ is the Fourier transform of the amplitude of the i^{th} source and $\underline{s}^i(f)$ its directional vector (cf. [1], [2]).

The NxN spectral matrix of the recorded signals is then :

$$\gamma(f) = E\{\,\underline{R}(f)\,\underline{R}^{+}(f)\,\}$$
$$= \sum_{i=1}^{P} E\{\,|A^i(f)|^2\,\}\,.\,\underline{s}^i(f)\,.\,\underline{s}^i(f)^{+} + \gamma_b(f)$$

where $^+$ denotes the conjugate transpose of a vector, and $\gamma_b(f)$ is the spectral matrix of the noises .
The set of the linearly independant vectors $\{\underline{s}^1(f),\ldots,\underline{s}^P(f)\}$ spans the P dimensional source subspace.

Depending on the spectral estimation procedure, $\gamma(f)$ can be estimated by :

$$\hat{\gamma}(f) = \frac{1}{K} \sum_{1}^{K} \underline{R}_k(f)\,\underline{R}_k(f)^{+}$$

where $\underline{R}_k(f)$ are independant realisations of the observation vector and K represents the number of periodograms in the periodogram method or the product "bandwidth x time of integration" in the smoothed periodogram method ([3]).

The log-likelihood of R(f) is :
$\log V(R) = -KN \log \pi - K \log |\gamma(f)| - K\,tr[\,\hat{\gamma}(f)\,\gamma^{-1}(f)\,]$
where $|\,.\,|$ and $tr[\,.\,]$ denote the determinant and the trace of a matrix.

Assuming that the noises are uncorrelated and of equal variances, the spectral matrix of the noises is scalar and depends on a single parameter : $\gamma_b(f) = \beta(f)\,I$. If there are P sources, $\underline{R}(f)$ depends on a set of parameters ε_p (characterizing the P sources and the noise variance).
If P is given, it is known ([5], [6]) that the maximum likelihood estimate of ε_p provides the M.L.E. of the source subspace and of the noise variance :

$$\hat{\beta}_p(f) = \frac{1}{N-P} \sum_{i=P+1}^{N} \hat{\lambda}_i(f) \qquad (2)$$

where $\{\,\hat{\lambda}_i(f)\,\}$ are the eigenvalues (in decreasing order) of the computed spectral matrix $\hat{\gamma}(f)$.

The maximum of the log-likelihood becomes ([5], [6]) :

$$\max_{\varepsilon_p} \log [V(R)\,|P] = -K \log [\,\pi^N . \prod_{i=1}^{P} \hat{\lambda}_i(f) \,.\, (\hat{\beta}_p(f))^{N-P}\,]$$
$$= -K \log \mu_p(f) + C_1(f) \qquad (3)$$

$$\text{where } \mu_p(f) = \hat{\beta}_p(f)^{N-P} / \prod_{j=P+1}^{N} \hat{\lambda}_i(f) \qquad (4)$$

and $C_1(f)$ does not depend on P.

1.1. Tests of detection

We recall briefly these tests that are more detailed in the referenced paper [7], [8], [4].
The Akaike Information Criterion (AIC) and the Minimum Description Length Criterion (MDL) select the number $\hat{P}$ that minimize respectively the expression :

$$AIC(P) = -\max_{\varepsilon_p} [\,\log V(R)\,|P] + NPL$$

$$MDL(P) = -\max_{\varepsilon_p} [\,\log V(R)\,|P] + (NPL\,.\,\log K)\,/\,2$$

where NPL is the number of free parameters of the model (NPL = P (2N - P) + 1) and the maximum of the log-likelihood is computed using (3).

The third criterion is issued from the generalized likelihood ratio test defined (cf. [4], [8]) as :

$$A(P, f) = \frac{\max_{H_p} [V(R, f)|\varepsilon_p]}{\max_{H_q} [V(R, f)|\varepsilon_q]}$$

When hypotheses H_p and H_q are :

H_p : there exists at most P sources

H_q : there exists more than P sources,

the generalized likelihood ratio test takes the value :

$A(P, f) = [\mu_p(f)]^{-k}$

where $\mu_p(f)$ is known by (4).

Asymptotically (K large) the statistic - 2 log A(P,f) has a χ^2 distribution with $(N-P)^2$ -1 degrees of freedom. In practice K is small and a better statistic is X(P, f) = C(P).log A(P,f) where $C(P) = 1 - P/K - [2(N-P)^2 + 1] / 6K(N-1)$. X(P,f) has approximatly the same distribution (cf. [9]).

The detection criterion, denoted LRT, is then :

$$X(P, f) \underset{H_p}{\overset{H_q}{\lessgtr}} \chi_p^2$$

where χ_p^2 is a fixed level determined when the probability η_p is chosen :

$\text{Prob}\{ X(P, f) \geq \chi_P^2 \mid H_P \} = \eta_P$

$P_d = 1 - \eta_P$ is called the probability of detection.

A heuristic criterion that estimate $\hat{P}$ as the number of eigenvalues of $\hat{\gamma}(f)$ larger than a fixed level S can be used (cf. [4]). This level is chosen as :

$$S = \frac{1}{N} \max_{f_j} \left[\frac{1}{N} \sum_{i=1}^{N} \hat{\lambda}_i (f_j)\right]$$

and represents a mean energy level on the sensors. The corresponding simplified test is denoted C.S.S.

1.2. Results of the tests on simulations

The simulated array consists of 5 non equi-spaced sensors, the incoming wave field consists of three plane-waves at incidences θ_1, θ_2 and $\theta_3 = 0$ (cf. figure 1).

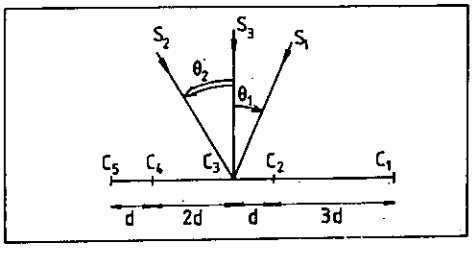


Figure 1

S_1 transmits a narrow-band signal centered around $f_1 = 32$ Hz, S_2 transmits 2 narrow-band signals centered around f_1 and $f_2 = 64$ Hz, and S_3 transmits 3 similar signals centered around f_1, f_2 and $f_3 = 96$ Hz. Hence the correct desision should be : $\hat{P} = 3$ around f_1, $\hat{P} = 2$ around f_2 and $\hat{P} = 1$ around f_3.

Uncorrelated noises of equal variances were simulated leading to SNR varying between 5 db and -1 db depending on the sensors. With K = 12.5, criterions AIC and MDL gave parasitic sources whence CSS and LRT ones gave the desired results.

To study the robustness of these criteria viz uncorrelated noises of unequal variances, simulations were conducted with a diagonal noise spectral matrix : its elements were unequal and their maximal ratio was greater than 2. Hundred ten independant realisations were generated and in each case the estimated $\hat{P}$ was computed. Figure 2 shows the mean value of the estimated $\hat{P}$.

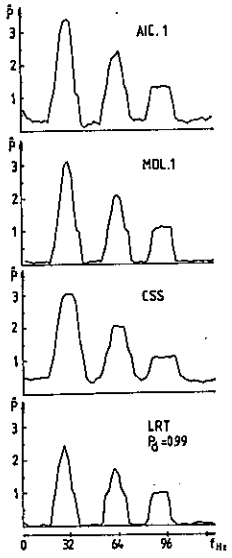


Figure 2

The best criteria are MDL and LRT. MDL tends to overestimate $\hat{P}$ giving parasitic sources whence LRT with $P_d = 0.99$, on the contrary, tends to underestimate $\hat{P}$.

1.3. Results of the tests on real underwater acoustic signals

The studied signals were recorded during a sea-experiment with a calm weather. An immerged source transmits a low-frequency monochromatic signal (f_0 frequency) that is recorded, after propagation through the sea, on a towed linear passive array of 96 sensors (total length : 2 400 m, distance between sensors : 25 m). The source is towed by a ship that keeps a constant direction, the array is towed by a second ship so as the sonar source remains on the normal of the array. The distance transmittor-array varies of 7 kms per hour up to 2000 kms.

Recorded signals are split in "firings" lasting 16 s. and the time-interval between two firings is nearly 4 s. As the signal is monochromatic, a large B_e T cannot be used to estimate the spectral matrix of the records (B_e T = 5 was used). A partial result is presented on figure 3 : 6 sensors are chosen (distance between sensors equal to 375 m). The spectral matrix is obtained by smoothing mean periodograms (average on 5 consecutive firings). In this real situation also, only the LRT criterion gives desired conclusions.

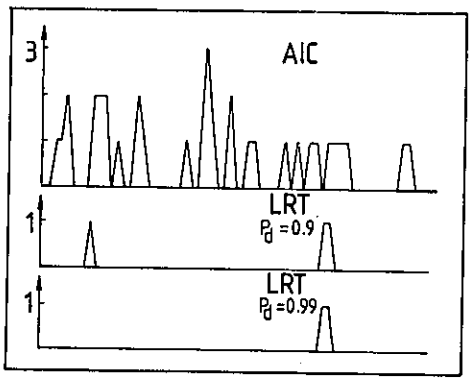


Figure 3

2. ESTIMATION OF THE NOISE SPECTRAL MATRIX

The usual hypothesis of uncorrelated noises of equal variances is usually made because lacking of better a-priori knowledge. Supposing now that noises are uncorrelated but of unequal variances, the diagonal noise spectral matrix can be estimated from the records using technics of principal factor analysis.

Equation (1) can be rewritten in matrix notation :

$$\underline{R}(f) = S(f) \cdot \underline{A}(f) + B(f) \qquad (5)$$

where $\underline{A}(f) = [\, A^1(f), \ldots, A^P(f) \,]^T$ and the N x P matrix S(f) is :

$S(f) = [\underline{s}^1(f), \ldots, \underline{s}^P(f)\,]$

Thus (cf. [10], [11]) it is said that a P factor model holds for $\underline{R}(f)$; components of $\underline{A}(f)$ are the **common factors** and components of $\underline{B}(f)$ the **specific factors.**

The covariance matrix of the (zero-mean) records is the spectral matrix :

$$\gamma(f) = S(f) \cdot \Phi(f)\, S^+(f) + \gamma_B(f) \qquad (6)$$

where $\Phi(f)$ is the covariance matrix of the common factors.

Supposing that a P factor model (5) or equivalently (6) holds for the observation $\underline{R}(f)$, to perform the principal factor analysis of the data means to determine matrices $[\, S(f), \gamma_B(f)\,]$ such that $\tilde{\gamma}(f) = S(f)\, \Phi(f)\, S(f) + \gamma_B(f)$ must be as "near" as possible to the estimate $\hat{\gamma}(f)$ of the spectral matrix of the data.

2.1. Estimation of the noise spectral matrix

As the system (5) is undetermined, the following hypotheses are made :

i) the specific factors are uncorrelated ie :

$\gamma_B(f) = \text{diag}\,[\, \beta_1(f), \ldots, \beta_N(f)\,]$

ii) the common factors are uncorrelated and their variances are equal to 1 ie $\Phi(f) = I$.

iii) the commun factors must verify the constraint that $S^+(f) \cdot \gamma_B(f)\, S(f)$ is diagonal.

These conditions are not restrictive because the common factors are unaffected by re-scaling and rotation ([10]).

The estimation algorithm supposes that P is known ; it can be estimated using the LRT test that is robust viz these modified hypotheses.

The algorithm is not detailed here ; it works by iterations on the estimations $\gamma_B^{(i)}(f)$ and uses explicitly all the eigenvalues and eigenvectors of $\hat{\gamma}(f)$. It stops when $\gamma_B^{(i+1)}$ is sufficiently close to $\gamma_B^{(i)}$.

2.2. Results on simulations

To prove how this algorithm works, a linear array of 10 sensors is simulated ; a one-source and a 2-sources cases are studied, at a fixed frequency. The signal-to-noise ratio for source k of power density α_k, is defined as :

$$\text{SNR}(k) = 10 \log\, [\, \alpha_k / \sum_{i=1}^{N} \beta_i \,]$$

One-source case : the diagonal matrix γ_B is :

[10 ; 20 ; 30 ; 40 ; 50 ; 60 ; 70 ; 80 ; 90 ; 100]

The SNR is equal to - 10.4 db. After 6 iterations, the estimation values of β_i are :

[10.03 ; 20.03 ; 30.03 ; 40.03 ; 50.03 ; 60.03 ; 70.03 ; 80.0 ; 90.03 ; 99.69]

Two-source case : matrix γ_B is :

[50 ; 55 ; 60 ; 65 ; 70 ; 75 ; 80 ; 85 ; 90 ; 95 ; 100]

The two planar-wave impinging sources have incidences of 22° and 30° ; SNR have values of -11.23 db and -11.07 db.

After 4 iterations, the estimated β_i are :

[57.40 ; 60.24 ; 63.00 ; 66.03 ; 69.79 ; 74.64 ; 80.72 ; 87.70 ; 95.04 ; 107.27]

2.3. Improvement of high resolution methods

This improvement is displayed on figure 4 : in the preceeding two-source case, a classical high resolution method ([6]) is used to estimate the directions of the incoming plane-waves. If the noise matrix is, as usually, supposed to be scalar, the method cannot separate the sources whence it does if the diagonal noise matrix is estimated and substracted.

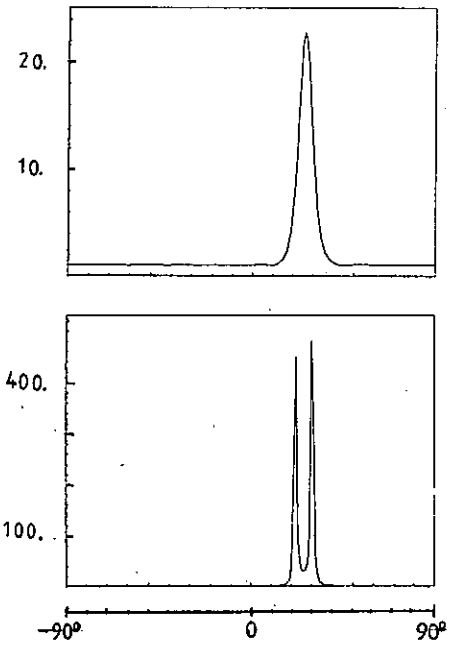


Figure 4

CONCLUSION

Different tests were presented and compared. After simulations, the LRT test appears to be the better one because it does not detect as many parasitic sources as the others ; it works correctly on the treated real data. Principal factor analysis was applied in this passive array treatment to estimate the diagonal noise spectral matrix, providing hence a significant improvement in the localization of the sources.

ACKNOWLEDGEMENTS

We are grateful to GERDSM (Le Brusc) who provided the acoustic underwater signals.

REFERENCIES

[1] LATOMBE C. , Non conventional array treatment using the eigensystem of the spectral matrix, Proc. EUSIPCO 1983, pp. 499-503.

[2] GLANGEAUD F. and LATOMBE C., Identification of electromagnetic sources, Annales Geophysicae, vol. 1, n° 3, May-June 1983, pp. 245-252.

[3] JENKINS, WATTS, Spectral Analysis and its applications, Holden Day, N.Y., 1968.

[4] TAS I., LATOMBE C., Détection multiple par les valeurs propres de la matrice spectrale, Traitement du Signal, in print.

[5] LATOMBE C., Détection et caractérisation des signaux à plusieurs composantes à partir de la matrice interspectrale, Thèse d'Etat, Grenoble, 1982.

[6] BIENVENU G., KOPP L., Optimality of high resolution array processing using the eigensystem approach, IEEE Trans. on ASSP, vol. ASSP-31, n° 5, Oct. 1983, pp. 1235-1247.

[7] WAX M., KAILATH T., Detection of Signal by information theoretic criteria, IEEE Trans. ASSP - 33, Apr. 1985, pp. 387-392.

[8] BIENVENU G., KOPP L., Multiple detection using eigenvalues when the noise spatial coherence is partially unknown, in Adaptive methods in underwater acoustics, ed. HEINZ G. URBAN, NATO, ASI Series, 1984.

[9] JAMES A.T., Tests of equality of the latent roots of the covariance matrix, in Multivariate Analysis, vol. II, ed. KRISHANIAH, Academic Press, N. Y., 1969.

[10] MARDIA K.V., KENT J.T. BIBBY J.M., Multivariate analysis, Academic Press, London, 1979.

[11] LEBART L., MORINEAU A., FENELON J.P., Traitement des données statistiques, Dunod, 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

COMPARISON OF THREE SIMPLE ESTIMATORS FOR THE IDENTIFICATION OF AN UNKNOWN, CONSTANT OR SLOWLY VARYING PARAMETER

P. Gruber, J. Tödtli,

LGZ Landis & Gyr Zug Corp., Central Laboratory, CH-6301 Zug, Switzerland

The problem of estimating an unknown parameter $\bar{a}(t)$ from noisy measurements is considered in this paper. For the unknown, constant or slowly varying parameter $\bar{a}(t)$ the random walk is used as a discrete stochastic model. Three different simple estimators are then compared with respect to the variance of the estimation error in the steady state
1) two optimized moving averagers
   a) one for which a time delay is allowed between the generation of $\bar{a}(kT)$ and the occurrence of its estimate
   b) one for which no time delay is allowed
2) the optimal recursive filter (steady state Kalman filter).

1. INTRODUCTION

If an unknown parameter $\bar{a}(t)$ should be identified from noisy measurements, one simple possibility of doing it is by averaging the noisy measurements. If the parameter $\bar{a}(t)$ is constant then the quality of the estimate increases with increasing averaging length L. If the parameter $\bar{a}(t)$ however is slowly time varying then for a good quality of the estimate L should be not too small in order to average out the noise and not too long in order to follow the time variation of the parameter [1]. To find a suitable length L one needs some knowledge about the time variation of $\bar{a}(t)$. One such possibility is to use a simple stochastic model that allows on the one hand to generate the time dependy of $\bar{a}(t)$ and on the other hand to determine an optimal averaging length. In the special case of the nonstationary random walk model for $\bar{a}(kT)$, $k = 0, 1, \ldots$, the following questions can be asked:

- If a time delay between the generation of $\bar{a}(kT)$ and the occurence of its estimate by a moving averager is allowed, then
  a) which time delay minimizes the variance of the estimation error for a given averaging length L?
  b) which length L minimizes the variance of the estimation error (e.g. the one evaluated under a)) for a given time delay between the generation of $\bar{a}(kT)$ and the occurence of its estimate?
- If no time delay is allowed between the generation of $\bar{a}(kT)$ and the occurence of its estimate then
  a) which averaging length L minimizes the variances of the estimation error?
  b) what improvement can be achieved by the best recursive filter for the random walk model?

The slowly time varying behaviour of parameters is often encountered if solid state devices are used for sensors of physical quantities (e.g. 1/f-noise [2], [3], [4]).

In the following it is now assumed that for the estimation of the unknown, constant or slowly varying parameter $\bar{a}(t)$ the following discrete time random walk model for $\bar{a}_k = \bar{a}(kT)$ may be used

$$a_{k+1} = a_k + v_k \tag{1}$$

$$y_k = a_k + w_k \tag{2}$$

where a_k represents $\bar{a}_k$, w_k the measurement noise, v_k the random walk noise (system noise) and T the sampling period. a_0 is nonrandom but unknown, the sequences of random variables $\{v_k\}$ and $\{w_k\}$ are uncorrelated that means

$$\begin{aligned} &\mathrm{Var}[v_k] = \sigma_v^2 \qquad E[v_k]=0 \\ &\mathrm{Var}[w_k] = \sigma_w^2 \qquad E[w_k]=0 \end{aligned} \quad k=0,1,\ldots$$

$$\begin{aligned} &\mathrm{Cov}[v_k, v_i]=0 \qquad \text{for } i\neq k \\ &\mathrm{Cov}[w_k, w_i]=0 \qquad \text{for } i\neq k \\ &\mathrm{Cov}[v_k, w_i]=0 \qquad \text{for } \forall k,i \end{aligned} \tag{3}$$

The statistics of (1) and (2) is thus given with

$$a_k = a_0 + \sum_{i=0}^{k-1} v_i \tag{4}$$

by the mean values

$$E[a_k] = E[a_0] + E\left[\sum_{i=0}^{k-1} v_i\right] = a_0 \tag{5}$$

$$E[y_k] = a_0 \tag{6}$$

and the variances

$$\mathrm{Var}[a_k] = E[a_k^2] - (E[a_k])^2 = E\left[\left(a_0 + \sum_{i=0}^{k-1} v_i\right)^2\right] - a_0^2 = k\,\sigma_v^2 \tag{7}$$

$$\mathrm{Var}[y_k] = k\,\sigma_v^2 + \sigma_w^2 \qquad (8)$$

Both variances (7) and (8) increase linearly with k. The estimation of a_k will now be done by filtering the signal y_k (see Fig. 1).

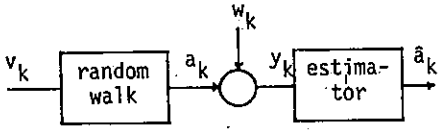


Fig.1: Block diagram of the estimation scheme

2. AVERAGING

The first estimator under consideration is the simple moving averager of length L

$$m_k^L = \frac{1}{L}\sum_{i=0}^{L-1} y_{k-i} \qquad (9)$$

If $\hat{a}_{k-L+n|k}$, L fix, $1 \le n \le L$, denotes the estimate of a_{k-L+n} given measurements up to k, then (9) can be viewed as an estimate $\hat{a}_{k-L+n|k}$ of a_{k-L+n}, $1 \le n \le L$. If $n<L$, a time delay is inherent between the generation of a_{k-L+n} and the occurence of its estimate $\hat{a}_{k-L+n|k}$.

The statistics of the estimation error

$$\tilde{a}_{k-L+n|k} = m_k^L - a_{k-L+n} \qquad (10)$$

are given by the mean

$$E\left[\tilde{a}_{k-L+n|k}\right] = E\left[\frac{1}{L}\sum_{i=0}^{L-1} y_{k-i}\right] - E\left[a_{k-L+n}\right] \qquad (11)$$

and the variance

$$V(n,L)=\mathrm{Var}[\tilde{a}_{k-L+n|k}]=E[(\tilde{a}_{k-L+n|k})^2] - (E[\tilde{a}_{k-L+n|k}])^2 \qquad (12)$$

With (1), (2) and (3) one obtains for (11)

$$E\left[\tilde{a}_{k-L+n|k}\right]=E\left[\frac{1}{L}(La_0+\sum_{j=1}^{L}\sum_{i=0}^{k-j} v_i+\sum_{i=0}^{L-1} w_{k-i})\right]-a_0=0 \qquad (13)$$

and for (12)

$$V(n,L)=E[(\frac{1}{L}(La_0+L\sum_{i=0}^{k-L} v_i+\sum_{i=1}^{L-1}(L-i)v_{k-L+i}+\sum_{i=0}^{L-1} w_{k-i} -a_0-\sum_{i=0}^{k-L} v_i-\sum_{i=1}^{n-1} v_{k-L+i})^2]$$

$$=E[(\sum_{i=1}^{n-1}(\frac{L-i}{L}-1)v_{k-L+i}+\sum_{i=n}^{L-1}(\frac{L-i}{L})v_{k-L+i} +\frac{1}{L}\sum_{i=0}^{L-1} w_{k-i})^2]$$

$$=\frac{1}{L^2}\sigma_v^2\sum_{i=1}^{n-1} i^2 + \frac{1}{L^2}\sigma_v^2\sum_{i=1}^{L-n} i^2 + \frac{1}{L}\sigma_w^2 \qquad (14)$$

The estimator is therefore unbiased and the variance V(n,L) has the interesting properties:

1) V(n,L) is dependent on n and L but not on k, that means V(n,L) is not a function of time.
2) V(n,L) is dependent on the noise ratio
$$R = \sigma_v^2/\sigma_w^2$$
3) The influence of the measurement noise σ_w^2 is decreasing with 1/L.
4) The influence of the "system noise" σ_v^2 is increasing linearly with L
5) For high values of R the minimum of V(n,L) might be at the constraint L = 1.

To obtain the best estimator of the type (9), for which an inherent time delay is allowed, one has to find now the minimum of the function V(n,L). This is done analytically by setting first $\partial V(n,L)/\partial n=0$ (while L is kept constant), resulting in $n_{opt}(L)$ and $V_{nopt}(L)$. Then $V_{nopt}(L)$ is minimized with respect to L by setting $\partial V_{nopt}(L)/\partial L = 0$, yielding L_{opt}.

2.1. Derivation of $n_{opt}(L)$

Using the following summation formula

$$\sum_{i=1}^{N-1} i^2 = \frac{(N-1)N(2N-1)}{6}$$

one obtains for (14)

$$V(n,L)= \frac{1}{L^2}(L\sigma_w^2 +\sigma_v^2(\frac{(n-1)n(2n-1)}{6} + \frac{(L-n)(L-n+1)(2(L-n)+1)}{6})) \qquad (15)$$

Interpreting n as a continuous variable, one can compute $\partial V/\partial n$ and then setting the derivative equal to 0.

$$\frac{\partial V}{\partial n} = 0 \longrightarrow n_{opt}(L) = \frac{L+1}{2} \qquad (16)$$

The minimum value of V is

$$V(n_{opt},L) = \frac{1}{L}(\sigma_v^2 \frac{(L+1)(L-1)}{12} + \sigma_w^2) \qquad (17)$$

The minimal value of the estimate error is therefore obtained for the time instant that lies in the middle of the averaging interval, that means the best estimator is the one that interpretes (9) as the estimate of $a_{k-(L-1)/2}$.

If the minimal and maximal values for n,(n=1 or L) are plugged into (15) one gets for V

$$V(1,L) = V(L,L) = \frac{1}{L}(\sigma_v^2 \frac{(L-1)(2L-1)}{6} + \sigma_w^2) \quad (18)$$

From (18) it follows that the estimator which works with the maximum time delay (n=1) and the one which has no inherent time delay (n=L) show the same performance with respect to V. The ratio μ between (18) and (17) is

$$\mu = 2\frac{(L-1)(2L-1)R + 6}{(L-1)(L+1)R + 12} \qquad R=\sigma_v^2/\sigma_w^2 \qquad (19)$$

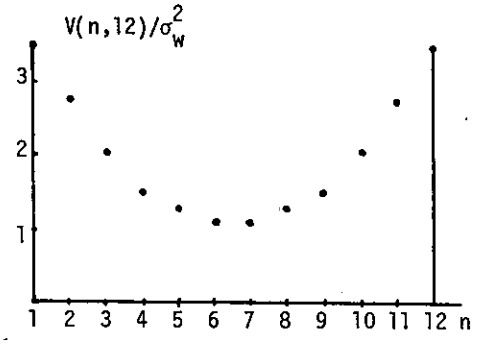


Fig.2: $V(n,12)/\sigma_w^2$ as a function of n (R=1)

μ is always greater than 1 except for the special case R = 0 (equation (1) reduced to $a_{k+1} = a_k$). The maximal value is given with $R \to \infty$ by

$$\mu_{max} = \frac{2(2L-1)}{L+1}$$

$$1 \leq \mu \leq \frac{2(2L-1)}{L+1} \qquad (20)$$

2.2. Derivation of L_{opt}

To obtain the optimum length $L_{opt}(n=n_{opt})$ of the averager with inherent time delay allowed the derivative of (17) with respect to L (again interpreted as a continuous function) is set to zero. This yields

$$L_{opt}(n=n_{opt}) = \sqrt{12/R - 1} \qquad (21)$$

As L_{opt} has to be a positive integer in order to realize the averager one can take

$$L^Q_{opt} = [L_{opt}] \qquad (\text{with } L^Q_{opt} \geq 1)$$

where [x] means the integer part of x.

For the optimum length of the averager for which no inherent time delay is allowed a similar derivation yields

$$L_{opt}(n=L_{opt}) = \sqrt{0.5+3/R} \qquad (22)$$

Both optimal lengths are only a function of the noise ratio R. If R <<1, then

$$L^Q_{opt}(n=n_{opt}) \simeq 2L^Q_{opt}(n=L_{opt})$$

whereas for R>>1

$$L^Q_{opt}(n=n_{opt}) \simeq L^Q_{opt}(n=L_{opt}) = 1$$

3. OPTIMAL RECURSIVE FILTER

In applications where no time delay between the signal generation and the occurence of its estimate is allowed, one can compare the moving averager without time delay (n=L) with the steady state Kalman filter for the model (1), (2). The time varying Kalman filter estimates at every time instant k a_k such that

$$\tilde{p}_k = E[(\hat{a}_k - a_k)^2] = Var[\hat{a}_k - a_k] \qquad (23)$$

is minimal. For the stationary filter one obtains with (1), (2) and (3) the following scalar equation for the variance in the steady state

$$p^*_k = Var[a_k - a^*_k]$$

$$p^*_{k+1} = p^*_k - \frac{(p^*_k)^2}{p^*_k + \sigma_w^2} + \sigma_v^2 = p^*_k \qquad (24)$$

and by solving

$$p^*_k = \sigma_v^2(1+\sqrt{1+4/R})/2 \qquad (25)$$

a^*_k denotes the one step prediction of a_k which satisfies the relation

$$a^*_{k+1} = \hat{a}_k = a^*_k + K(y_k - a^*_k) \qquad (26)$$

For $\hat{a}_k$ the following difference equation holds

$$\hat{a}_{k+1} = \hat{a}_k + K(y_{k+1} - \hat{a}_k) \qquad (27)$$

K is the steady state Kalman gain which is equal to

$$K = \frac{p^*}{\sigma_w^2 + p^*} \qquad p^* = p_k^* \tag{28}$$

$\tilde{p}$ is then given by

$$\tilde{p} = p^*(1-K) \qquad \tilde{p} = \tilde{p}_k \tag{29}$$

with (25) and (28) one obtains for $\tilde{p}$

$$\tilde{p} = \frac{\sigma_w^2}{1 + \frac{2/R}{1+\sqrt{1+4/R}}} \tag{30}$$

$\tilde{p}$ is like (17) and (18) again a function of the noise ratio R. $\tilde{p}/\sigma_w^2$ tends for R>>1 to 1 and for R<<1 to 0.
In Table 1

$$\sqrt{\tilde{p}/\sigma_w^2}\ ,\ \sqrt{V(n_{opt},L^Q_{opt})/\sigma_w^2}\ ,\ \sqrt{V(L^Q_{opt},L^Q_{opt})/\sigma_w^2}$$

are listed for a range of values of R.

R	$\sqrt{\tilde{p}/\sigma_w^2}$	L^Q_{opt} $(n=n_{opt})$	$\sqrt{V/\sigma_w^2}$	L^Q_{opt} $(n=L_{opt})$	$\sqrt{V/\sigma_w^2}$
10^{-3}	0.176	109	0.135	54	0.190
10^{-2}	0.308	34	0.240	17	0.333
10^{-1}	0.520	10	0.427	5	0.565
0.5	0.707	4	0.637	2	0.791
1	0.786	3	0.745	1	1
2	0.856	2	0.866	1	1
10	0.957	1	1	1	1
10^2	0.995	1	1	1	1
10^3	≈1	1	1	1	1

Table 1: Performance of the three estimators for different noise ratios R

R	$(\sqrt{V/\tilde{p}} - 1)100\%$ with delay		$(\sqrt{V/\tilde{p}} - 1)100\%$ without delay	
10^{-3}	-23.3		7.95	
10^{-2}	-22.1		8.1	
10^{-1}	-17.9		8.6	
0.5	-9.9		11.9	
1	-5.2		27.2	
2	1.2		16.8	
10	4.5	trivial	4.5	trivial
10^2	0.5	L=1	0.5	L=1
10^3	~0		~0	

Table 2: Relative gains of the Kalman filter (steady state) versus the two averagers

Table 2 shows how much can be gained in the steady state if a steady state Kalman filter is used instead of the two averagers. The gain does not exceed 27.2%. For R>>1 and R<<1 the gain approaches zero. It is interesting to notice that the comparison with the best moving averager with time delay is not in favor of the Kalman filter. This is not surprising however, because the Kalman filter is only the best recursive estimator without time delay. To compare the best recursive estimator with time delay with the best moving averager with time delay the optimal fixed lag smoother [5] would have to be determined.

4. OTHER APPLICATIONS

A second applicatiopn of the above averager analysis is the estimation of a constant but unknown parameter, where the measurements are corrupted by a noise that is a sum of a white Gaussian noise source and a random walk like noise source. The estimation error to be analyzed is

$$\tilde{a}_{L|L} = m^L_L - a_0$$

which is a special case of (10) [4].

The determination of n_{opt} and L_{opt} can be extended to other stochastic models for the generation of the time dependency of the unknown parameter $\bar{a}(t)$.

5. CONCLUSIONS

In this article it has been shown, how much the estimation of a slowly varying parameter which is modelled by a random walk can be improved by a steady state Kalman filter compared to a moving averager (with no time delay) whose averaging length has been optimized. For moving averagers with inherent delays between the generation of $\bar{a}(kT)$ and the occurence of its estimate the best estimator has a delay of half of the averaging time period L, and produces better estimates than the considered optimal recursive estimator.

REFERENCES

[1] Papoulis, A., Probability, Random Variables and Stochastic Processes, McGraw Hill 1965.
[2] Keshner, M.S., 1/f-noise Proc. IEEE, Vol.70 March 1980.
[3] Gruber, P., 1/f-noise Generator, Noise in physical systems and 1/f-noise, 1985 North Holland.
[4] Gruber, P., Untersuchung von verschiedenen Verfahren zur Schätzung von Grössen, die durch instationäre Rauschquellen gestört sind. Laborbericht ZL-LB-85/678, LGZ Landis & Gyr Zug AG.
[5] Anderson and B.D.O., Moore, J., Optimal Filtering, Prentice Hall 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N3.9

REAL-TIME MEASUREMENT OF TIME VARYING PROBABILITY DENSITY FUNCTIONS

Danuta RUTKOWSKA

Technical University of Częstochowa, Poland

A nonparametric procedure for the real-time measurement of time varying probability density functions is presented. The procedure is based on the stochastic approximation algorithm, working in a nonstationary environment. Convergence properties are investigated in details.

1. INTRODUCTION

A wide variety of real systems have a nonstationary nature and are described by time-varying probability density functions. Many examples can be found in biomedical, geophysics and chemical problems. In this paper we are interested in measurement of time-varying probability densities by a nonparametric method.

Let X_n be a sequence of independent random variables with unknown probability densities f_n. We shall use a complete orthonormal system g_j, $j=0,1,\ldots$, defined on a set $A \subset R$. The functions f_n, $n=1,2,\ldots$, can be expanded in the orthogonal series

$$f_n(x) = \sum_{j=0}^{\infty} a_{jn}\, g_j(x). \qquad (1)$$

It means that

$$a_{jn} = \int_A f_n(x)\, g_j(x)\, dx$$

$$= E[g_j(X_n)]. \qquad (2)$$

The coefficients a_{jn} are estimated by the Robbins-Monro stochastic approximation method

$$\hat{a}_{j\,n+1} = \hat{a}_{jn} - \gamma_n(\hat{a}_{jn} - g_j(X_{n+1})), \qquad (3)$$

where $\hat{a}_{jo}=0$ for all j and γ_n is a sequence of positive numbers.

We propose the following procedure tracking for f_n

$$\hat{f}_n(x) = \sum_{j=0}^{N(n)} \hat{a}_{jn}\, g_j(x), \qquad (4)$$

where $N(n)$ is a sequence of integers.

In this paper the convergence properties of the algorithm (4) will be investigated using ideas suggested in the previous paper of the author [2].

2. THE MEAN SQUARE ERROR CONVERGENCE

Suppose that

$$|g_j(x)| \leq G_j, \qquad j=0,1,\ldots \qquad (5)$$

for all $x \in A$, where G_j is a sequence of numbers. Define

$$d_n = \sup_{N_1,N_2}\left\{\sum_{j=N_1}^{N_2} E(\hat{a}_{jn}-a_{jn})^2 / \sum_{j=N_1}^{N_2} G_j^2\right\}.$$

The mean square error convergence of procedure (4) is given by the following theorem.

Theorem 1. If

$$d_n^{1/2} \sum_{j=0}^{N(n)} G_j^2 \xrightarrow{n} 0, \qquad (6)$$

then

$$E(\hat{f}_n(x) - f_n(x))^2 \xrightarrow{n} 0$$

at every point $x \in A$, at which

$$\left| \sum_{j=0}^{N(n)} a_{jn}\, g_j(x) - f_n(x)\right| \xrightarrow{n} 0. \qquad (7)$$

Proof. Observe that

$$\hat{f}_n(x) - f(x) = \sum_{j=0}^{N(n)} (\hat{a}_{jn} - a_{jn})\, g_j(x) + \sum_{j=0}^{N(n)} a_{jn}\, g_j(x) - f(x).$$

By Cauchy's inequality, the expectation of the squared first term in the above equality is not greater than

$$\sum_{j=0}^{N(n)} E(\hat{a}_{jn}-a_{jn})^2 \sum_{j=0}^{N(n)} G_j^2 \leqslant d_n \left[\sum_{j=0}^{N(n)} G_j^2\right]^2.$$

Consequently

$$E(\hat{f}_n(x) - f(x))^2 \leqslant 2 d_n \left[\sum_{j=0}^{N(n)} G_j^2\right]^2 + 2\left[\sum_{j=0}^{N(n)} a_{jn}\, g_j(x) - f(x)\right)\Big]^2.$$

The proof has been completed.

3. THE MEAN INTEGRATED SQUARE ERROR CONVERGENCE

The global properties of the procedure (4) are given by the next theorem.

Theorem 2. If $f_n \in L_2$, $n=1,2,\ldots$, and

$$d_n \sum_{j=0}^{N(n)} G_j^2 \xrightarrow{n} 0, \qquad (8)$$

$$\left\| \sum_{j=0}^{N(n)} a_{jn}\, g_j(x) - f_n(x)\right\|_{L_2} \xrightarrow{n} 0, \qquad (9)$$

then

$$\left\| \hat{f}_n(x) - f_n(x)\right\|_{L_2} \xrightarrow{n} 0.$$

Proof. Note that

$$\left\| f_n(x) - f_n(x)\right\|_{L_2}^2 \leqslant 2 \sum_{j=0}^{N(n)} E(\hat{a}_{jn} - a_{jn})^2 + 2\left\| \sum_{j=0}^{N(n)} a_{jn}\, g_j(x) - f_n(x)\right\|_{L_2}^2.$$

Since

$$\sum_{j=0}^{N(n)} E(\hat{a}_{jn} - a_{jn})^2 \leqslant d_n \sum_{j=0}^{N(n)} G_j^2$$

the proof is complete.

4. THE RATE OF THE CONVERGENCE OF d_n

The Robbins-Monro stochastic approximation algorithm (3), working in the nonstationary conditions, was first investigated by Dupač [1]. Using his

arguments and those in [2] we get the rate, at which the sequence d_n converges to zero.

Theorem 3. Let

$$\gamma_n = \delta n^{-r}, \ \delta>0, \ 0<r<1 \qquad (10)$$

$$\sup_{x\in A} |f_{n+1}(x)-f_n(x)| = O(n^{-p}), \ r<p. \qquad (11)$$

Then

$$d_n = O(n^{-s}),$$

where

$$s = \begin{cases} 2(p-r) \text{ for } r \geq 2p/3 \\ r \quad \text{otherwise.} \end{cases}$$

Remark. If

$$f_n(x) = f(x-c_n)$$

and f satisfies the Lipschitz condition, then (11) becomes

$$|c_{n+1}-c_n| = O(n^{-p}).$$

5. EXAMPLES

If A is a real line, we can use a system

$$g_j(x) = (2^j j!\,\pi^{1/2})^{-1/2} e^{-x^2/2} H_j(x),$$

where

$$H_0(x) = 1,$$

$$H_j(x) = (-1)^j e^{x^2}(d^j e^{-x^2}/dx^j), \quad j=1,2,\ldots,$$

are Hermite polynomials. It can be found in Szegö [5] that

$$|g_j(x)| \leqslant \text{const } j^{-1/12}.$$

Therefore conditions (6) and (8) take the forms:

$$d_n^{1/2} \left[N(n)\right]^{5/6} \xrightarrow{n} 0,$$

$$d_n \left[N(n)\right]^{5/6} \xrightarrow{n} 0.$$

If $A=[0,\infty)$ we can apply the Laguerre system

$$g_j(x) = e^{-x/2} L_j(x),$$

where

$$L_0(x) = 1,$$

$$L_j(x) = (j!)^{-1} e^x (d^j e^{-x} x^j/dx^j), \quad j=1,2,\ldots,$$

are Laguerre polynomials. In this case

$$|g_j(x)| \leqslant \text{const } j^{-1/4}$$

(see Szegö [5]), and conditions (6) and (8) become

$$d_n^{1/2} \left[N(n)\right]^{1/2} \xrightarrow{n} 0,$$

$$d_n \left[N(n)\right]^{1/2} \xrightarrow{n} 0.$$

6. FINAL REMARKS

It should be noted that alternative procedures, working in the nonstationary environment, are presented in [4].

The approach based on the orthogonal series is very convenient from the computational point of view (see [3]).

REFERENCES

[1] Dupač, V., A dynamic stochastic approximation method, Ann. Math. Statist.,36, (1965), pp. 1695-1702.
[2] Rutkowska, D., et. al., An orthogonal series estimate of time-varying regression, Annals of the Institute of Statistical Mathematics, vol.35, No.2, A, (1983), pp. 215-228.
[3] Rutkowska, D., Computer based real-time measurement of probability density functions, Proceedings of the 5th International IMEKO Symposium 'Intelligent Measurement', in print.
[4] Rutkowski, L., On nonparametric identification with prediction of time-varying systems, IEEE Trans. Autom. Contr., vol. AC-29, (1984), pp. 58-60.
[5] Szegö, G., Orthogonal Polynomials, Amer. Math. Soc. Coll. Publ., 23, (1959).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EFFICIENT AND ROBUST COVARIANCE LADDER ALGORITHMS FOR LINEAR PREDICTION

Peter Strobach

SIEMENS Information Systems Laboratory
ZT ZTI INF 121, Otto-Hahn-Ring 6
D-8000 München 83, West Germany

ABSTRACT: This paper presents the theory for a new class of numerically robust and efficient least squares (LS) covariance ladder algorithms. Such adaptive algorithms are highly desirable in dynamic systems where the limitation of hardware costs is an issue. The new algorithms are derived by the algebraic method of "generalized residual energies" (GREs) which provides the order recursive updating of the ladder reflection coefficients using exclusively inner products. The computation of residuals or ill-conditioned LEVINSON-type recursions can thus be avoided. This way the new coefficient updating procedures are optimum in the sense of round-off error sensitivity. Two algorithms based on the new formalism are given. This study can ultimately lead to efficient VLSI implementations of adaptive ladder estimation algorithms.

1. INTRODUCTION

The fitting of an autoregressive (AR) model of order p onto the non-stationary time series $\underline{x}(t)$

$$\underline{x}(t) = \left[x(t),x(t-1),\ldots,x(t-L)\right]^T \qquad (1.1)$$

can be performed by the adaptive canonical ladder form determined by the following vector order recursions

$$\underline{e}_0(t) = \underline{r}_0(t) = \underline{x}(t) \qquad (1.2a)$$

$$\underline{e}_m(t) = \underline{e}_{m-1}(t) + K_m^f(t)\underline{r}_{m-1}(t-1) \qquad (1.2b)$$

$$\underline{r}_m(t) = \underline{r}_{m-1}(t-1) + K_m^b(t)\underline{e}_{m-1}(t) \qquad (1.2c)$$

where $\underline{e}_m(t)$ and $\underline{r}_m(t)$ are the forward and backward residual vectors of time step t. The scalars $K_m^f(t)$ and $K_m^b(t)$ have been termed the time-varying forward and backward reflection coefficients of stage m. To meet the LS error criterion

$$E_m(t) = \underline{e}_m^T(t)\underline{e}_m(t) \stackrel{!}{=} \min \qquad (1.3a)$$

$$R_m(t) = \underline{r}_m^T(t)\underline{r}_m(t) \stackrel{!}{=} \min \qquad (1.3b)$$

the reflection coefficients must be adjusted as follows

$$K_m^f(t) = -C_{m-1}(t)/R_{m-1}(t-1) \qquad (1.4a)$$

$$K_m^b(t) = -C_{m-1}(t)/E_{m-1}(t) \qquad (1.4b)$$

The quantities

$$E_{m-1}(t) = \underline{e}_{m-1}^T(t)\underline{e}_{m-1}(t) \qquad (1.5a)$$

$$R_{m-1}(t) = \underline{r}_{m-1}^T(t)\underline{r}_{m-1}(t) \qquad (1.5b)$$

$$C_{m-1}(t) = \underline{e}_{m-1}^T(t)\underline{r}_{m-1}(t-1) \qquad (1.5c)$$

are known as the residual energies of ladder stage m-1. They are defined by inner products of residual vectors. There is quite a number of algorithms available for recursive updating of the ladder reflection coefficients /5,8/. However, different forms of ladder update recursions, which behave identical when implemented with infinite precision, can perform quite differently when finite precision arithmetic is used.

The research presented here was motivated by the experimental work which has been carried out on the analysis of round-off error effects in ladder coefficient update recursions /3,6,7,10,11/. It has been found, that best results in the sense of minimum round-off error sensitivity can be achieved by an algorithm structure which seperates time and order recursions in two independent subalgorithms /12/:

- The tracking of input data is performed via the time-recursively updated covariance matrix.
- The ladder reflection coefficients are computed by a numerically well behaved pure order recursive ladder algorithm (PORLA) based on GREs which are initialized from the data covariance matrix.

Such a "distributed" algorithm structure has several advantages like

- Mixed precision computations can be applied to obtain a higher accuracy of the time-recursively updated covariance matrix.
- Arbitrary recursive windowing techniques /1,13/ can be used to control the tracking behaviour.
- The updating of the ladder reflection coefficients must occur only each M-th time step (parameter update rate M).

2. THE CONCEPT OF GREs

This section constitutes the new algebraic method of GREs which ultimately leads to the desired pure order recursive ladder algorithm (PORLA). The GREs are defined as matrix forms of residual energies

$$E_{m,i,j}(t) = \underline{e}_m^T(t-i)\underline{e}_m(t-j) \qquad (2.1a)$$

$$R_{m,i,j}(t) = \underline{r}_m^T(t-1-i)\underline{r}_m(t-1-j) \qquad (2.1b)$$

$$C_{m,i,j}(t) = \underline{e}_m^T(t-i)\underline{r}_m(t-1-j) \qquad (2.1c)$$

Substituting the order recursions of shifted residual vectors (1.2b,c) into the definitions of the GREs (2.1a,b,c) gives

$$E_{m,i,j}(t) = \underline{e}^T_{m-1}(t-i)\underline{e}_{m-1}(t-j) + \quad (2.2)$$
$$+ K^f_m(t-i)\underline{r}^T_{m-1}(t-1-i)\underline{e}_{m-1}(t-j) +$$
$$+ K^f_m(t-j)\underline{e}^T_{m-1}(t-i)\underline{r}_{m-1}(t-1-j) +$$
$$+ K^f_m(t-i)K^f_m(t-j)\underline{r}^T_{m-1}(t-1-i)\underline{r}_{m-1}(t-1-j)$$

Similar matrix order recursions are obtained for updating the GREs $R_{m,i,j}(t)$ and $C_{m,i,j}(t)$. Clearly, the right hand side of (2.2) can be expressed in terms of the GREs of stage m-1

$$E_{m,i,j}(t) = E_{m-1,i,j}(t) + \quad (2.3)$$
$$+ K^f_m(t-i)C_{m-1,j,i}(t) + K^f_m(t-j)C_{m-1,i,j}(t) +$$
$$+ K^f_m(t-i)K^f_m(t-j)R_{m-1,i,j}(t)$$

Analogous recursions exist in case of $R_{m,i,j}(t)$ and $C_{m,i,j}(t)$. The ladder reflection coefficients can alternatively be defined in terms of GREs

$$K^f_m(t) = - C_{m-1,0,0}(t)/R_{m-1,0,0}(t) \quad (2.4a)$$

$$K^b_m(t) = - C_{m-1,0,0}(t)/E_{m-1,0,0}(t) \quad (2.4b)$$

The order recursions of GREs together with (2.4 a,b) already establish a closed recursion for the computation of PORLA. Substituting the initialization of residual vectors (1.2a) into the definition of the GREs (2.1a,b,c) yields the initialization scheme of the GREs at stage zero of PORLA

$$E_{0,i,j}(t) = W_{i,j}(t) \quad (2.5a)$$
$$R_{0,i,j}(t) = W_{i+1,j+1}(t) \quad (2.5b)$$
$$C_{0,i,j}(t) = W_{i,j+1}(t) \quad (2.5c)$$

where $W_{i,j}(t) = \underline{x}^T(t-i)\underline{x}(t-j)\ \ 0 \leqslant i,j \leqslant p \quad (2.6)$

is the covariance matrix of input data. In the following we investigate two interesting properties of GREs which result in substantial computational savings. First it can be seen from the definitions of the GREs (2.1a,b,c) that the GREs $E_m(t)$ and $R_m(t)$ are symmetric

$$E_{m,i,j}(t) = E_{m,j,i}(t) \quad (2.7a)$$
$$R_{m,i,j}(t) = R_{m,j,i}(t) \quad (2.7b)$$
$$\text{but:}\quad C_{m,i,j}(t) \neq C_{m,j,i}(t) \quad (2.7c)$$

Next we find that the residual vectors for successive values of t are shifted versions of each other. As these vectors build up the GREs, it is expected, that subsequent GREs are also shifted versions of each other. Indeed, using MORFs rule yields the following useful time-shift properties of GREs which can be conveniently written as

$$E_{m,i+1,j+1}(t) = E_{m,i,j}(t-1) \quad (2.8a)$$
$$R_{m,i+1,j+1}(t) = R_{m,i,j}(t-1) \quad (2.8b)$$
$$C_{m,i+1,j+1}(t) = C_{m,i,j}(t-1) \quad (2.8c)$$

Evaluating the order recursions of GREs under consideration of the symmetric property (2.7a,b) and the time-shift property (2.8a,b,c) yields the following efficient $O(p^2)$ true LS vector order recursions

$$E_{m,0,j}(t) = E_{m-1,0,j}(t) + \quad (2.9a)$$
$$+ K^f_m(t)C_{m-1,j,0}(t) + K^f_m(t-j)C_{m-1,0,j}(t) +$$
$$+ K^f_m(t)K^f_m(t-j)R_{m-1,0,j}(t)$$

$$R_{m,0,j}(t) = R_{m-1,0,j}(t-1) + \quad (2.9b)$$
$$+ K^b_m(t-1)C_{m-1,0,j}(t-1) + K^b_m(t-1-j)C_{m-1,j,0}(t-1) +$$
$$+ K^b_m(t-1)K^b_m(t-1-j)E_{m-1,0,j}(t-1)$$

$$C_{m,0,j}(t) = C_{m-1,0,j+1}(t) + \quad (2.9c)$$
$$+ K^f_m(t)R_{m-1,0,j+1}(t) + K^b_m(t-1-j)E_{m-1,0,j+1}(t) +$$
$$+ K^f_m(t)K^b_m(t-1-j)C_{m-1,j+1,0}(t)$$

$$C_{m,j+1,0}(t) = C_{m-1,j,0}(t-1) + \quad (2.9d)$$
$$+ K^f_m(t-1-j)R_{m-1,0,j}(t-1) + K^b_m(t-1)E_{m-1,0,j}(t-1) +$$
$$+ K^f_m(t-1-j)K^b_m(t-1)C_{m-1,0,j}(t-1)$$

For a more detailed presentation of this new LS ladder algorithm including a flowchart and a computer program see /12,13/.

3. TWO COVARIANCE LADDER ALGORITHMS

A closer look at the vector order recursions (2.9a,b,c,d) indicates, that the true LS ladder recursions still suffer from high storage costs of $O(p^2)$ required for saving of previous recursion vectors and reflection coefficients. Many applications of adaptive signal processing, however, do not need a true LS solution. Assuming stationarity, the GREs show some useful properties leading to much simpler implementation schemes. We introduce three basic approximations which greatly facilitate the development of two fast covariance ladder algorithms. The loss in performance due to these approximations is minor in most applications.

APPROXIMATION 1 (piecewise constant K's)
In case of a piecewise stationary process the reflection coefficients can be assumed to be piecewise constant

$$K^f_m(t-i) = K^f_m(t) \qquad 0 \leqslant i,j \leqslant p \quad (3.1a)$$
$$K^b_m(t-j) = K^b_m(t) \quad (3.1b)$$

The stack of previous reflection coefficients can be omitted using this approach.

APPROXIMATION 2 (Toeplitz structure of GREs)
Assuming stationarity the GREs exhibit a Toeplitz structure

$$E_{m,i+1,j+1}(t) = E_{m,i,j}(t) \quad (3.2)$$

The same is true for $R_{m,i,j}(t)$ and $C_{m,i,j}(t)$.

Obviously the Toeplitz-property of GREs (3.2) can serve as a substitute of the true LS time-shift property of GREs (2.8a,b,c). The Toeplitz approach requires only a storage amount of $O(p)$

in contrast to the true LS time-shift property which requires a storage amount of $O(p^2)$.

APPROXIMATION 3 (BURGs method)
In those applications, where the stability of the all-pole model must be guaranteed (e.g. parametric data compression), one is often interested in a PARCOR-model of the observed process. The most involved method of computing the PARCOR-coefficient $K_m(t)$ is the so-called harmonic-mean-method (also referred as BURGs method /2,8/)

$$K_m(t) = 2K_m^f(t)K_m^b(t)/(K_m^f(t) + K_m^b(t)) \qquad (3.3)$$

The right hand side of (3.3) can be expressed in terms of GREs

$$K_m(t) = -2C_{m-1,0,0}(t)/(E_{m-1,0,0}(t)+R_{m-1,0,0}(t)) \qquad (3.4)$$

ALGORITHM 1
The algorithm 1 is obtained by substituting approximation 1 (3.1a,b) and approximation 2 (3.2) into the LS vector order recursions (2.9a,b,c,d). We shall omit the algebraic manipulations and restrict ourselves to the flowchart of this algorithm, where we use the following simplified notations for convenience

$$E_{m,0,j}(t) = E_j;\ R_{m,0,j}(t) = R_j;\ C_{m,0,j}(t) = C_j$$
$$C_{m,j,0}(t) = C_j^*$$

FOR j=0,1,...,p-1 initialize:

$$\begin{bmatrix} E_j = W_{0,j} & C_j = W_{0,j+1} \\ R_j = W_{1,j+1} & C_j^* = W_{1,j} \end{bmatrix}$$

$$K_1^f = -C_0/R_0 \qquad K_1^b = -C_0/E_0$$

FOR m=1,2,...,p-1

FOR j=0,1,...,p-m-1

$$E_j = E_j + K_m^f(C_j + C_j^*) + K_m^{f\,2}R_j$$
$$R_j = R_j + K_m^b(C_j + C_j^*) + K_m^{b\,2}E_j$$
$$C_j = C_{j+1} + K_m^f R_{j+1} + K_m^b E_{j+1} + K_m^f K_m^b C_{j+1}^*$$

$$C_0^* = C_0$$

FOR j=0,1,...,p-m-2

$$C_{j+1}^* = C_j^* + K_m^f R_j + K_m^b E_j + K_m^f K_m^b C_j$$

$$K_{m+1}^f = -C_0/R_0 \qquad K_{m+1}^b = -C_0/E_0$$

Table 3.1: Summary of covariance ladder algorithm 1. The recursion vectors E, R, C and C* need to be stored twice to avoid "overwriting". One might take advantage of identical expressions to reduce the total number of operations.

Algorithm 1 refers to the true ladder form using different reflection coefficients in the forward and backward predictor. Therefore algorithm 1 can track time-varying parameters very rapidly. We recommend to use this algorithm in those applications, where one is interested in the mathematical model of the process or the generating system, e.g., in case of failure detection applications where the residual energy $E_p(t)$ is used by any statistical detector /19/. Note that the stability of the all-pole model is not guaranteed by algorithm 1.

ALGORITHM 2
The algorithm 2 is obtained from algorithm 1 by additional application of approximation 3 (3.4). The stability of the all-pole model is always guaranteed by this algorithm, thus making it highly attractive for data compression applications. Algorithm 2 performs very similar to the well-known covariance ladder algorithms proposed by MAKHOUL /8/ and CUMANI /4/. Our new approach, however, requires much fewer computations, has a simpler implementation scheme and offers much better numerical properties.

FOR j=0,1,...,p-1 initialize:

$$\begin{bmatrix} E_j = W_{0,j} & C_j = W_{0,j+1} \\ R_j = W_{1,j+1} & C_j^* = W_{1,j} \end{bmatrix}$$

$$K_1 = -2C_0/(E_0 + R_0)$$

FOR m=1,2,...,p-1

FOR j=0,1,...,p-m-1

$$E_j = E_j + K_m(C_j + C_j^*) + K_m^2 R_j$$
$$R_j = R_j + K_m(C_j + C_j^*) + K_m^2 E_j$$
$$C_j = C_{j+1} + K_m(R_{j+1} + E_{j+1}) + K_m^2 C_{j+1}^*$$

$$C_0^* = C_0$$

FOR j=0,1,...,p-m-2

$$C_{j+1}^* = C_j^* + K_m(R_j + E_j) + K_m^2 C_j$$

$$K_{m+1} = -2C_0/(E_0 + R_0)$$

Table 3.2: Summary of covariance ladder algorithm 2. The recursion vectors E, R, C and C* need to be stored twice to avoid "overwriting". Exploiting identical expressions can yield further reductions of the computational costs.

4. CONCLUSIONS

The problem of constructing efficient and numerically robust covariance ladder algorithms for implementation in numerically restricted enviroments has been adressed in this paper. One possible solution of this implementation problem is the separation of time and order recursions into two independent subalgorithms facilitating mixed precision computations. This algorithmical concept was developed from the key idea of generalized residual energies (GREs).
The true LS ladder algorithm has been derived by means of GREs and it has been shown, that two efficient covariance ladder algorithms can be deduced from the exact solution by some straightforward approximations. Interestingly algorithm 2 represents the widely used MAKHOUL covariance ladder algorithm which can now be computed by much more efficient and numerically robust recursions. Furthermore fast algorithms for finite impulse response (FIR) system identification have been derived by means of GREs. This is the subject of another paper /16/. The interested reader is referred to /12-18/ for a detailed discussion of the new ladder methods.

REFERENCES

/1/ T.P. Barnwell, "Recursive windowing for generating autocorrelation coefficients for LPC analysis", IEEE Trans. on ASSP, ASSP 29(5), pp. 1062-1066, 1981.

/2/ J. Burg, "Maximum entropy spectral analysis", Ph.D. dissertation, Stanford Univ., Stanford, CA, 1975.

/3/ C. Caraiscos and B. Liu, "A roundoff error analysis of the LMS adaptive algorithm", IEEE Trans. on ASSP, ASSP-32(1), pp. 34-41, 1984.

/4/ A. Cumani, "On a covariance lattice algorithm for linear prediction", Proc. Int. Conf. on ASSP, Paris, pp. 651-654, 1982.

/5/ D.T.L. Lee, "Canonical ladder form realizations and fast estimation algorithms", Ph.D. dissertation, Stanford Univ., Stanford, CA, 1980.

/6/ F. Ling and J. Proakis, "Numerical accuracy and stability: Two problems of adaptive algorithms caused by round-off error", Proc. Int. Conf. on ASSP, San Diego, pp. 30.3.1-30. 3.4, 1984.

/7/ F. Ling, D. Manolakis and J. Proakis, "New forms of LS lattice algorithms and an analysis of their round-off error characteristics", Proc. Int. Conf. on ASSP, Tampa, 1985.

/8/ J. Makhoul, "Stable and efficient lattice methods for linear prediction", IEEE Trans. on ASSP, ASSP-25 (5), pp. 423-428, 1975.

/9/ J. Markel and A.H. Gray, "Round-off noise charactersitics of a class of orthogonal polynomials", IEEE Trans. on ASSP, ASSP-23, pp. 473-486, 1975.

/10/ C. Samson and V.U. Reddy, "Fixed-point error analysis of the normalized ladder algorithm", IEEE Trans. on ASSP, ASSP-31(5), pp. 1177-1191, 1983.

/11/ E. Satorius, et al., "Fixed-point implementation of adaptive digital filters", Proc, Int. Conf. on ASSP, Boston, pp. 33-36, 1983.

/12/ P. Strobach, "Pure order recursive least squares ladder algorithms", IEEE Trans. on ASSP, August issue, 1986.

/13/ P. Strobach, "Efficient covariance ladder algorithms for finite arithmetic applications", SIGNAL PROCESSING, under review, 1986.

/14/ P. Strobach, "New forms of least squares lattice algorithms and a comparison of their round-off error characteristics", Proc. Int. Conf. on ASSP, Tokyo, 1986.

/15/ P. Strobach, "Robust least squares covariance ladder algorithms for vector autoregressive processes", IEEE Trans. on Automatic Control, under review, 1986.

/16/ P. Strobach, "A fast recursive approach to FIR system identification", this volume.

/17/ P. Strobach, "A new class of least squares covariance ladder estimation algorithms", Proc. 19th Annual Asilomar Conf. on Circuits, Systems and Computers, Monterey, CA, 1985.

/18/ P. Strobach, "Schnelle adaptive Algorithmen zur ordnungsrekursiven Kleinste-Quadrate-Schätzung autoregressiver Parameter", Ph.D. dissertation, Bundeswehr Univ. Munich, West Germany, 1985.

/19/ A.S. Willsky and H.L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems", IEEE Trans. on Automatic Control, AC-21, pp. 108-112, 1976.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ANALYSIS AND DETECTION OF KNOCKING SIGNALS FROM SPARK IGNITION ENGINES*

N. Härle, J.F. Böhme

Lehrstuhl für Signaltheorie, Ruhr-Universität Bochum
Universitätsstr. 150, 4630 Bochum 1, West Germany

A global analysis method (the estimation of a higher order statistical moment of the periodogram) is developed and applied to structural vibration signals. Furthermore, a model which is planned to be used for the detection is found and verified. The model consists of a sum of damped oscillations with shifting frequencies and corresponds to the cavity resonances in the combustion chamber.

1. INTRODUCTION

1.1. Significance of the Knocking Problem

Since the beginning of spark ignition engines, knocking has been known as an abnormal combustion of the gas mixture. It can lead to damages of the engine particularly when the engine is knocking at high speed. Knocking is one important factor limiting the efficiency of an engine /1/ and its understanding is therefore of great interest.

1.2. Basic Physical Processes

The exact mechanisms which lead to knocking and which occur during the knocking are not well known yet. Roughly, the following processes occur and may lead to knocking. During the combustion, the temperature and the pressure of the still unburned gas mixture in the combustion chamber increase. Prereactions in the unburned section of the mixture develop and advance. Knocking can occur if the still unburned part of the mixture autoignites when the temperature and pressure have exceeded a critical point and the prereactions have advanced far enough /2/. A shock wave may result leading to possible damages in the combustion chamber /3/.

1.3. The Problem

There are several problems connected with the knocking such as choice of materials, the geometry of the combustion chamber etc. Here, only the measurement of the knocking phenomenon is being taken into consideration. The aim of this work is to find a detection method which is sensitive particularly to light knocking.

2. THE SIGNALS AND GLOBAL METHODS

2.1. Structural Vibration Signals

The basis for the planned detection scheme will be the structural vibration signal which is simple to measure because of the ease of use of the accelerometers. However, it has a very poor signal to noise ratio (S/N) specially for high rotation speed. The noise is generated by several sources, such as valves, gearings and bearings in particular. Figure 1 shows the structural vibrations of an unusually heavy knocking cycle to illustrate the here called "knocking signal".

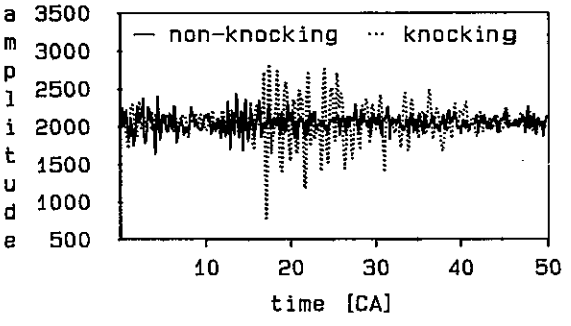


Figure 1. Typical structural vibration signals.

2.2. Cylinder Pressure Signals

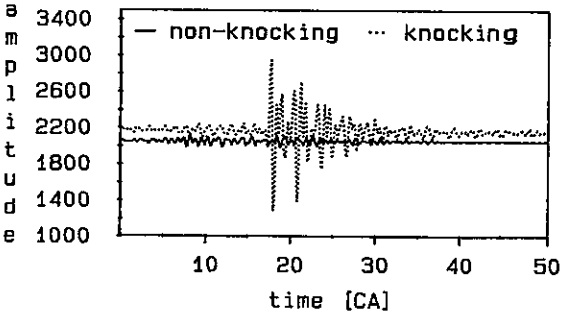


Figure 2. Typical high pass filtered cylinder pressure signals.

*sponsered by the Stiftung Volkswagenwerk

The cylinder pressure is recorded and analysed only for reference purposes. It is very sensitive to knocking and its measurement is performed very close to the place where the knocking occurs. The low frequency part of this signal includes the compression and the expansion of the gas and is removed by high pass filtering (figure 2).

2.3. Signal Power

The power in several ignition cycles for the structural vibrations is measured and plotted in figure 3 as a histogram. The high power amplitudes for the knocking cycles can easily be seen in figure 3. Besides, the average power increases also due to the different operation mode when changing the ignition angle.

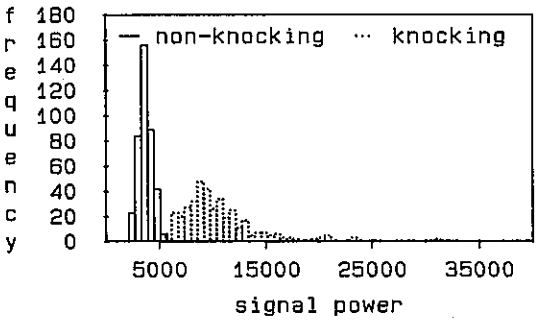


Figure 3. Histograms for the signal power of several ignition cycles.

2.4. Knocking Values

From the histograms in figure 3 it follows that normalized characteristic values for the probability distribution can be taken as measures for the knocking. The value

$$k = \frac{(E(X-EX)^3)^{1/3}}{EX} \qquad (2.4/1)$$

was experienced to be a sensitive measure for the knocking /4/ and is here called "knock value" (X - random variable, E - expected value operator). It was estimated for different ignition angles, e.g. knocking status of the engine, and results in -0.08, 0.14, 0.20, 0.10, 0.24, 0.25, 1.07. for the vibration signal. The knocking value increases very slowly for non-knocking and, in this example, reaches 1.07 for the first detectable knocking. For the cylinder pressure with its far better S/N, the values 0.61, 0.63, 0.50, 1.27, 1.80, 2.70, 2.46 are estimated and, therefore, indicate knocking earlier (at 1.27).

It follows that generally the knocking value is approximately constant for non-knocking and increases with the rate of knocking, but it decreases when the knocking becomes too strong. Knock values were first discussed in /5/.

2.5. Power Spectra

Usually, the average of periodograms calculated from intervals corresponding to the individual ignition cycles is used to find the spectral characteristics of the knocking signal. Averages for different ignition angles are compared and the differences are interpreted as "knocking frequencies".

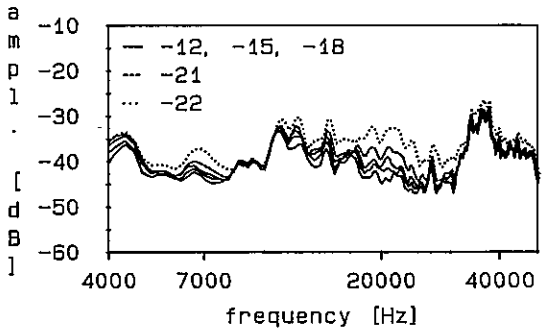


Figure 4. Estimated spectra for different ignition angles (unit: °CA - degree crank angle).

Figure 4 shows the resulting estimation for the power spectra using the vibration signals. But, differences in power at single frequencies may again be caused by the different operation modes. Therefore, a normalized measure is considered such as the knock value.

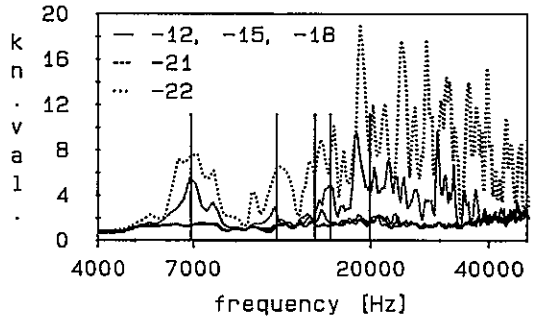


Figure 5. Knock value spectra for different ignition angles (in °CA).

The knock value is applied to the periodograms for individual frequencies and the result is shown in figure 5. The knock value is approximately constant equal to 1.26 for the "non-knocking mode" as referred to the asymptotic Chi-squared distribution of the periodogramm. Hence, the knock value of the periodograms shows to be far more sensitive than the mean.

3. MODELING AND VERIFICATION

3.1. Purpose of Modeling

A good performance of a detection can generally be achieved if a proper model of the measured signal is used. Therefore, we will describe in this section a modelisation of the knocking process.

Here the modelling is closely related to the physical nature of the phenomenon in such a way as to bring in as much priori information about the physical process as possible and therefore improve the performance of the detector.

3.2. Cavity Resonances

3.2.1. Resonances in an Ideal Cylinder

Combustion chamber resonances are one consequence of knocking. An accustical model for the resonances can be used because the oscillation amplitudes are presumed to be far smaller than the absolute pressure /6/. In an ideal cylinder which has accustically hard walls and which is filled with homogenous gas, the resonance frequencies amount to /7/

$$f_{m,n,q} = c \left(\frac{\eta_{m,n}^2}{(2\pi R)^2} + \frac{q^2}{(2h)^2} \right)^{1/2} \qquad (3.2.1/1)$$

with $J_m'(\eta_{m,n}) = 0$ (3.2.1/2)

and $c = c_o T^{1/2}$ (3.2.1/3)

if

J_m' - first derivative of the m-th order bessel function of the first kind,
n - n-th zero,
R - radius of the cylinder,
c_o - phase velocity constant,
T - temperature of the gas,
h - height of the cylinder,
q - mode number for the heigth direction of the cylinder.

The movement of the piston results for $q \neq 0$ in a very fast shift in resonance frequencies which are difficult to measure. Thus, q is set to 0 and the resonance frequencies are simplified to

$$f_{m,n} = \frac{c\eta_{m,n}}{2\pi R} \qquad (3.2.1/4)$$

Figure 6 illustrates some resonance modes.

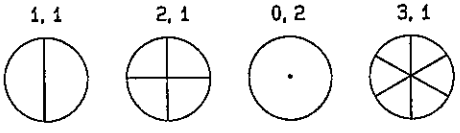


Figure 6. Positions where the gas velocity equals to zero for the modes with lower frequencies

3.2.2. Resonances in the Combustion Chamber

The shape of the combustion chamber of an engine differs much from the shape of an ideal cylinder, particularly if the piston is near the top point. However, if the piston is moving away from the top point the shape of the combustion chamber reaches more and more the shape of a cylinder. Therefore, the resonance frequencies are shifting beginning at some starting values towards the resonance frequencies of an ideal cylinder. The course of the individual resonance frequencies depend on the type of the individual mode as shown in /7/ where resonance frequencies in Diesel engines were investigated.

Shifts in frequency may also result due to temperature changes (3.2.1/3). For an engine, the temperature in the gas drops after the knocking due to the expansion of the gas which is caused by the movement of the piston. There is no energy supply and the gas can be asssumed to be approximately homogeneous after the knocking took place because the whole gas is burned very shortly after the autoignition.

The model for the knocking signal is therefore chosen to a sum of damped oscillations with certain frequencies as well as certain shifts of the frequencies.

3.3. Verification

3.3.1. Resonance Frequencies in the Knock Value Spectrum

The specific frequencies which are presumed to be the lower resonance frequencies are shown as vertical lines in figure 5. The lowest frequency with 6900Hz would result in an average temperature of approximately 2660°K of the gas when $c_o = 21\ m/(sK^{1/2})$ and is therefore a resonable value for the gas temperature in the combustion chamber of the engine.

3.3.2. Verification of the Shifts in Frequency

Damped oscillations with constant frequencies are fitted to a short part of the knocking signal data of which it is presumed that its frequencies are approximately constant. The window which extracts the examined part of the data is shifted from the beginning, step by

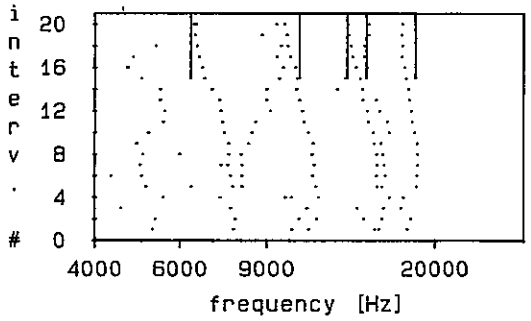


Figure 7. Course of resonance frequencies in the cylinder pressure signal

step to the end of the knocking signal. All the estimations were performed using Prony's method /8/.

The estimates for the frequencies are plotted as single points in figure 7 for the cylinder pressure. The analysis intervals are moved from early (bottom) to late (top). The shifting frequencies can easily be recognized in the cylinder pressure signal.

The resonance frequencies are plotted in figure 7 again as vertical lines. They indicate that the relative values of the resonance frequencies for later intervals are almost identical with those of the ideal cylinder.

The spectra are calculated using the parameters estimated by the Prony method and plotted in figure 8 and 9. They are plotted for the different intervals one above the other without any frequency shifts between them. They appear in reversed order which means that the curve in front corresponds to the latest interval. The shift of the peaks are therefore only due to the frequency shifting of the resonance frequencies. The figures show that the shifting resonances can be measured in the structural vibrations, too.

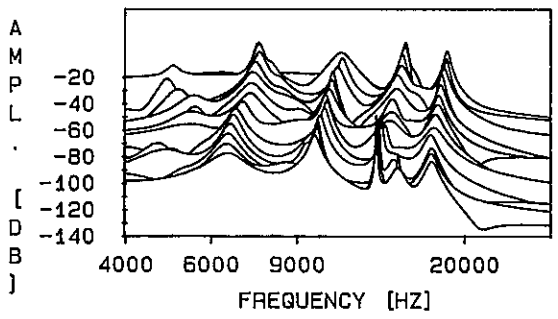


Figure 8. Estimated spectra via Prony's method for several crank angle intervals from the cylinder pressure signal.

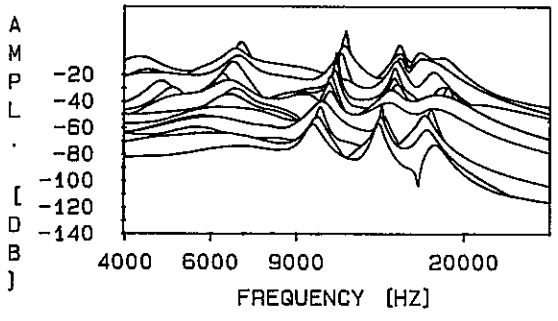


Figure 9. Estimated spectra via Prony's method for several crank angle intervals from the vibration signal.

4. CONCLUSIONS

Knock values applied to periodograms have been found to be a relatively easy but powerful way to analyse the power which is affected by knocking at certain frequencies. It is shown that the resonance oscillations of the gas in the combustion chamber due to knocking result in oscillations in the structural vibrations. These oscillations are not constant in frequency. Moreover, the frequency shifts are due to the change in geometry of the combustion chamber and the change in temperature of the gas. For later time intervals when the piston is away from the top point the geometry of a cylinder is appropriate.

ACKNOWLEDGEMENTS

We like to thank Peter Paris who helped performing a great part of the programming and the ARAL Research which put their test bed for the engine to our disposal.

REFERENCES

/1/ S. Rhode, Closed Loop Knock Control for SI-Engines, in: Proceedings of "Knocking of Combustion Engines", Wolfsburg (1981).

/2/ A. Donaud, Modeling the Knocking Phenomenon in Engines, in: Proceed. of "Knocking of Combustion Engines", Wolfsburg (1981).

/3/ G. Betz, J. Ellermann, Knock-Related Piston Damage in Gasoline Engines, Knock Measurement Technique, Aspects of Piston Failure Preventation, in: Proceedings of "Knocking of Combustion Engines", Wolfsburg (1981).

/4/ N. Härle, J.F. Böhme, Analysis of the Structural Vibrations of Spark Ignition Engines in Order to Detect Knocking and Connections Between Knocking Signals and Resonance Oscillations in the Combustion Chamber, in: Fortschritte der Akustik (DAGA'85), pp. 265-268.

/5/ R. Klein, Analyse von Zylinderdruck und Körperschall von Ottomotoren zum Nachweis des Klopfens, in: Motortechnische Zeitung (40/1979), 167-169.

/6/ R. Hickling, F.H.K. Chen, D.A. Feldmaier, Pressure Pulsations in Engine Cylinders, in: R. Hickling, M.M. Kamal, Engine Noise - Excitation, Vibration and Radiation (Plenum Press, New York-London, 1982).

/7/ F. Pischinger, K. Schmillen, M. Schneider, Untersuchung des Mechanismus der Anregung von Resonanzen bei der dieselmotorischen Verbrennung, in: Kolloquium des Sonderforschungsbereiches 224 "Motorische Verbrennung" (1985), pp. 214-228.

/8/ S.M. Kay, S.L. Marple, Spectrum Analysis - A Modern Perspective, in: Proceedings of the IEEE (1982), pp. 1380-1419.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON ESTIMATION OF ENTROPY AND MUTUAL INFORMATION OF CONTINUOUS DISTRIBUTIONS

R. MODDEMEIJER

Technische Hogeschool Twente, Afd. Elektrotechniek
Postbus 217
7500 AE Enschede, The Netherlands

Mutual information is used in a procedure for estimation of time-delays between electroencephalogram (EEG) signals. Our presentation deals with a histogram method to estimate entropy and mutual information from signal samples and discusses its accuracy. The results are compared with earlier work.

1. INTRODUCTION

For the estimation of time-delays between electroencephalogram (EEG) signals, a procedure using estimates of mutual information (AAMI-method, AAMI = Average Amount of Mutual Information) is used. This procedure introduced by Mars [1], to locate epileptic foci in animals and patients, has produced promising results, e.g. [2,3]. Our presentation deals with a method to estimate entropy and mutual information from signal samples using a histogram method. It is assumed that X- and Y-signals are disturbed responses to a common cause, originating from stationary stochastic processes. The probability density of the pair $\{x(t), y(t-\tau)\}$ is $f_{xy}(x,y;\tau)$. The time-shift τ which maximizes the mutual information I_{xy} is regarded as the delay of X with respect to Y. In our analysis τ does not play a role, so we suppress it.
The definitions of entropy and mutual information go back to Shannon [4] and these are for continuous distributions:

$$1.1 \quad I_{xy} = H_x + H_y - H_{xy}$$

$$1.2 \quad H_{xy} = -\int_{-\infty}^{\infty}\int f_{xy}(x,y) \log f_{xy}(x,y)\, dx\, dy$$

and similar for H_x and H_y. We assume base "e" of the logarithm so the unit of measurement is "nat". In his work Mars estimated $f_{xy}(x,y)$ by a kernel method [5,6] and he used this estimate to calculate the estimate $\hat{I}_{xy}$. The main disadvantage of this method is its complexity, especially because Mars determined the optimal kernel-width iteratively.
For simplicity we discretize X and Y (histogram method). We also estimate the density and determine $\hat{I}_{xy}$ from this estimate. The problem of choosing a kernel-width is replaced by the problem of choosing a rectangular grid which devides the x-y plane in cells. As a step in the evaluation of this method we calculate bias variance of our estimators.

2. ESTIMATOR

We define a rectangular grid in the x-y plane by lines parallel to the axes. In this way a part of the x-y space is divided in (I×J) equally sized ($\Delta x \times \Delta y$) cells with coordinates (i,j). Instead of the probability functions $f_{xy}(x,y)$ we define a probability density p_{ij} of observing a sample in a cell (i,j). The observed number of samples in cell (i,j) is k_{ij}: the total number of samples is N. Row and column sums are noted as $k_{i.} = \sum k_{ij}$ and $k_{.j} = \sum k_{ij}$. If the samples are assumed to be independent, then the observations are multinomially distributed with expectation:

$$2.1 \quad E\{k_{ij}\} = \tilde{k}_{ij} = N\, p_{ij}$$

and covariance:

$$2.2 \quad \mathrm{Cov}\{k_{ij},k_{mn}\} = Np_{ij}(1-p_{ij}) \quad \text{if } i=m \text{ and } j=n$$
$$= -N\, p_{ij}\, p_{mn} \quad \text{all others}$$

For the time being, it is assumed that entropy and mutual information can be estimated properly by:

$$2.3 \quad \hat{I}_{xy} = \hat{H}_x + \hat{H}_y - \hat{H}_{xy}$$

$$2.4 \quad \hat{H}_x = \sum_i -\frac{k_{i.}}{N} \log \frac{k_{i.}}{N} + \log \Delta x$$

$$2.5 \quad \hat{H}_{xy} = \sum_{ij} -\frac{k_{ij}}{N} \log \frac{k_{ij}}{N} + \log(\Delta x \Delta y)$$

3. BIAS

Three sources of bias can be distinguished: a) limited integration area, b) non-linear transformation of density estimates to local contributions to the entropy and c) finite resolution.

a) Although the integration variables of (1.2) run from $-\infty$ to ∞, the grid only reaches out to a finite area. For a binormal distribution with $|x|<3\sigma_x$ and $|y|<3\sigma_y$ this leads to an error in I_{xy} of maximal $0.011\, I_{xy}+0.019\,|\rho|$ with $\rho = \mathrm{Cov}\{x,y\}/\sigma_x\sigma_y$ [7]. This error is small compared to the errors caused by b) and c).
b) The entropy estimator (2.5) is in fact a non linear function of probability estimates k_{ij}/N. Because in general $E\{-a \log a\} \leq E\{a\} \log E\{a\}$ if $a>0$, we expect a biased estimate. We calculate the bias by a Taylor expansion of the terms of (2.5) in $k_{ij}=\bar{k}_{ij}$

$$3.1\quad \hat{H}_{xy} = \sum_{ij}\sum \Big\{ -\frac{\bar{k}_{ij}}{N}\log\frac{\bar{k}_{ij}}{N} - \Big(\frac{1}{N}+\frac{1}{N}\log\frac{\bar{k}_{ij}}{N}\Big)\cdot(k_{ij}-\bar{k}_{ij}) - \frac{1}{2N\bar{k}_{ij}}(k_{ij}-\bar{k}_{ij})^2\Big] + O\Big(\frac{1}{N^2}\Big) + \log(\Delta x\Delta y)$$

The expectation of (3.1), with substitution of (2.1) and (2.2) equals

$$3.2\quad E\{\hat{H}_{xy}\} = \sum_{ij}\sum -\frac{\bar{k}_{ij}}{N}\log\frac{\bar{k}_{ij}}{N} - \frac{IJ-1}{2N} + O\Big(\frac{1}{N^2}\Big) + \log(\Delta x\Delta y)$$

and similar for $E\{\hat{H}_x\}$ and $E\{\hat{I}_{xy}\}$.
c) Also without the statistical effect in b) the finite resolution leads to a bias in $\hat{I}_{xy}$ because $f_{xy}(x,y)$ is not constant within a cell. We approximate locally (for one cell) the probability function by

$$3.3\quad f_{xy}(x,y) \approx f_{xy}(x_i,y_j) + \frac{\partial f_{xy}(x_i,y_j)}{\partial x}(x-x_i) + \frac{\partial f_{xy}(x_i,y_j)}{\partial y}(y-y_j)$$

in which x_i and y_j represent the center of cell (i,j). According to (3.2) the contribution of one cell to the entropy estimate is approximated by

$$3.4\quad E\{\hat{h}_{ij}\} = -\frac{\bar{k}_{ij}}{N}\log\frac{\bar{k}_{ij}}{N} + \frac{1}{IJ}\log(\Delta x\Delta y)$$

and it should be

$$3.5\quad h_{ij} = -\iint_{\text{cell}} f_{xy}(x,y)\log f_{xy}(x,y)\,dx\,dy$$

The difference is caused by the finite resolution. We approximate the integrant of (3.5) by a Taylor expansion in $f_{xy}(x,y)=f_{xy}(x_i,y_j)$

$$3.6\quad h_{ij} \approx -\iint_{\text{cell}} \Big[f_{xy}(x_i,y_j)\log f_{xy}(x_i,y_j) + (1+\log f_{xy}(x_i,y_j)).(f_{xy}(x,y)-f_{xy}(x_i,y_j)) + \frac{1}{2f_{xy}(x_i,y_j)}(f_{xy}(x,y)-f_{xy}(x_i,y_j))^2\Big]dxdy$$

We substitute (3.3) in (3.6) and integrate

$$3.7\quad h_{ij} \approx -f_{xy}(x_i,y_j)\log f_{xy}(x_i,y_j)\,\Delta x\,\Delta y - \frac{1}{24 f_{xy}(x_i,y_j)}\left[\left\{\frac{\partial f_{xy}(x_i,y_j)}{\partial x}\right\}^2(\Delta x)^2 + \left\{\frac{\partial f_{xy}(x_i,y_j)}{\partial y}\right\}^2(\Delta y)^2\right]\Delta x\,\Delta y$$

According to (3.3) $f_{xy}(x_i,y_j)\Delta x\Delta y \approx \bar{k}_{ij}/N$, so we can compare (3.4) with (3.7). Summing h_{ij} over all cells and approximating this summation by an integral we find instead of (3.2)

$$3.8\quad E\{\hat{H}_x\} \approx H_x - \frac{I-1}{2N} + \int_{-\infty}^{\infty}\frac{1}{24 f_x(x)}\left\{\frac{\partial f_x(x)}{\partial x}\right\}^2(\Delta x)^2dx$$

$$3.9\quad E\{\hat{H}_{xy}\} \approx H_{xy} - \frac{IJ-1}{2N} + \int_{-\infty}^{\infty}\frac{1}{24 f_{xy}(x,y)}\cdot\left[\left\{\frac{\partial f_{xy}(x,y)}{\partial x}\right\}^2(\Delta x)^2 + \left\{\frac{\partial f_{xy}(x,y)}{\partial y}\right\}^2(\Delta y)^2\right]dxdy$$

For a binormal distribution this leads to

$$3.10\quad E\{\hat{H}_x\} \approx H_x - \frac{I-1}{2N} + \frac{1}{24}\Big(\frac{\Delta x}{\sigma_x}\Big)^2$$

$$3.11\quad E\{\hat{H}_{xy}\} \approx H_{xy} - \frac{IJ-1}{2N} + \frac{1}{24(1-\rho^2)}\Big[\Big(\frac{\Delta x}{\sigma_x}\Big)^2+\Big(\frac{\Delta y}{\sigma_y}\Big)^2\Big]$$

$$3.12\quad E\{\hat{I}_{xy}\} \approx I_{xy} + \frac{(I-1)(J-1)}{2N} - \frac{\rho^2}{24(1-\rho^2)}\Big[\Big(\frac{\Delta x}{\sigma_x}\Big)^2+\Big(\frac{\Delta y}{\sigma_y}\Big)^2\Big]$$

In 1955 Miller [8] derived the first order approximation of the bias caused by b). To our surprise, the approximation is independent of the distribution. Because of the slow convergence of expansion (3.1) the expression (3.2) looses its validity for $k_{ij}\to 0$; this case will occur if there are too many cells and some of them are almost empty. As expected the bias due to c) depends on the size of the cells and deteriorates as the cell sizes increase. The integral expressions of (3.8) and (3.9) are a measure of the smoothness of the probability density functions. If these are smooth, the bias reaches a minimum. If we compare (3.11) with (3.12) we expect for normal distributions a smaller bias in $\hat{I}_{xy}$ then in $\hat{H}_{xy}$.

4. VARIANCE

To derive an expression for the variance, we use the same method as for the derivation of the bias b). The variance of an entropy estimator is defined as:

$$4.1 \quad VAR\{\hat{H}_{xy}\} = E\{\hat{H}_{xy}^2\} - E\{\hat{H}_{xy}\}^2$$

Because we square $\hat{H}_{xy}$ we drop the terms with $(k_{ij}-\bar{k}_{ij})^2$ of approximation (3.1). Substituting (3.1) in (4.1) we find, using (2.1) and (2.2), for the variance

$$4.2 \quad VAR\{\hat{H}_x\} = \frac{1}{N}\sum_i \frac{\bar{k}_{i.}}{N}\log^2\frac{\bar{k}_{i.}}{N} - \frac{\tilde{H}_x^2}{N} + O\left(\frac{1}{N^2}\right)$$

$$4.3 \quad VAR\{\hat{H}_{xy}\} = \frac{1}{N}\sum_{ij}\frac{\bar{k}_{ij}}{N}\log^2\frac{\bar{k}_{ij}}{N} - \frac{\tilde{H}_{xy}^2}{N} + O\left(\frac{1}{N^2}\right)$$

$$4.4 \quad VAR\{\hat{I}_{xy}\} = \frac{1}{N}\sum_{ij}\frac{\bar{k}_{ij}}{N}\log^2\frac{\bar{k}_{ij}N}{\bar{k}_{.i}\bar{k}_{.j}} - \frac{\tilde{I}_{xy}^2}{N} + O\left(\frac{1}{N^2}\right)$$

If we replace $\bar{k}$ by the observed number of events k we obtain a variance estimator for an arbitrary distribution. For a known distribution we can approximate these expressions by replacing the summation by an integration; for a binormal distribution we find doing so:

$$4.5 \quad VAR\{\hat{H}_x\} \approx \frac{1}{2N}$$

$$4.6 \quad VAR\{\hat{H}_{xy}\} \approx \frac{1}{N}$$

$$4.7 \quad VAR\{\hat{I}_{xy}\} \approx \frac{\rho^2}{N}$$

Remarkebly the variance is approximately independent of the cell sizes; their astonishing simplicity might be an indication that these results can be derived in another way. We compare $VAR\{\hat{I}_{xy}\}$ (4.7) with $VAR\{\hat{\rho}\}$ of the Maximum Likelihood estimate of $\hat{\rho}$ for large N [9]

$$4.8 \quad VAR\{\hat{\rho}\} = \frac{(1-\rho^2)^2}{N}$$

Using the relation between I_{xy} and ρ for the binormal distribution

$$4.9 \quad I_{xy} = -\tfrac{1}{2}\log(1-\rho^2)$$

we can demonstrate that for large N the variances of both estimators $\hat{\rho}$ and $\hat{I}_{xy}$ are equivalent; in this respect there is no preference to determine I_{xy} via ρ or directly.
Determination of the optimal cell sizes is difficult because a priori knowledge of the distributions would be needed. Because we took $6\sigma_x = I\,\Delta x$ and $6\sigma_y = J\,\Delta y$, a ρ-dependent grid can be found so that for binormal distributions the biases b) en c) compensate eachother (3.12) This grid is then the optimal grid, because the variances are almost independent of the sizes of the cells.

5. SIMULATIONS

To verify our theory we generated 100 sequences of N=256 binormally distributed samples. We estimated I_{xy} using different cell sizes. The averaged results over the sequences are presented in figure 1. In the ideal case $\hat{I}_{xy}$ as function of I_{xy} is a straight line.
Characteristically I_{xy} is overestimated due to cause b); this overestimation increases with the number of cells. However, if I_{xy} is large then I_{xy} is underestimated, as in our simulations, due to domination of cause c); the underestimation decreases with an increasing number of cells, or in other words with an increasing resolution. After full bias correction we obtain the improved graphs of figure 2.
Because of the approximate character of the bias corrections, small deviations can be expected. If we only correct for the distribution independent bias b) we obtain the graphs of figure 3.
According to the graphs of figures 1-3 $E\{\hat{I}_{xy}\}$ is a monotonously increasing function of I_{xy}. This indicates the the maximum of $\hat{I}_{xy}$, as function of τ, is bias independent, if the grid is independent of τ. We expect therefore that a non-optimal grid will not change the delay estimate.
The estimated standard deviation obtained from 100 sequences with different N is given in table 1. The mean of the standard deviation calculated by using our variance estimator (4.4) is presented in table 2. The approximate standard deviation according to formula (4.7) is given in table 3. We see a good agreement between these tables; except for table 3 and $\rho=0$, which results can not be realistic.

6. DISCUSSION

We have shown that our bias and variance calculations are in good agreement with the simulation results. Further improvements can be achieved by more accurate calculations. We can e.g. take more terms of the Taylor expansion into account. For bias b) an exact expression can be derived by summation of all terms of the expansion (3.1). We can take also an exact sum instead of approaching it by an integral. We doubt whether such extensions to improve the estimate justify the effort of using more complicated calculations.
After comparing our discrete simulations with the continuous ones of Mars [1], we conclude that his AAMI estimates are effected by the same kind of biases. This indicates that his iterative method for finding an optimal kernel-width hardly improves the instimate. After further tests, also with real EEG-data, we

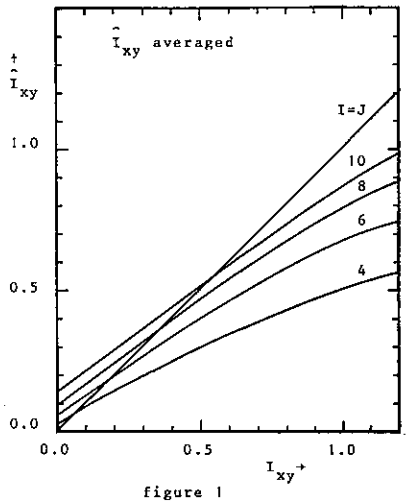


figure 1

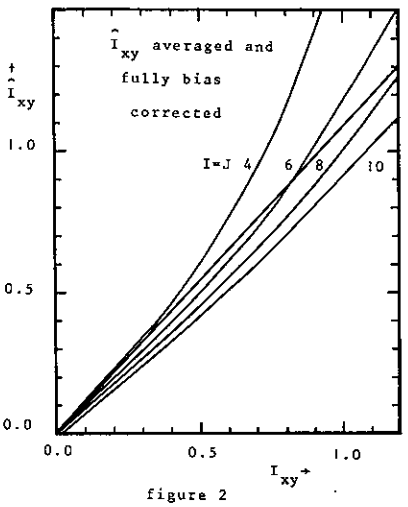


figure 2

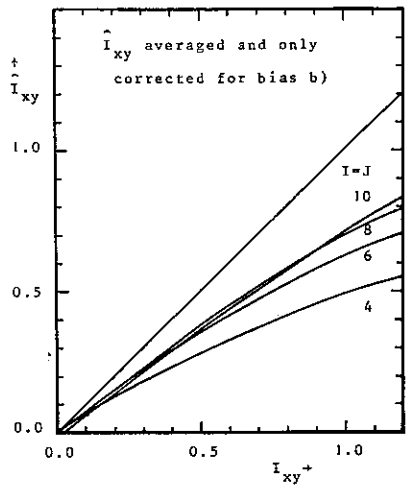


figure 3

concluded that both methods: Mars's and ours, lead to equivalent time-delay estimates. Our variance estimator enables us to judge the significance of a maximum in $\hat{I}_{xy}$. The derivation of a covariance estimator for $\hat{I}_{xy}$'s belonging tot different τ's is a problem, because this estimator depends on the covariance of subsequent data samples of our signal. Because of this dependancy a priori knowledge about the correlation function or the mutual information function is needed.
Our calculations should be tested using different probability densities, to get a better understanding of the validity of our estimators and corrections. To reduce the bias b) and the variance we consider to equalize the expected number of samples per cell by choosing a non-equidistant rectangular grid.
Of course our methods are applicable on entropy and mutual information estimation of discrete systems. In these systems only bias b) and the variance have to be considered.

ACKNOWLEDGEMENT

Its a pleasure to thank prof.ir. E.W. Gröneveld for the fruitful discussions which led to this work.

REFERENCES

[1] Mars, N.J.I. e.a., Signal processing 4 (1982) 139.
[2] Mars, N.J.I. e.a., Electroenceph. clin. Neurophysiol. 56 (1983) 194.
[3] Mars, N.J.I. e.a., Epilepsia 26 (1985) 85.
[4] Shannon, C.E., Bell Syst, Techn. Journ. 27 (1948) 379 and 27 (1948) 623.
[5] Parzen, E., The Ann. of Math. Stat. 33 (1962) 520.
[6] Epanechnikov, V.A., Theory of Prob. and its Appl. 14 (1969) 153.
[7] R. Moddemeijer, Internal report THTwente, 077.2519 (Dutch Language).
[8] Miller, G.A., Information theory in Psychology, Ed. by Quaster, H. (Glencoe, Illinois, 1955).
[9] Kendall, M.G. and Stuart, A., The advanced theory of statistics. Vol 2, par. 26 (Griffin, London, 1961).

TABLE 1: Standard deviation of $\hat{I}_{xy}$ obtained from 100 sequences.

	ρ	0.00	0.30	0.45	0.70	0.85	0.95
	I_{xy}	0.00	0.05	0.11	0.34	0.64	1.16
N	I=J						
128	6	0.020	0.037	0.035	0.054	0.061	0.075
	8	0.037	0.035	0.042	0.062	0.065	0.084
256	6	0.014	0.018	0.024	0.037	0.047	0.060
	8	0.019	0.021	0.027	0.042	0.049	0.063
512	6	0.007	0.013	0.019	0.027	0.031	0.041
	8	0.009	0.016	0.021	0.029	0.032	0.042

TABLE 2: Estimated standard deviation using (4.4) of $\hat{I}_{xy}$ averaged over 100 sequences

	ρ	0.00	0.30	0.45	0.70	0.85	0.95
N	I=J						
128	6	0.030	0.036	0.042	0.056	0.065	0.079
	8	0.039	0.043	0.048	0.060	0.068	0.077
256	6	0.016	0.023	0.027	0.039	0.046	0.056
	8	0.022	0.026	0.031	0.041	0.048	0.055
512	6	0.009	0.014	0.018	0.027	0.033	0.040
	8	0.012	0.016	0.020	0.029	0.034	0.038

TABEL 3: Approximate standard of $\hat{I}_{xy}$ according to (4.7)

	ρ	0.00	0.30	0.45	0.70	0.85	0.95
N							
128		0.000	0.027	0.035	0.062	0.075	0.084
256		0.000	0.019	0.028	0.044	0.053	0.059
512		0.000	0.013	0.020	0.031	0.038	0.042

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MEASUREMENT ACCURACY AND RESOLVING POWER OF HIGH RESOLUTION PASSIVE METHODS

D. THUBERT and L. KOPP

THOMSON-SINTRA ASM
B.P. 53
06801 CAGNES-SUR-MER, France

The so-called adaptative goniometer is a high resolution method using eigen-elements decomposition of the spectral density matrix. This array processing achieves the perfect deconvolution of the noise field when the model is completely known. In practice, noise is always present and corrupts this deconvolution. A set of formulas gives as a function of integration time, the measurement accuracy and the resolving power of adaptative goniometer. A same set of formulas can also be used to predict performances of conventional and adaptive beamformers. Results for all three localization methods are compared.

1- INTRODUCTION

Measurement accuracy and resolving power are two essential criteria to compare localization methods of an underwater passive listening system. This paper concentrates on three typical localization methods named conventional beamformer, adaptive beamformer and adaptive goniometer. The two first methods are now well known. The adaptive goniometer is based on eigenvectors decomposition of the spectral density matrix. It uses the property of orthogonality between the source subspace and the noise subspace [1]. Here are presented some useful tools to carry out performances computations. In the first section a classical formalism is recalled to compute bias and variance for the aforementioned localization methods. In the second section, their resolving powers are compared on a pair of close sources using the mid-point curvature of their responses. All these results are rigorously derived when the number of available independant datas is large enough.

The model is described as follows [6]. Let $\underline{S}(t)$ be the signal vector received on the K sensors of the array and $\underline{X}(f)$ the Fourier transform of $\underline{S}(t)$ at frequency f. The sources are assumed point like with perfect spatial coherence. The shape of the wavefront is a known function of the source position (range, bearing). For a zero mean gaussian process the cross spectral density matrix $\Gamma(f)$ is an exhaustive representation of $\underline{S}(t)$ at the considered frequency. For sake of brievety the term f is omitted when it is not necessary. An efficient method to compute the estimated cross spectral density matrix $\hat{\Gamma}$ is given by the periodogram :

$$\hat{\Gamma} = \frac{1}{N} \sum_{i=1,N} \underline{X}_i \, \underline{X}_i^+$$

N = number of snapshots

The vectors $\underline{X}_i$ are the Fourier transform of $\underline{S}(t)$ on N adjacent intervals of equal lengths. Under the previous assumptions

$$\Gamma(f) = \Gamma = E(\hat{\Gamma})$$

$$\Gamma = \sum_{i=1,p} \gamma_i \, \underline{d}(\underline{\theta}_i) \, \underline{d}(\underline{\theta}_i)^+ + \sigma J \qquad (1)$$

where - γ_i is the spectral density of the signal received from the i^{th} source on the array
- $\underline{d}(.)$ is the source position vector of dimension K ; K is the number of sensors. This vector is composed of the K transfer functions between a source and each sensor
- σ and J are respectivelly the spectral density and the cross spectral density matrix of the background noise. The shape of the matrix J is assumed a priori known. With a correct spatial whitening it is always possible to take J = I where I is the K x K identity matrix
- p is the number of sources
- $\underline{\theta}_i$ is a vector representing the position of the i^{th} source (range, bearing, ...). It is a (r x 1) vector
  $\underline{\theta}^+ = (\theta_1, \ldots, \theta_r)$
- E (.) = expectation value of (.)
- $(.)^+$ = conjugate transposed of (.)

2- MEASUREMENT ACCURACY : BIAS AND VARIANCE

A- Formalism

For narrow band signal processing, all the previous localization methods look for an extremum of a quadratic form $L(\underline{\theta}, \hat{A})$:

$$L(\underline{\theta}, \hat{A}) = \underline{d}(\underline{\theta})^+ \hat{A} \, \underline{d}(\underline{\theta}) \qquad (2)$$

$\underline{d}(\underline{\theta})$ is the steering vector which depends on the look direction $\underline{\theta}$ = (range, bearing ...)

$\hat{A}$ is a K x K matrix derived from the estimated cross density matrix $\hat{\Gamma}$. For conventional beamformer and adaptive beamformer $\hat{A}$ is respectively equal to $\hat{\Gamma}$ and $\hat{\Gamma}^{-1}$. For the adaptive goniometer $\hat{A}$ is the projector $\hat{P}_N$. The $\hat{P}_N$ matrix projects any vectors on the noise subspace N spanned by the K-p estimated eigenvectors associated with the K-p lowest eigenvalues of $\hat{\Gamma}$.

When the sources are detected, the extrema of $L(\underline{\theta}, \hat{A})$ give the estimated sources positions $\hat{\underline{\theta}}_i$. These vectors are given by :

$$\underline{g}(\hat{\underline{\theta}}_i, \hat{A}) = \mathrm{grad}_{\underline{\theta}} \left(L(\underline{\theta}, \hat{A}) \right)_{\underline{\theta}=\hat{\underline{\theta}}_i} = \underline{0} \qquad (3)$$

equation (2) leads to

$$\underline{g}\ (\underline{\hat{\theta}}_i,\ \hat{A}) = 2\ \mathrm{Re}\ \{U^+\ (\underline{\hat{\theta}}_i)\ \hat{A}\ \underline{d}(\underline{\hat{\theta}}_i)\} \qquad (4)$$

where $U(\underline{\hat{\theta}}_i) = (u_i\ (\hat{\theta}_i),\ \ldots\ u_r\ (\hat{\theta}_i))$

$U\ (\underline{\hat{\theta}}_i)$ is a (K x r) matrix

$\underline{u}_j(\underline{\theta}) = \partial\ (\underline{d}(\theta))/\partial(\theta_j)\ j \quad (1,r)$

Re {.} = real part of {.}

The vector $g\ (\underline{\hat{\theta}}_i,\ \hat{A})$ is a function of $\underline{\hat{\theta}}_i$ and $\hat{A}$. A first order expansion of $\underline{g}\ (\underline{\theta}_i,\ A)$ around the generalised point $(\underline{\theta}_i,\ A)$ gives :

$$\underline{g}\ (\underline{\hat{\theta}}_i,\ \hat{A}) = \underline{g}\ (\underline{\theta}_i,\ A) + h\ (\underline{\theta}_i,\ A)\ \Delta\underline{\hat{\theta}}_i + \underline{g}\ (\underline{\theta}_i,\ \hat{A}-A)$$

where . $\Delta\underline{\hat{\theta}}_i = \underline{\hat{\theta}}_i - \underline{\theta}_i$

. $h\ (\underline{\theta},\ A)$ is the (r x r) Hessian matrix of $L\ (\underline{\theta},\ A)$:

$$(h\ (\underline{\theta}_i,\ A))_{k,l} = (\partial^2\ l\ (\underline{\theta},\ A)/\partial\theta_k\ \partial\theta_l)_{\underline{\theta}=\underline{\theta}_i}$$

Approximation (5) is correct if the generalized points $(\underline{\hat{\theta}}_i,\ \hat{A})$ and $(\underline{\theta}_i,\ A)$ are closed. A good choice for $\underline{\theta}_i$ is given by

$$\underline{g}\ (\underline{\theta}_i,\ A) = 0 \qquad (6)$$

So that the relation (5) leads to

$$\Delta\underline{\hat{\theta}}_i = -2\ h^{-1}(\underline{\theta}_i,A)\ \mathrm{Re}\ \{U^+(\underline{\theta}_i)\ \hat{A}\ \underline{d}(\underline{\theta}_i)\} \qquad (7)$$

Bias and variance of $\Delta\underline{\hat{\theta}}_i$ are obtained by calculating its first and second moments. Let us define $\underline{b}\ (\underline{\theta}_i)$ and $\mathrm{cov}\ (\underline{\theta}_i,\ \underline{\theta}_j)$ respectively bias on the i^{th} position source and covariance matrix between the geometrical parameters of the i^{th} and j^{th} source

$$\cdot\ \underline{b}\ (\underline{\theta}_i) = E\ (\Delta\underline{\hat{\theta}}_i) \qquad (8)$$

$$\underline{b}\ (\underline{\theta}_i) = -2\ h^{-1}(\underline{\theta}_i,A)\ \mathrm{Re}\ \{U^+(\theta_i)\ \tilde{A}\ \underline{d}(\underline{\theta}_i)\}$$

$$\tilde{A} = E\ (A)$$

$$\cdot\ \mathrm{cov}\ (\underline{\theta}_i,\ \underline{\theta}_j) = E\ (\Delta\underline{\hat{\theta}}_i\ \Delta\underline{\hat{\theta}}_j{}^+) - \underline{b}(\underline{\theta}_i)\underline{b}(\underline{\theta}_j)^+$$

If the (r x r) matrix $R\ (\underline{\theta}_i,\ \theta_j)$ is defined by

$$R(\underline{\theta}_i,\ \underline{\theta}_j) = h(\underline{\theta}_i,\ A)\ \mathrm{cov}(\underline{\theta}_i,\ \underline{\theta}_j)\ h(\underline{\theta}_j,A) \qquad (9)$$

relation (7) leads to

$$R\ (\underline{\theta}_i,\underline{\theta}_j)_{k,l} = 2\ \mathrm{Re}\ \{\sigma\ (u_k\ (\underline{\theta}_i),\ \underline{d}(\underline{\theta}_i),\ u_l\ (\underline{\theta}_j),\ \underline{d}(\underline{\theta}_j)) + \sigma\ (u_k\ (\underline{\theta}_i),\ \underline{d}(\underline{\theta}_i),\ \underline{d}(\underline{\theta}_j),\ \underline{u}_l\ (\underline{\theta}_j))\} \qquad (10)$$

with

$$\sigma\ (\underline{a},\underline{b},\underline{c},\underline{d}) = E\ (\underline{a}^+\ \hat{A}\ \underline{b}\ \underline{c}^+\ \hat{A}\ \underline{d}) - E\ (\underline{a}^+\ \hat{A}\ \underline{b})\ E\ (\underline{c}^+\ \hat{A}\ \underline{d}) \qquad (11)$$

$\underline{a},\underline{b},\underline{c},\underline{d}$ are any deterministic (k,1) vectors. The bias vector $\underline{b}\ (\underline{\theta}_i)$ is a function of $\hat{A}$, the covariance matrix $\mathrm{cov}\ (\underline{\theta}_i,\ \theta_j)$ depends on $\sigma\ (\underline{a},\underline{b},\underline{c},\underline{d})$.

Let us assume that the Fourier transform of the signal vector received on the sensors are some complexe zero mean gaussian process.

The quantities $\tilde{A}$ and σ may then be derived with a first order approximation.

The computations use classical algebra but are long and tedious and we only presente the final results.

a) $\hat{\tilde{A}} = \hat{\Gamma}$

$$\tilde{A} = \Gamma \qquad (12)$$

$$\sigma\ (\underline{a},\ \underline{b},\ \underline{c},\ \underline{d}) = \frac{1}{N}\ \underline{a}^+\ \Gamma\ \underline{d}\ \underline{c}^+\ \Gamma\ \underline{b} \qquad (13)$$

In this case no approximation has been made. (13) was obtained using the well known theorem on the fourth order moment of a complex zero mean gaussian process.

b) $\hat{A} = \hat{\Gamma}^{-1}$

$$\tilde{A} = \frac{N}{N-K}\ \Gamma^{-1} \qquad \tilde{A} \simeq \Gamma^{-1} \text{ for } N \gg K \qquad (14)$$

$$\sigma\ (\underline{a},\ \underline{b},\ \underline{c},\ \underline{d}) = \frac{1}{N}\ \underline{a}^+\ \Gamma^{-1}\ \underline{d}\ \underline{c}^+\ \Gamma^{-1}\ \underline{d} \qquad (15)$$

Relation (14) is given by R.J. MUIRHEAD [3] pp. 97. Matrix $\hat{\Gamma}^{-1}$ has been approximated by $\hat{\Gamma}^{-1} = \Gamma^{-1} - \Gamma^{-1}\ \hat{\Gamma}\ \Gamma^{-1}$ to derive (15).

c) $\hat{A} = \hat{P}_N$

In this case the computation is more complicated, use is made of the perturbation theory to get an approximate solution. Moreover, the eigenvectors associated with an isolated eigenvalue are asymtotically normal [3]. Thus, for large N, classical theorems on gaussian process hold for all the vectors belonging to the source subspace. With a first order approximation of these eigenvectors given by perturbation theory, one finds :

$$\tilde{A} = (1-\beta)\ P_N + \frac{K-p}{N}\ Q_S \qquad (16)$$

$$\sigma\ (\underline{a},\underline{b},\underline{c},\underline{d}) = \frac{1}{N}\ \underline{a}^+\ P_N\ \underline{d}\ \underline{c}^+\ Q_S\ \underline{b} + \frac{1}{N}\ \underline{a}^+\ Q_S\ \underline{d}\ \underline{c}^+\ P_N\ \underline{b} \qquad (17)$$

N = noise subspace

S = source subspace

$P_N = \sum_{k \in N} \underline{V}_k\ \underline{V}_k{}^+$; k [1, k-p] ; $\Gamma\ \underline{V}_k = \sigma\ \underline{V}_k$

$Q_S = \sum_{i \in S} \eta_i\ \underline{V}_i\ \underline{V}_i{}^+$; i [k-p+1,K] $\Gamma\ \underline{V}_i = \lambda_i\ \underline{V}_i$

$$\beta = \sum_{i \in S} \eta_i \qquad \eta_i = \frac{\lambda_i\ \sigma}{(\lambda_i - \sigma)^2}$$

Using these equations, it is now easy to derive the bias and the covariance matrix of the unknown geometrical parameters estimates. In most practical cases it is difficult to find simple analytical expressions of these quantities but on a computer the previous expressions are easy to program.

B- Example

Computations of bearing bias and variance are easy when the noise field is composed of a single spatially coherent source in an incoherent noise background (independant between the sensors of the array). For omnidirectional sensors relations (8) and (9) leads for the three localization methods to:

$$b(\theta) = 0$$

$$\mathrm{cov}(\theta, \theta) = \frac{1}{2N} \frac{\sigma(K\gamma+\sigma)}{\gamma^2} F^{-1}(\theta)$$

where $F(\theta) = K \|\underline{u}(\theta)\|^2 - \|\underline{d}^+(\theta)\,\underline{u}(\theta)\|^2$

$$\underline{u}(\theta) = d(\underline{d}(\theta)/d(\theta) \quad \|.\| = \text{modulus}$$

In this case cov (θ, θ) is exactly the Cramer-Rao lower bound of the estimate bearing θ when signal of a single source and an incoherent noise background are received on omnidirectional sensors [8].

At the end of this paper are presented some results obtained on a computer with a noise field composed of two point-sources of the same power in an incoherent noise background. The antenna is a linear array of nine omnidirectional sensors at half wavelength. Pictures 1 and 2 show the bearing bias of one source plotted against the angular distance between the two sources for the three localization methods. The bias induced by the conventional beamformer is independant of the SNR. For an input SNR of 0 dB. and an angular distance $\Delta\theta = 14°$, the bias in bearing of the adaptive goniometer is ten times lower that the one of the adaptive beamformer and one hundred times lower than the one of the conventional beamformer. Pictures 3, 4 and 5 show the standart deviation of the angular distance $\Delta\theta$ plotted versus $\Delta\theta$ (unit is the 3 dB. half beamwidth of the conventional beamformer). Picture 3 shows an interesting behaviour of the standart deviation for the conventional beamformer. This function does not always decrease when $\Delta\theta$ increases as it does for the adaptive beamformer and adaptive goniometer. Most of the time the adaptive goniometer gives a standart deviation lower than those of the conventional and adaptive beamformers, as expected.
Monte-Carlo simulations have been carried out for the same cases, their results are plotted on the same pictures. One hundred snaphots were taken : this seems enough to use this asymptotic approach and somewhat justifies our approximations.

3- ANGULAR RESOLVING POWER

Angular resolving power is naturally an essential criterion of performance for any passive localization method. (It is known that the adaptive goniometer achieves the perfect deconvolution of the noise field for an infinite observation time).

For finite data lengths, one can caracterize the resolving power by a geometrical criterion based on the sign of the second derivative of the quadratic form $L(\underline{\theta}, \hat{A})$ (2) (The vector $\underline{\theta}$ is here an unique scalar θ : bearing).
With that criterion, two sources of same power are just separated when the second derivative $h(\theta, A)$ is equal to zero at the mid-point bearing θ_0. With relation (2) :

$$h(\theta, \hat{A}) = 2\,\mathrm{Re}\{\underline{u}^+(\theta)\,\hat{A}\,\underline{u}^+ + S^+(\theta)\,\hat{A}\,\underline{d}(\theta)\}$$

where : $u(\theta) = \partial(\underline{d}(\theta))/\partial(\theta)$

$$S(\theta) = \partial(\underline{u}(\theta))/\partial(\theta)$$

Actually $\hat{A}$ depends on the angular separation of the sources. A good approximation of the angular resolving power is then given by relation (20)

$$h(\theta_0, E(\hat{A})) = 0 \qquad (20)$$

$E(\hat{A})$ is available in relations (12), (14) and (16). In the case under study, the two sources have the same power and the cross spectral density matrix is :

$$\Gamma = \gamma(\underline{d}(\theta_1)\,\underline{d}(\theta_1)^+ + \underline{d}(\theta_2)\,\underline{d}(\theta_2)^+) + \sigma I$$

On a computer, equation (20) is very simple to solve for the three methods. Analytical expressions of the angular resolving powers of the three methods have been derived by approximating (20), for a linear array of k equispace and omnidirectional sensors. A fourth order expansion of the antenna directivity function leads to CRP, ARP and GRP respectively conventional, adaptive and goniometer resolving powers.

$$CRP = \frac{\lambda}{\pi d}\sqrt{\frac{10}{2K^2-3}} \qquad (21)$$

$$ARP = 1.2\,CRP\left(\frac{K\gamma}{\sigma}\right)^{-1/4} \qquad K \gg 1 \qquad (22)$$

$$GRP = 1.6\,CRP\left(\frac{N\gamma}{\sigma}\right)^{-1/4} \qquad N,K \gg 1 \qquad (23)$$

λ = wave length
d = distance between two sensors
K = number of sensors
N = number of snapshots

As expected CRP depends only on the geometrical parameters of the array. ARP is limited by the signal to noise ratio and does not depend on the observation time. GRP is also limited by the signal to noise ratio but it improves when the observation time increases. Defining the gain over conventional beamformer by the ratios CRP/ARP and CRP/GRP, it clearly appears that this gain increases as the fourth root of the signal to noise ratio. As shown on relation (23) GRP improves slowly with the number of snapshots N ; for example when N increases from 10^2 to 10^4 the gain is improved by a factor 3. Some simulations have been carried out to verify these approximations (figure 6).

CONCLUSION

In this study some very useful theorems have been recalled to compute measurement accuracy of the conventional beamformer, adaptive beamformer and adaptive goniometer. In the general case, asymptotic expressions to evaluate bias and variance of the sources positions are rather complicated but remain tractable with a computer. When the signals received on omnidirectional sensors come from a single source in a incoherent noise background the three methods are unbiased and reach the Cramer-Rao lower bounds of the unknown geometrical parameters. When two sources are present in the noise field analytical compulations are unpratical. Numerical implementation of the asymptotic equations have shown the geometrical bias to decrease with the input SNR for adaptive beamformer and adaptive goniometer. This is not true for conventional beamformer. Of course these expressions for the bias can also be used to compute geometrical bias induced by some modeling errors (like bad calibration of the sensors). These error induced biases will often been stronger than those we have studied here. The expressions derived for the variance have been valided by Monte-Carlo simulations.
Analytical approximations of the resolving power have been derived for the three localization methods using the same criterion of sources separation.
Resolving power of the conventional beamformer depends only on geometry. Resolving powers of the adaptive beamformer and the adaptive goniometer improve as the fourth root of the input SNR. Resolving power of the adaptive goniometer further improves as the fourth root of integration time.

REFERENCES

[1] G. BIENVENU, L. KOPP, "Source power estimation method with high resolution bearing estimater" Proc. ICASSP 81, April 1st, 1981. pp. 153-156

[2] W.S. LIGETT, "Passive sonar : fitting models to multiple times series" Nato Asi on signal processing, Loughborpugh (UK) 1972

[3] Robb J. MUIRHEAD "Aspects of Multivariate statistical theory" pp. 417

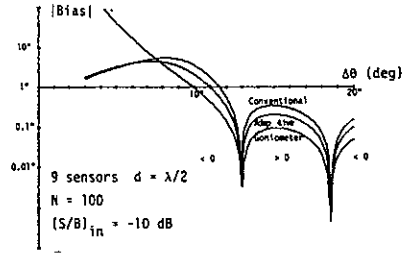
[4] J. MUNIER "Pouvoir séparateur en estimation non linéaire en présence de bruit faible GRETSI 1977 26/30 avril

[5] H.L. VAN TREES "Detection, estimation and modulation theory" Vol. 1 J. WILEY 1968

[6] G. BIENVENU, L. KOPP "Optimality of high resolution array processing using the eigensystem approach" IEEE ASSP 31, 5, pp. 1235-1248 October 1983

[7] K.C. SHARMAN and T.S. DURRANI "Resolving power of signal subspace methods for finite data lengths" IEEE ASSP, pp. 1501-1504 August 1985

[8] G. CLIFFORD CARTER "Variance bounds for passively locating on acoustic source with a symetrical line array" JASA Vol. 62 n° 4 pp. 922 October 1977



Bias Fig. 1 and 2

Figure 1

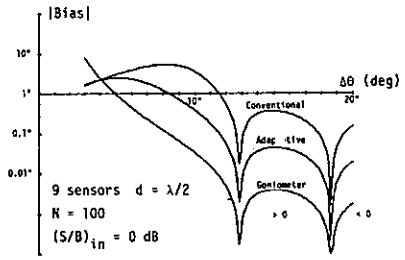


Figure 2

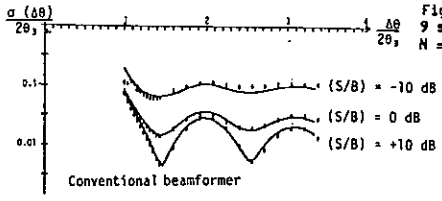


Fig. 3, 4, 5
9 sensors
N = 100 d = λ/2

Figure 3

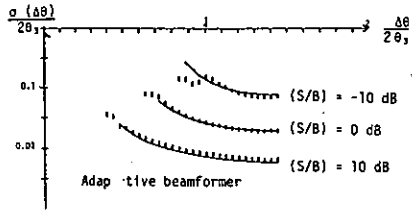


Figure 4

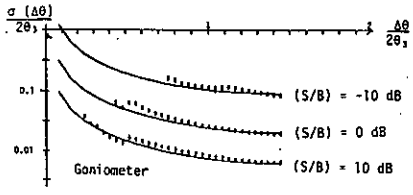


Figure 5

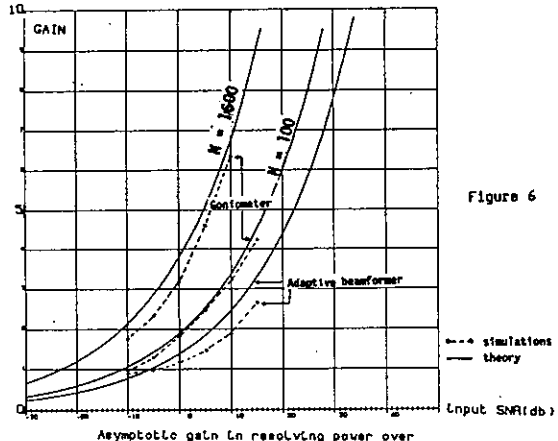


Figure 6

Asymptotic gain in resolving power over Conventional beamformer for a linear array of nine equispace sensors

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N4.5

A NOVEL OPTIMUM ENERGY SOLUTION IN ITERATIVE CONSTRAINED RESTORATION

R. FOKA S.Y. KUNG(*)
THOMSON-SINTRA Activités Sous-Marines, 1, avenue Aristide Briand
94117 ARCUEIL Cedex/France

In this paper, the Toeplitz Approximation Method (TAM) of stochastic system identification is applied to the linear equal spaced array narrowband source direction finding problem. The proposed algorithm provides high resolution direction finding capability and is designed for an arbitrary noise, low signal noise ratio (SNR) multipath signal environment. As such, it extends existing capability in fields such as passive sonar, radar and communications.

1. INTRODUCTION

Eigenstructure based methods for direction finding have recently been developed. These methods utilize the decomposition of the array data correlation matrix, employing either the signal subspace or the noise subspace basis obtained as eigenvectors of this matrix. This approach is used in the "MUSIC" (Multiple Signal Classification) algorithm of Schmidt (1) and in the described methods by Bienvenu and Kopp (2), Owsley (3). These methods can be viewed as a generalization of an algorithm which uses only the eigenvector associated with the smallest eigenvalue. These approaches have been examined in the case where the additive noise is either spatiallly white or of known covariance.
A novel direction finding algorithm, based on a reduced order Toeplitz approximation of an estimated spatial covariance matrix, is proposed in this paper. The estimated covariance matrix, in the case in which sources are uncorrelated and statistically stationary, is Toeplitz. In a multipath environment, however, the source paths are fully correlated, and this covariance matrix is not Toeplitz. The Toeplitz structure can be guaranteed by employing spatial smoothing, which destroys cross correlation between directional components. In the TAM approach, the spatial data may be modeled as the output of a self generating ARMA process with poles, corresponding to arrival directions, on the unit circle. A state space representation is estimated from a covariance matrix low order approximate. The algorithm used to obtain this low order matrix, which is based on the Singular Value Decomposition (SVD) of the spatial data matrix, has low sensitivity to data perturbation. It is our claim that, because of this reduced sensitivity to data perturbation, the TAM method is designed for robustness in an arbitrary ambient noise environment. Beyond this statement, the claim is verified via empirical examination. Further analytical justification is still an open issue. In this paper, after a brief mathematical formulation of the problem, the Toeplitz Approximation Method (TAM) is described and simulation results are presented.

2. MATHEMATICAL FORMULATION

Consider a linear receiving array with L equally spaced elements and m direct sources with a total of p paths. At the array, source wavefront propagation is assumed planar and source energy is incident on the array a distinct angles $(\theta_k, k=1,\ldots,p)$. The propagation channel is homogeneous and the signals are narrowband. The received signal at the ith sensor $y_i(t)$ is given by

$$x_i(t) = \sum_{k=1}^{p} a_k(t)\exp\left[-j\left((i-1)2\pi(d/\lambda_o)\sin\theta_k+\phi_k\right)\right] \qquad (1)$$

$$y_i(t) = x_i(t) + n_i(t)$$

where d is the sensor spacing, λ_o is the processing center wave length, ϕ_k is the random phase delay, and $n_i(t)$ the additive noise at the i^{th} sensor. The k^{th} source is characterized by θ_k the arrival angle measured relative to array, broadside, and $a_k(t)$, the narrowband source envelope.
It is assumed that the spacing between sensors is less than half of the minimum signal wave length (i.e. to avoid the spatial aliasing), and sources are uncorrelated with additive noise. For algorithm development is also assumed that the noise processes are zero mean Gaussian, uncorrelated from element to element, independent of the signal, and have variance σ^2. Rewriting equation (1) in matrix notation, we obtain

$$Y(t) = D\,S(t) + N(t) = X(t) + N(t) \qquad (2)$$

(*) Signal and Image Processing Institute University of Southern California
Los Angeles, California 90089, U.S.A.

where $Y(t) = \left(y_1(t),\ldots,y_L(t)\right)^t$

$S(t) = \left(s_1(t), s_2(t),\ldots,s_p(t)\right)$

$s_k(t) = a_k(t)e^{j\omega_o t}$, $D = \left(D_{\theta_1}, D_{\theta_2},\ldots D_{\theta_p}\right)$

$D_{\theta_k} = \left(1, e^{-j\tau_k}, e^{-j2\tau_k},\ldots,e^{-j(L-1)\tau_k}\right)^t$

$\tau_k = 2\pi(d/\lambda_o)\sin\theta_k$, $N(t) = \left(n_1(t),\ldots n_L(t)\right)^t$

Therefore, the spatial covariance matrix is :

$R = E(Y(t)Y') = R_x + R_N$

$$R = DR_SD' + \sigma^2 I \qquad (3)$$

where "'" denotes the complex conjugate transpose; "t" denotes the simple transpose; "E" stands for expectation, an ensemble average operator ; I is L x L identity matrix; $R_S = E\left(S(t)S'(t)\right)$ is the p x p path signal covariance matrix; and D is the L x p direction matrix, or a Vandermonde matrix, whose columns are the steering vectors of the impinging planar wavefronts. R is non-negative definite (i.e. eigenvalues of R are non-negative). In direction finding problems, eigenstructure methods of estimation the directions-of-arrival (θ_k) are based on exploiting this structure of R. That is, R_x has a range basis consisting of the direction vectors of source plane waves, and R is the sum of a p-rank matrix DR_SD' and $\sigma^2 I$.

3. TOEPLITZ APPROXIMATION FORMULATION

Direction finding is more a process of spectral estimation and detection than a process of filtering (beamforming). This stems from the fact that for direction finding, like spectral estimation, power is investigated in Fourier transform domain. Since the second-order information is vital for characterizing stationary processes, the Toeplitz matrix has naturally an indispensable place in stationary direction finding.

Assuming that, at any instant in time, received sources appear across the linear array as complex sinusoids, the array snapshot vector can be considered as the output of a very special ARMA model. The success of many sophisticated model reduction and approximation methods, which use the state spece approach for the deterministic case, suggests that we look for a state space parameterization of the ARMA model instead of transfer function parameterization. If we take this viewpoint, the problem of direction finding in the presence of ambient noise becomes intimately related to the stochastic model reduction problem, and can be treated as such.

State Space Formulation

Since the sources received by the array are assumed to appear as complex sinusoids the ARMA model is one with poles on unit circle and zero input noise power. Poles on the unit circle make the system self generating. Because the spatial covariance of X(t) (defined as the noise free measurement) has rank p, $x_k(t)$ is a pth order Markov process with respect to spatial index k, and $z_k(t)$ can be predicted form $\left(x_{k-1}(t), x_{k-2}(t),\ldots,x_{k-p}(t)\right)$. As indicated in (4) we can formulate a special state space representation as

$$Z_{k+1} = FZ_k,\ x_t(t) = hZ_k \qquad (4)$$

One choice of the p-dimensional state vector Z_k leads to

$Z_1 = \left(s_1(t),\ldots,s_p(t)\right)^t$.

$F = \mathrm{diag}(e^{-j\tau_1}, e^{-j\tau_2},\ldots,e^{-j\tau_p})$

$h = (1,1,1,\ldots,1)$.

(Note that the temporal index is dropped in the state representation. This is because, in this formulation, we are interested in modeling spatial caracteristics only). Any other choice is related to the representation expressed above as a similarity (coordinate) transformation of F. The state transformation may not be unique but the transfer function is (i.e. poles and zeros do not change). Therefore, after any coordinate transformation, the eigenvalues of F will always be $\exp(-j\tau_i)$, i=1,...,p.

We can also write $X = \left(x_1, x_2,\ldots x_L\right)^t$ as

$$X = \left(h, hF, hF^2,\ldots,hF^{L-1}\right)^T Z_1 = \Theta Z_1 \qquad (5)$$

Therefore, $R_x = \Theta E\left(Z_1Z_1'\right)\Theta'$.

In system theory, Θ is called the observability matrix and if all τ_i's are distinct, i.e. if the p paths do not overlap, the p columns of Θ are independent. If all paths are uncorrelated, $E\left(Z_1Z_1'\right)$ is diagonal and strictly positive definite, and R_x has rank p. However, when the path signals are coherent, some of the ϕ_k variables are dependent, $E\left(Z_1Z_1'\right)$ will be singular and R_x will have rank < p. In fact, R_x will not be Toeplitz and the spatial process X will no longer be spatially stationary. Thus, the coherence of path signals destroys both the Toeplitz property and the p-rank property of R_x. Clearly, if R_x is non-Toeplitz, the spatial correlation matrix R, pertaining to the observed signal vector, is also non-Toeplitz. If we use time-averages to estimate R_x then, asymptotically, the estimate

$$\lim_{T\to\infty}\left(\frac{1}{T}\right)\sum_{t=1}^{T} X(t)X'(t) \qquad (6)$$

will be Toeplitz with rank p only if the path

signals are uncorrelated. If some of the paths are coherent, the estimate will have rank less than p and will not be Toeplitz.

Spatial Averaging

If we use spatial averages instead of time averages, then asymptotically we will obtain a Toeplitz matrix C that does have rank p. Let

$$c(m) = \lim_{L \to \infty} \left(\frac{1}{L}\right) \sum_{i=1}^{L} x_i x'_{i+m} \qquad (7)$$

Since $x_{i+m} = hF^m Z_i$ and $x_i = hZ_i$

$$c(m) = hF^m Ph' \qquad (8)$$

where $P = \lim_{L \to \infty} \left(\frac{1}{L}\right) \sum_{i=1}^{L} Z_i Z'_i$

is a p x p state "covariance" matrix. Note that P is not an estimate of $E(Z_i Z'_i)$ is a function of i and spatial averaging destroys this dependence. As a matter of fact, $E(Z_i Z'_i)$ is singular if some ϕ_i's are dependent (path signals are coherent), but P is always full rank. Also note that

$$FPF' = \lim_{L \to \infty} \left(\frac{1}{L}\right) \sum_{i=1}^{L} (FZ_i)(FZ_i)'$$

$$= \lim_{L \to \infty} \left(\frac{1}{L}\right) \sum_{i=1}^{L} Z_i Z'_i = P \qquad (9)$$

Now, we form the new Toeplitz covariance matrix as following :

$$C = \begin{bmatrix} c(0) & c(1) & c(2) & \cdots \\ c(1) & c(0) & c(1) & \cdots \\ c(2) & c(1) & c(0) & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Using $c(m) = hF^m Ph'$, we can easily verify the following factorization :

$$C = \begin{bmatrix} h \\ hF \\ hF^2 \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} \cdot \left(Ph', F^{-1}Ph', \ldots\right) = \Theta P \Theta'. \qquad (10)$$

Since Θ has only p columns, the rank of C must be $< p$. As stated earlier, Θ and P are always full rank, thus C has rank p irrespective of the coherence of paths.

Estimation

Numerous covariance matrix estimators exist in the literature. One very popular choice is the unbiased estimator. When the array length is finite, an unbiased estimate of c(m) is

$$\left(\frac{1}{(L-m)}\right) \sum_{i=o}^{L-m-1} x_i \, x_{i+m'} \qquad m = 0, \ldots, L-1 \qquad (11)$$

If the time series is composed of $\frac{p}{2}$ sinusoids, the covariance matrix C should ideally be a Toeplitz matrix of rank p. However, because (L) is often relatively small for practical arrays, this estimate may not be good and the ideal matrix characteristics may not be realized. The estimate may be improved by using a pseudo-ensemble average; a combined spatial and temporal average. It is therefore suggested that the estimator,

$$\hat{c}(m) = \left(\frac{1}{T(L-m)}\right) \sum_{t=1}^{T} \sum_{k=o}^{L-m-1} x_k(t) \; x_{k+m}(t)$$

be used where t is the temporal index and T is the number of snapshots.

Toeplitz Approximation Method (TAM)

The objective of the Toeplitz approximation method is to retrieve a p rank estimate of the matrix C_x from the principal components (via SVD) of $\hat{C}$, and to then enforce the structure of D to obtain estimates of F and h from the principal components. In summary, the two steps of the TAM approach (4,5) are :

Step (1) :

Perform an SVD on $\hat{C}$ and arrange the singular values $(\sigma_k, k=1,\ldots L)$ in decreasing order. The SVD is

$$\hat{C} = U\Sigma V' \qquad (12)$$

$$= \left(U_1 U_2\right) \begin{bmatrix} \Sigma_1 & o \\ o & \Sigma_2 \end{bmatrix} \begin{bmatrix} V'_1 \\ V'_2 \end{bmatrix}$$

where the p x p diagonal matrix Σ_1 contains the largest singular values. A p rank approximant to $\hat{C}$ is $U_1\Sigma_1 V'_1$, and the (minimal) approximation error in the spectral norm is

$$\left|\left|\hat{C} - U_1\Sigma_1 V'_1\right|\right|_E = \sigma_{p+1}.$$

Step (2) :

Though the component selection step provides estimates of the product of ΘC, the actual choice of the factors is not unique. For numerical reason, the choice of coordinate can be crucial. For numerical stability, a good choice is the balanced coordinate system. A realization in balanced coordinates is obtained by choosing the observability matrix as

$$\Theta = U_1 \Sigma_1^{1/2} \qquad (13)$$

Ideally, the observability matrix Θ should have the structure $\left(h \; hF \; hF^2 \ldots\right)^t$ and satisfy $\Theta F = \left(hF \; hF^2 \ldots\right)^t = \Theta^{\uparrow}$. But our estimate $\Theta = U_1 \Sigma_1^{1/2}$ will not. However, one can resort to a least squares solution for F as follows :

$$F = \Theta^{+}\Theta^{\uparrow} \text{ (state transition matrix)} \qquad (14)$$

$$\Theta^{+} = (\Theta'\Theta)^{-1}\Theta' \qquad (15)$$

where $\uparrow$ means shifted one row up with an added last row of zeros.

The minimal error $||\Theta F - \Theta^{\uparrow}||_F$ (Frobenius norm) can be shown to be $O(\sigma_{p+1})$ (4,5). The eigenvalues of $F=\mathrm{diag}\left(\exp(-j\tau_1),\exp(-j\tau_2),\ldots,\exp(-\tau_p)\right)$ give the directions of the paths, since the eigenvalues are

$$\exp(-j\tau_i) = \exp\left(-j2\pi(\frac{d}{\lambda_o})\sin\theta_i\right)$$

Also, because $\tilde{F} = QFQ^{-1}$, the envelope modulus information can be estimated as

$$|a_i| = ||h^i Q||^{1/2},$$

where h is the first row of observability matrix Θ. It is known that the obtained estimate of F is always stable.

Spatial Spectrum

In general, the source energy, distributed as a function of angular direction, can be described as

$$P_{TAM}(\tau) = \sum_{i=1}^{p} \left| \frac{|a_i|}{1-r_i\exp(-j(\tau-\tau_i))} \right|^2 \qquad (16)$$

where $|a_i|$ is the estimated amplitude, r_i is the radius of ith pole $(z_1,z_2,\ldots z_p)$ location, τ_i corresponds to the ith direction angle of source parths, and τ corresponds to the steering angle of the array. Therefore, the bearing of the source paths (θ) are determined by finding the spatial frequenties at which $P_{TAM}(\tau)$ achieves a maxima.

4. SIMULATIONS

In order to perform TAM applied to the direction finding, the temporal snapshot data and Gaussian noise data were generated artificially by a computer. The snapshot model consists of several sinusoids plane waves with different phase impinging upon a linear array of 32 equispaced with inter-element spacing of one half wavelength. Each sensor is contaminated by an additive white noise source with zero-mean at a SNR of 0 dB. The source signals were generated as direct paths and two reflected paths which are equal in magnitudes. The source frequencey, normalized by the sample frequency, is .25 Hz. The sample covariance matrix was formed using 128 snapshots of the data.

The figure shows Monte Carlo simulation results for TAM. This figure shows resolution and direction-finding results for four equal-powered emitters at angular locations 15°,20°,25° and 70°.

5. CONCLUSIONS

TAM algorithm for direction finding problems, with application in multipath environment with linear equi-spaced arrays, have been presented. Computational complexity analysis shows that it will be possible to implement these algorithms in real time on high speed parallel processing computer systems within the next five years. Simulations of the algorithms based on a small number of input snapshots (64-128) yield satisfactory results. Preliminary results indicate that one can be optimistic in expecting high resolution and good accuracy when a relatively small number (10-32) of array elements is used.

The authors are grateful to J.Y. GUEDON of THOMSON-SINTRA ASM for his very valuable contributions and discussions.

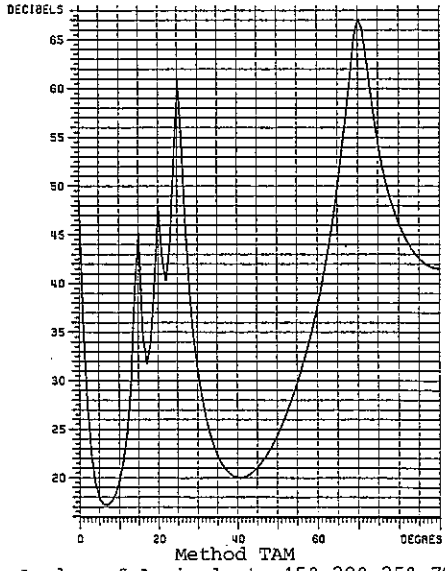
References

(1) R.O.Schmidt, "Multiple Emitter Location and Signal Parameter Estimation", Proc. RDAC Spectral Estimation Workshop, pp.243-258, 1979.

(2) G. Bienvenu and L. Kopp, "Source Power Estimation Method Associated with High Resolution Bearing Estimation", Proc. IEEE ICASSP 1983, Atlanta, GA, pp.153-156, 1981.

(3) N.L. Owsley, "Spectral Signal Set Extraction", Aspects of Signal Processing, Part II, G. Tacconi (ed.) D. Reidel Publishing Company, Dordrecht-Holland, 1977.

(4) S.Y. Kung, K.S. Arun, and D.V. Bhasker Rao, "Tate Space and Singular Value Decomposition Based Approximation Methods for the Harmonic Retrieval Problem", Optical Society of America, Vol 73, pp.1799-1811, 1983.



Method TAM
Angles of Arrival at : 15°,20°,25°,70°
SNR = 0 dB

Tutorial on ADAPTIVE DETECTION

B. Picinbono
Université de Paris-Sud
Ecole supérieur d'Electrotechnique
Gif sur Yvette
France

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N4.7

DETECTION METHODS USING EIGENVALUES : THEORETICAL PERFORMANCES AND PRACTICAL LIMITS IN UNDERWATER PASSIVE LISTENING

J.M. PASSERIEUX and L. KOPP

THOMSON-SINTRA ASM
B.P. 53
06801 CAGNES-SUR-MER CEDEX (FRANCE)

In underwater passive listening the number of sources may be determined from the latent roots of the measured cross-spectral density matrix of signals received by an array. Expressions are derived from multivariate statistical theory to predict performance of such methods and applied to two cases :

- idealized situations (one or two perfectly coherent sources in an incoherent noise background),
- more realistics ones taking into account coherence mismatches.

INTRODUCTION

During the last few years new array processing methods have been proposed for underwater passive listening : the so-called "high resolution" methods which are based on the properties of the eigensystem of the cross-spectral density matrix of signals received by an array ([1] to [8]). Here we just deal with the first stage of these methods : detection, i.e. determination of the number of sources.

The key of improved performances of these methods compared to previous ones (as conventional or adaptative beamforming) is that they exploit the contrast between :
- a spatially incoherent background noise (statistically independent on sensors)
- one or several perfectly coherent sources.

Using these hypotheses one can show that the eigenvectors of the exact spectral matrix may be split in two orthogonal subspaces : the source subspace, which is spanned by the source position vectors, and the noise subspace ([1], [2], [3]). An important point is that the noise subspace is the eigenspace related to the minimum eigenvalue of the spectral matrix which has multiplicity K - p (K = number of sensors, p number of sources).

Actually the exact density matrix is unknown and one just disposes of an estimate on a finite observation length. Then the smallest eigenvalues are no longer exactly equal and a statistical test has to be used to decide on the number of equal latent roots.
Several tests have been proposed : sphericity test ([2], [4]) or Akaike's like criteria ([6], [7]).In most publications their performances are just evaluated on a limited set of computer simulations ([1] to [7]). Here, using a few results about the distribution of latent roots of complex Wishart matrix ([10], [11]), we derive approximate analytical expressions for these performances.

This paper is organized in the following way : section I recalls the formulation of the tests, section II gathers a few results on the statistical distribution of useful quantities, sections III and IV compute performances in detection of one or two sources,sections V and VI examine some effects of coherence models mismatches.

I - DESCRIPTION OF MODEL AND TEST

An array of K perfect sensors (linear, omnidirectionnal and point-like) is insonified by p perfectly coherent point-like sources which radiate uncorrelated signals superimposed on a background noise whose spatial coherence is exactly known. Then the cross-spectral density matrix takes the form ([1],[2]) :

$$\Gamma(f) = \sigma(f)\, J(f) + \sum_{i=1}^{p} \gamma_i(f)\, \underline{d}_i(f)\, \underline{d}_i^{+}(f) \quad (1)$$

where :
$\sigma(f)$ = spectral density of background noise
$J(f)$ = " "matrix " " "
$\gamma_i(f)$ = spectral density of source i
$\underline{d}_i(f)$ = position vector " "
($^+$ designs the complex conjugate and transpose and frequency f will now be omitted).
With this model when the background noise is incoherent or after a spatial "whitening" (pre and post-multiplication of Γ by C^{-1} and C^{-1+} such that $J = C.C^+$, see [1]) the latent roots of Γ are :

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p > \lambda_{p+1} = \ldots = \lambda_K = \sigma \quad (2)$$

The equality of the smallest (K-p) eigenvalues of Γ is used to determine the number of sources p.
Actually the true matrix is unknown and an estimate $\hat{\Gamma}$ is first obtained : the input signals (assumed stationnary and gaussian) are digitized and Fourier transformed at frequency f over N adjacent time intervals of equal length T into a set of N complex gaussian vectors :$\{\underline{X}_i(f), i = 1,N\}$ whose covariance matrix is $\Gamma(f)$.

$$\hat{\Gamma}(f) = \frac{1}{N} \sum_{i=1}^{N} \underline{X}_i(f)\, \underline{X}^{+}_i(f) \quad (3)$$

When the length of Fourier transforms T is large ([3]) the matrix $N.\hat{\Gamma}$ follows the complex Wishart distribution with N degrees of freedom and expectation $N\Gamma$ (see [10] , [11]).
Various objective techniques have been proposed to estimate the multiplicity of the smallest eigenvalue from $\hat{\Gamma}$ ([1]).
They all use statistics V_p which are directly related to the maximum of the likelihood function under the assumption H_p : "at most p sources" ([2])

$$V_p = (\hat{g}_p/\hat{a}_p)^{k-p} \quad (4)$$

where $\hat{g}_p$ and $\hat{a}_p$ are geometrical and arithmetical means of the (K-p) smallest latent roots of $\hat{\Gamma}$.

Sphericity test ([4] - Liggett)
The estimated number of sources $\hat{p}$ is the smallest value of p for which the assumption Hp is accepted

according to the test :

$$V_p \underset{H_p}{\overset{H_p}{\gtrless}} \eta_p \qquad (5)$$

The thresholds η_p are fixed by :

$$\text{Prob}\{ V_p < \eta_p / H_p \} = \alpha_p \qquad (6)$$

and α_p controls the significance level of the tests.

NB : this test has been derived without any hypothesis on the shapes of the wavefront.

II - SOME USEFUL PROPERTIES OF THE EIGENSYSTEM OF MATRIX Γ

In this section we briefly expose a few properties of latent roots of a complex Wishart matrix. All but the first are only valid when the observation time N.T becomes large ([11]).

Independence of eigenvalues and vectors

The distribution of the eigenvalues $\hat{\lambda}_i$ of matrix $\hat{\Gamma}$ is independent of the eigenvectors $\hat{v}_i$. It depends only on the latent roots λ_i of matrix Γ.

Well separated latent roots

When N is large it becomes possible to associate each eigenvalue of $\hat{\Gamma}$ to a corresponding one of Γ. Then eigenvalues of $\hat{\Gamma}$ which correspond to well-separated distinct roots of Γ are asymptotically (with N) independent random variable. Latent roots of Γ are "well separated" if their relative spacing $(\lambda_k-\lambda_l)/\lambda_k$ is large compared to $N^{-1/2}$. In this case the test will almost surely separate them.

Using this property it is often possible to reduce a complex noisefield with (p+1) sources to a simpler one :

- p strong and well-separated sources which induce p widely spaced largest roots and will always be detected,
- a weakest $(p+1)^{th}$ source whose detection is problematic.

The behaviour of statistic V_p, useful to detect this last source, may be evaluated with the following expressions.

Asymptotic distribution of V

For p = 0 we define :

$$M_0 = 2\,[N - (2K^2 + 1)/6K] \qquad (8)$$

and the matrix Ω, which takes into account all the phenomena which make the latent root σ not to have multiplicity K, by :

$$\Gamma = \sigma\,[I + M_0^{-1/2}\,\Omega_0] \qquad (9)$$

Then the quantity $-M_0 \cdot \text{Log}\, V_0$ is asymptotically (with fixed Ω_0 and increasing N or M_0) distributed as a non-central χ^2 (with ν_0 degrees of freedom and δ_0 as the non centrality parameter). More precisely :

$$\Pr\{-M_0 \text{Log}\, V_0 \geq \eta_0\} = \Pr\{\chi'^2\nu_0(\delta_0) \geq \eta_0\} + O(M^{-1/2}) \qquad (10)$$

$$\nu_0 = K^2 - 1$$
$$\sigma_k = \text{tr}\,(\Omega_0^k)$$
$$\delta_0 = [\sigma_2 - \sigma_1/K]/2 \qquad (11)$$

This approximation allows us :

- to fix the threshold η_0 as a function of α_0 (according to (6) with Ω_0 and δ_0 null and a central χ^2 - as in [1] or [4]).
- to obtain the power of the sphericity test (compute Pd, the probability of deciding $p \geq 1$, with η_0 fixed).

The demonstration of these formulae, that we did not find in the litterature for complex Wishart matrixes, is tedious but similar to the real case ([12]) and consists in expanding the characteristic function of $-M_0.\text{Log}\, V_0$ in increasing powers of $M_0^{-1/2}$. A more accurate expression, which is the same as in the real case ([12]pp347), may be obtained :

$$\Pr\{-M_0 \text{Log}\, V_0 \geq \eta_0\} = \Pr\{\chi'^2\nu_0(\delta_0) \geq \eta_0\} + M_0^{-1/2}\,[a_0 \Pr\{\chi'^2\nu_0(\delta_0) \geq \eta_0\} + a_1 \Pr\{\chi'^2\nu_0+2(\delta_0) \geq \eta_0\} + a_2 \Pr\{\chi'^2\nu_0+4(\delta_0) \geq \eta_0\} + O(M^{-1}) \qquad (12)$$

$$\text{where } a_0 = \sigma_3/3 - \sigma_1 \cdot \varepsilon \quad a_2 = \sigma_3/6 \quad a_1 = -\sigma_3/2 + \sigma_1\,\varepsilon \quad \varepsilon = (2K^2+1)/6K \qquad (13)$$

Asymptotic distribution of Vp

When the p largest eigenvalues of Γ are widely spaced from the (K-p) smallest ones the same type of approximate formulae may be derived. We just have to define Ω_p by :

$$\Gamma = \sum_{i=1}^{p} (\lambda_i - \sigma)\,\underline{v}_i\,\underline{v}_i + \sigma\,(I + M_p^{-1/2}\,\Omega_p) \qquad (14)$$

where $\underline{v}_i$ are the eigenvectors corresponding to source subspace, Ω_p is a (K-p) rank matrix and to replace K by (K-p) and N by (N-p) in (8) or (11)

III - DETECTION OF A SINGLE SOURCE

The matrix Γ takes the form :

$$\Gamma = \sigma \cdot I + \gamma \cdot \underline{d} \cdot \underline{d}^+ \qquad (15)$$

and its latent roots are :

$$\lambda_1 = \sigma + K\,\gamma > \lambda_2 = \ldots \lambda_K = \sigma \qquad (16)$$

Comparison with Monte Carlo results

Expressions (8) to (13) have been used to predict values of Pd ($\Pr \hat{p} \geq 1$). These predictions have been tested against Monte Carlo simulations with 10^3 experiments for each set of α,N and input SNR γ/σ. Figures 1 and 2 show some of the results with $\alpha = 10^{-3}$, N = 100 or 10^2 and a linear array of 4 or 16 half wavelengh equispaced sensors. Accuracy of both approximations has been found quite satisfactory, especially for large values of N/K.

Comparison with conventionnal detection methods

Sphericity test is now compared to a more conventionnal test r_c which uses the classical beamformer output. This test is obtai-

ned in computing the likelihood ratio of the two assumptions H_0 : "incoherent noise alone" or H_1 :"noise and one signal" with σ, γ and $\underline{d}$ (as in (15)) exactly known :

$$r_c = \frac{\underline{d}^+ \hat{\Gamma} \underline{d}}{\sigma} \underset{H_0}{\overset{H_1}{\gtrless}} \eta \qquad (17)$$

After multiplication by a proper constant, (which depends on the actual H_i), r_c is distributed as a central χ^2 with 2 K.N degrees of freedom which can be approximated by a normal distribution. Then the following relation may be derived between K,N, γ/σ and α (false alarm rate) when Pd equals 0.5.

$$N^{1/2} K \left(\frac{\gamma}{\sigma}\right) = Q^{-1}(\alpha) \qquad (18)$$

where $Q^{-1}(\alpha)$ is the α upper quantile of normal distribution.
Applying classical approximations for non-central χ^2 (see [12]) a rough equivalent of (18) can be obtained for the sphericity test from (8) to (11).

$$N^{1/2} K^{1/2} \left(\frac{\gamma}{\sigma}\right) = [\sqrt{2}\, Q^{-1}(\alpha)]^{1/2} \qquad (19)$$

Using (18)-(19) we can compare the performances of sphericity test and receiver r_c. The spatial gain of the receiver r_c grows faster with the number of sensors (K instead of $K^{1/2}$). We simply interpret this by saying that the receiver r_c uses the knowledge of wavefront shape and sums in a "coherent" way signals along the array. On the contrary receiver based on eigenvalues realizes an "incoherent" integration : the square modulus of non-diagonal elements of $\hat{\Gamma}$ is summed to get δ_0 in (10).

IV - RESOLVING POWER OF SPHERICITY TEST

The noisefield consists of two sources with the same spectral density γ spatially separated of $\Delta\Theta$ (small compared to $2\Theta_3$, the 3dB beamwidth of the directivity function). Expanding the directivity function in power of $(\Delta\Theta/2\Theta_3)$ one can easily obtain approximate expressions for the latent roots

$$\begin{aligned} \lambda_1 &\simeq \sigma + 2 \,.\, K.\, \gamma \\ \lambda_2 &\simeq \sigma + K \,.\, \gamma(\Delta\Theta/2\Theta_3)^2 \\ \lambda_3 = \lambda_4 &= \ldots\ldots = \lambda_K = \sigma \end{aligned} \qquad (20)$$

We assume γ/σ large enough for the receiver to detect almost surely at least one source. Then the problem is similar to the one of detecting a second weak source, well-separated of the first, with spectral density $\gamma(\Delta\Theta/2\Theta_3)^2$. Applying results of section II we obtain as in section III with a similar accuracy an equivalent of (19) :

$$N^{1/2} (K-1)^{1/2} \left(\frac{\gamma}{\sigma}\right) \left(\frac{\Delta\Theta}{2\Theta_3}\right)^2 \simeq [\sqrt{2}\, Q^{-1}(\alpha)]^{1/2} \qquad (21)$$

If one takes (from Rayleigh criterion) $2\Theta_3$ as the angular resolution of classical beamformer (equivalent to receiver r_c) one can obtain the gain in angular resolution of the sphericity test :

$$G = \frac{2\Theta_3}{\Delta\Theta} \simeq N^{1/4}(K-1)^{1/4} \left(\frac{\gamma}{\sigma}\right)^{1/2} (\sqrt{2}\, Q^{-1}(\alpha)) \qquad (22)$$

This gain, which is characteristic of the "high resolution" properties of test, depends strongly on the values of quantities N and γ/σ. In usual conditions (as N = 100, $\alpha = 10^{-3}$, 8 sensors and $\gamma/\sigma = 1$) one can obtain G ≃ 4.

V - SENSITIVITY TO BACKGROUND NOISE COHERENCE MISMATCHES

In preceding sections the background noise was assumed to be perfectly spatially "whitened". In fact,the exact matrix J is unknown and spatial whitening will be realized with a hypothetic matrix J_0, distinct from the true matrix J. Without any source and after spatial whitening with $C_0(J_0 = C_0\, C_0^+)$:

$$\Gamma = \sigma\,(I + \Omega) \qquad (23)$$

where $\Omega = C_0^{-1}\,(J - J_0)\, C_0^{-1+}$ is a perturbation matrix which takes into account error on J. This error will be interpreted (for large N and Ω) as one or several coherent sources ([1]) whose probabilities of detection can be predicted as in section III.
To illustrate this we use the surface noise model due to Cron and Sherman ([9]) whose horizontal coherence function depends on a parameter s :

$$coh_s(x) = \frac{2^s\, s!}{(2\pi\, x/\lambda)^s}\, Bes_s\,(2\pi\, x/\lambda) \qquad (24)$$

where :
x/λ = horizontal distance measured in wavelengths
Bes_s = first kind Bessel function of order m

Computer simulations, similar to the ones in section III, have been conducted with true parameter s varying around hypothetic s_0. The agreement between predicted and experimental values of Pd is excellent. The robustness of the sphericity test is rather bad and decreases with K and N : for s_0 = 0.5, K = 4, N = 100 and $\alpha = 10^{-3}$ at least one false source is detected with Pd = 0.5 when s is out of the interval 0.27 to 0.82 ; for K=16 and N=10 this interval becomes 0.44 to 0.55.

VI - SOURCE COHERENCE MISMATCHES

Partial decoherence of source signals has two effects :
- the spectral density matrix of a single source is not dyadic. So it may be interpreted by the test as several "false" sources,
- the probability of detection may also decrease because part of signal energy is converted into incoherent noise.

Spectral bias decoherence

This phenomenon, already discussed in [3], is due to the finite length of Fourier transforms T. For a linear array, a significant parameter is :

$$x = \frac{d}{c\,.\,T} \sin\Theta \qquad (25)$$

where d is the space between two sensors, c the sound speed and Θ the source bearing (from broadside).
From the coherence matrix shape given in [3] and results from section III one can get eigenvalues of a matrix Γ corresponding to a single source with bias decoherence in a white noise. The value of parameter δ_1, which characterizes detection of a second false source is given by :

$$\delta_1 = N \left(\frac{\gamma}{\sigma}\right)^2 \frac{K.(K^2-1)(K-2)(K+7)}{90}\, x^2 \qquad (26)$$

This expression is useful to evaluate the minimum value of T which insures that a second detected source is a real one and not an artefact due to spectral bias decoherence of the first one.

Source motion decoherence

One can easily demonstrate that in a first order

approximation the same spectral matrix Γ characterizes the two following noisefields :
- a simple source whose bearing linearly varies during the measurement interval from $\Theta_0 - \Delta\Theta/2$ to $\Theta_0 + \Delta\Theta/2$
- two fixed sources with respective bearings $\Theta_0 \pm \Delta\Theta/(2\sqrt{3})$.

From this one can obtain a constraint on the bearing rate $\dot{\Theta}$ of a strong single source with spectral density γ in order to reliably detect a second well-separated weaker source with density γ' :

$$\frac{\dot{\Theta} \cdot N \cdot T}{2\Theta_3} \leq \left(\frac{\gamma'}{\gamma}\right) \qquad (27)$$

This last constraint may be very restrictive. For a linear array of 8 or 32 half wavelength equispaced sensors at f = 375 Hz, two sources with input SNR 0 and - 13 dB, Pd = 0.5 and $\alpha = 10^{-3}$ one finds :

$$N.T \geq \begin{cases} 39 \text{ sec} \\ 65 \text{ sec} \end{cases} \quad \dot{\Theta} \leq \begin{cases} 340 \text{deg/hour} & (8 \text{ sensors}) \\ 43 \text{ " } \text{ " } & (32 \text{ " } \text{ " }) \end{cases}$$

CONCLUSIONS

Similar results to the ones shown here for the sphericity test have been obtained for other tests like AIC or MDL ([1],[6],[7]).Expressions are more complicated but they show very close performances for the three tests.
Eigenvalue methods may also be extended to broadband detection. Then their performances are roughly the same as in narrow band with an additionnal gain on input SNR of sources in $M^{1/4}$ (M is the number of independent frequency bins).
All these results may lead to a very mitigated judgement about the use of eigenvalues detection methods in underwater passive listening.

They certainly have more angular resolution than the classical beamformer but their performance on well-separated sources is poorer. Their lack of robustness to any coherence mismatche may increase false alarm rate.
In fact one has to recall that :
- eigenvalues detection is just the first stage of a complete high resolution localisation method : the number of sources is only used to partition eigenvectors.
- there are situations where wavefront shapes are unknown and classical beamforming not feasible (array of unknown geometry, severe multipath propagation, geophysics, etc,...)

Otherwise an excellent remedy, already discussed in [1] or [8], is to split the antenna into several subarrays steered in a given look direction and to use subarray outputs instead of sensors. Then a "partially coherent" integration of signals is made along the antenna and performances of tests on sources is improved. Robustness to coherence mismatches also significantly increases.

REFERENCES

[1] L. KOPP, G. BIENVENU "Multiple detection using eigenvalues when the noise spatial coherence is partially unknown "NATO ASI on signal processing, Luneburg (1984)
[2] G. BIENVENU, L. KOPP "Optimality of high resolution array processing using the eigensystem approach "IEEE trans on ASSP, Vol ASSP-31 ,3 (1983)
[3] G. BIENVENU, L. KOPP "Bias and variance effects on the eigensystem of the cross-spectral matrix" Proc. of Underwater Acoustic Group Conf., London (1982)
[4] W.S LIGGETT "Passive Sonar : fitting models to multiple time series" NATO ASI on signal processing. Loughborough (UK) (1972)
[5] R.O. SCHMIDT "A signal subspace approach to multiple emitters location "Ph D. Dissert Standford (1981)
[6] M.WAX,T.KAILATH "Determining the number of signals by information theoretical criteria" Proc. ASSP Spectrum Estimation Workshop. Tampa (FL) (1983)
[7] G. VEZZOZI, P. NICOLAS "Separation de fronts d'onde correlés" Proc of GRETSI, NICE (1983)
[8] G. BIENVENU, L.KOPP "Methodes haute resolution après formation de voies" Proc of GRETSI NICE (1985)
[9] B.F. CRON, C.H. SHERMAN "Spatial correlation functions for various noise models" JASA Vol 34, 11 (1962)
[10] N.R. GOODMAN "Statistical analysis based on a certain multivariate complex gaussian distribution (an introduction)" Ann. Math. Statist. Vol 34 pp 155-177 (1963)
[11] R.J. MUIRHEAD "Aspects of Multivariate statistical theory" J. Wiley ed. New-York (1982)
[12] N.L. JOHNSON, S. KOTZ "Distributions in statistics" J. Wiley ed., New-York (1970)

ACKNOWLEDGEMENTS

This work has been supported by G.E.R.D.S.M. (SIX-FOURS LES PLAGES - DCAN TOULON)

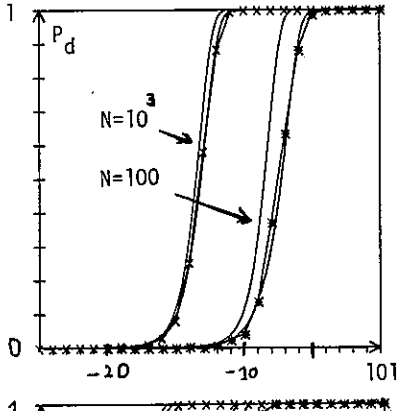


Figure 1 :

4 sensors

$\alpha = 10^{-3}$

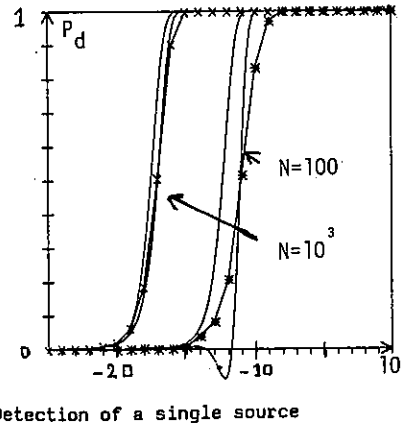


Figure 2 :

16 sensors

Detection of a single source

Pd is plotted versus input SNR γ/σ :
- theoretical values (I = expressions (8) to (11) ; II = more accurate ones (13))

* experimental values

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

EFFECTIVE INVARIANT COUPLING TECHNIQUE FOR DESIGNING THE NEW OR IMPROVED STATISTICAL PROCEDURES OF DETECTION AND ESTIMATION IN SIGNAL PROCESSING SYSTEMS

N. A. NECHVAL

Department of Control Systems, Civil Aviation Engineers Institute
Riga, USSR

The objective of this paper is to call attention to an effective technique of invariant coupling, which is developed by the author, for designing the new or improved statistical procedures of detection and estimation, concentrating on mathematical or statistical details as well as on applications of these procedures in signal processing systems. A given technique has the advantages over the known techniques: maximum likelihood (as well as minimum-variance unbiased estimation (MVUE)) and Bayesian. On the one hand, it opens up possibility for more complete extracting a useful information from data of observations in comparison with the maximum likelihood and MVUE techniques, and, on the other hand, it rules out the subjectivity of investigator (a limitation of the Bayesian technique) which is introduced through a priori distribution. An efficiency of decision rules, which are generated by the new or improved statistical procedures constructed through the use of a suggested technique, is displayed noticeably when there are the cases of small sample sizes of statistical data of observations.

1. INTRODUCTION

Depending on the type of signal of interest, it may be classified detection and estimation problems into three categories, as follows: 1) known signals in noise, 2) signals with unknown parameters in noise, 3) random signals in noise. The present paper is concerned with case 2). As a rule, in this case, when the signal models are applied to solve real-world problems, the parameters are estimated and then treated as if they were the true values. The risk associated with using estimates rather than the true parameters is called estimation risk and is often ignored. When data is limited and/or unreliable, estimation risk may be significant, and failure to incorporate it into the statistical procedures of detection and estimation may lead to serious errors. Its explicit consideration is important since decision procedures which are optimal in the absence of uncertainty need not even be approximately optimal in the presence of such uncertainty. The examples are given where using estimates as if they were the true parameters leads to poor results. The principal purpose of a technique, which is here submitted for discussion, is to provide, in above mentioned case 2), a set of statistical procedures (for carrying out the necessary detection and estimation processes) which are basically intended to detect the presence of a signal and to extract information of interest about the signal from data so correctly and completely as it is possible in order to ensure an optimal performance of signal processing systems. A proposed technique is based on the statistical invariant coupling method [1] which represents the constructive development of invariance principle in mathematical statistics. A fundamental substance of the invariant coupling method consists in that here it is introduced the idea of "invariant coupling" a nuisance (unknown) parameter (vector, in general) with a sufficient statistic (vector, in general), i.e., the idea of forming a some function of a nuisance parameter and sufficient statistic (an "invariantly coupled sufficient statistic") on the basis of the underlying probability distribution which is invariant under a group of transformations, and it is defined an "invariant free statistic" (a some function of statistical data) which represents an ancillary statistic in relation to invariantly coupled sufficient statistic. Both these statistics (an invariantly coupled sufficient statistic and invariant free statistic) are generated by the inva-

riant coupling method and have the sampling distributions independent of a nuisance parameter. In given case a property of invariance in inference or decision problems insures that the error probabilities or risk functions, generated by the invariant coupling method, will be independent of the nuisance parameters. This makes it possible readily and simply to design the new or improved statistical decision procedures generating the efficient decision rules which are admissible and/or minimax.

2. PRINCIPLE OF INVARIANCE

If the decision problem is symmetric, or invariant, with respect to certain operations, then it may seem reasonable to restrict the available rules to be symmetric, or invariant, with respect to those operations also. The principle of invariance involves groups of transformations over the three spaces: the parameter space Θ, the decision space $\mathcal{D}$, and the sample space $\mathcal{X}$, an n-dimensional Euclidean space. For instance, let Θ be a real line, and θ be a scale parameter of the observable random variable X. If the decision problem is invariant under the group G of transformations $g_{c_1}(X) = c_1 X$ $(X \in \mathcal{X})$, where $c_1 > 0$, with $\bar{g}_{c_1}(\theta) = c_1\theta$ $(\theta \in \Theta)$, $\bar{g} \in \bar{G}$, then the distribution of $g_{c_1}(X)$ given $\bar{g}_{c_1}(\theta)$ is the same as the distribution of X given θ, since θ is a scale parameter, and, furthermore, a loss function $I_\theta(d)$ $(d \in \mathcal{D})$ of the decision problem is invariant under G, i.e.,

$$I_{\bar{g}_{c_1}(\theta)}(g_{c_1}(d)) = I_\theta(d). \tag{1}$$

3. INVARIANT COUPLING TECHNIQUE

The idea of the invariant coupling technique consists in that we introduce a random variable X into a loss function $I_\theta(d)$ by means of transformation $g \in G$ (accordingly, $\bar{g} \in \bar{G}$) and obtain

$$I_{\bar{g}_X(\theta)}(g_X(d)) = I_\theta(d) \tag{2}$$

where $\bar{g}_X(\theta) \equiv V$ is the invariant coupling: the random variable X and the nuisance parameter θ (the invariantly coupled random variable V). The sampling distribution of V is independent of a nuisance parameter θ. The invariant coupling technique generates a risk function which is determined as follows:

$$R^*(g_X(\delta)) = E_X\{I_V(g_X(\delta))\} \tag{3}$$

where $E_X\{\cdot\}$ is the expected value operator with respect to V. Now the admissible decision rule δ^* (with respect to a random variable X), if it exists, can be obtained by minimizing the (3) as a function of δ, i.e.,

$$\delta^* = \arg \inf_{\{\delta:\delta\in\mathcal{D}\}} R^*(g_X(\delta)). \tag{4}$$

Here the following theorem is true.

Theorem. Suppose above assumptions hold. Let $X^n \in \mathcal{X}$ be an n-dimensional vector (a sample of size n of statistical data). Let $T = T(X^n)$ be a complete sufficient statistic for $\theta \in \Theta$. If the nuisance parameter θ is one-dimensional, then

$$\delta^* = \arg \inf_{\{\delta:\delta\in\mathcal{D}\}} R^*(g_T(\delta)) \tag{5}$$

is the admissible decision rule (where

$$R^*(g_T(\delta)) = E_T\{I_V(g_T(\delta))\}, \tag{6}$$

$V = \bar{g}_T(\theta)$ is the invariantly coupled sufficient statistic).

Proof. The proof being straightforward is omitted.

4. SIGNAL PARAMETER ESTIMATION

A commonly used model for signal fading in many types of communication channels is that the amplitude of the received signal at a given time is a Rayleigh-distributed random variable. Here we show how the invariant coupling technique may be applied to the problem of estimating the Rayleigh distribution parameter. Let $X^n = (X_1, \ldots, X_n)$ denotes the sample of the independent, identically distributed Rayleigh observations. Each observation has a density function

$$f_\theta(x) = \frac{x}{\theta^2} e^{-x^2/2\theta^2}, \quad x \in (0,\infty). \tag{7}$$

By independence, the joint density function is the product of the individual density functions and is given by

$$f_\theta(X^n) = \prod_{i=1}^{n} \frac{X_i}{\theta^2} e^{-X_i^2/2\theta^2} = \prod_{i=1}^{n} X_i \frac{1}{\theta^{2n}} e^{-\frac{1}{2\theta^2}\sum_{i=1}^{n} X_i^2} \tag{8}$$

As is seen from (8),

$$T(X^n) = \sum_{i=1}^{n} X_i^2 \qquad (9)$$

is a sufficient statistic for the parameter Θ. This statistic can be easily shown to be complete. The density function of T is

$$\varphi_\theta(t) = \frac{1}{\Gamma(n)(2\theta^2)^n} t^{n-1} e^{-\frac{t}{2\theta^2}}, \quad t \in (0,\infty). \qquad (10)$$

4.1. Maximum Likelihood Estimator

The maximum likelihood estimator for θ which is obtained by maximizing the $f_\theta(X^n)$ function as a function of Θ, is

$$\check{\theta} = \sqrt{\frac{1}{2n}\left(\sum_{i=1}^{n} X_i^2\right)}. \qquad (11)$$

This estimator can be shown to be consistent. Expected value of $\hat{\theta}$ is

$$E_\theta\{\check{\theta}\} = \int_0^\infty \sqrt{\frac{t}{2n}}\, \varphi_\theta(t)dt = \theta \frac{\Gamma(n+\frac{1}{2})}{\Gamma(n)\sqrt{n}}. \qquad (12)$$

The factor $(\Gamma(n+\frac{1}{2})/\Gamma(n)\sqrt{n})$ tends to unity as n approaches infinity, which verifies that $\hat{\theta}$ is asymptotically unbiased.

4.2. Unbiased Estimator

An unbiased estimator $\hat{\theta}$ for θ, which is introduced as

$$E_\theta\{\hat{\theta}\} = \int_0^\infty \hat{\theta}(t)\varphi_\theta(t)dt = \theta, \qquad (13)$$

can be easily obtained with the aid of the invariant coupling technique as follows: from (13)

$$\int_0^\infty \frac{\hat{\theta}}{\theta}\varphi_\theta(t)d(t) = 1. \qquad (14)$$

Now we carry out the following transformation of (14):

$$\int_0^\infty \frac{\hat{\theta}}{\theta}\varphi_\theta(t)dt = \frac{\hat{\theta}}{T^{1/2}} \int_0^\infty \frac{t^{1/2}}{\theta} \frac{t^{n-1}e^{-t/2\theta^2}}{\Gamma(n)(2\theta^2)^n} dt$$

$$= \frac{\hat{\theta}}{T^{1/2}} \frac{1}{\Gamma(n)2^n} \int_0^\infty v^{n-\frac{1}{2}} e^{-\frac{v}{2}} dv$$

$$= \frac{\hat{\theta}}{T^{1/2}} \frac{\Gamma(n+1/2)\,2^{1/2}}{\Gamma(n)} = 1 \qquad (15)$$

where $V = T/\theta^2$. From (15) we have the estimator

$$\hat{\theta} = \frac{\Gamma(n)}{\Gamma(n+\frac{1}{2})\sqrt{2}} \sqrt{\sum_{i=1}^{n} X_i^2} \qquad (16)$$

which is the minimum-variance unbiased estimator (MVUE). Similarly it can be obtained the MVUE $\hat{f}_T(x)$ of (7). Since

$$E_\theta\{\hat{f}_T(x)\} = \int_0^\infty \hat{f}_t(x)\varphi_\theta(t)dt = f_\theta(x), \qquad (17)$$

$$\int_0^\infty \frac{\hat{f}_t(x)}{f_\theta(x)}\varphi_\theta(t)dt = 1. \qquad (18)$$

Hence it follows

$$\int_0^\infty \frac{\hat{f}_t(x)}{f_\theta(x)}\varphi_\theta(t)dt = \frac{\hat{f}_T(x)}{x}\int_0^\infty \frac{t^{n-1}e^{-\frac{t-x^2}{2\theta^2}}}{\Gamma(n)2^n(\theta^2)^{n-1}}dt$$

$$= \frac{\hat{f}_T(x)}{x} \frac{T^{n-1}}{\Gamma(n)2^n[(T-x^2)_+]^{n-2}} \int_0^\infty v^{n-2}e^{-\frac{v}{2}}dv$$

$$= \frac{\hat{f}_T(x)}{x} \frac{T^{n-1}}{2(n-1)[(T-x^2)_+]^{n-2}} = 1 \qquad (19)$$

where $V = \frac{T-x^2}{\theta^2}$, $a_+ = \max(0,a)$. From (19),

$$\hat{f}_T(x) = \frac{2(n-1)x}{T}\left[\left(1-\frac{x^2}{T}\right)_+\right]^{n-2} \qquad (20)$$

where x^2/T is an invariant free statistic.

4.3. Admissible Estimator

A minimum-variance unbiased estimator is not necessarily a good estimator. Suppose that a loss function is

$$I_\theta(d) = \frac{(d-\theta)^2}{\theta^2} \qquad (21)$$

which is invariant under the group G of above transformations, i.e.,

$$I_{\bar{g}_{c_1}(\theta)}(g_{c_1}(d)) = \frac{(c_1 d - c_1\theta)^2}{(c_1\theta)^2} = \frac{(d-\theta)^2}{\theta^2}$$

$$= I_\theta(d) \qquad (22)$$

where a decision $d \in \mathcal{D} = (0,\infty)$. Using the invariant coupling technique we have

$$I_{\bar{g}_T(\theta)}(g_T(\delta)) = \frac{\delta^2}{T}\frac{T}{\theta^2} - 2\frac{\delta}{T^{1/2}}\frac{T^{1/2}}{\theta} + 1$$

$$= \frac{\delta^2}{T}V - 2\frac{\delta}{T^{1/2}}V^{1/2} + 1 \equiv I_V(g_T(\delta)). \qquad (23)$$

From (23), a risk, generated by the invariant coupling technique, can be found to be

$$R^*(g_T(\delta)) = E_T\{I_V(g_T(\delta))\}$$

$$= \frac{\delta^2}{T}2n - 2\frac{\delta}{T^{1/2}}\frac{\Gamma(n+\frac{1}{2})}{\Gamma(n)}\sqrt{2} + 1. \qquad (24)$$

Hence it follows the decision rule

$$\delta^* = \arg \inf_{\{\delta:\delta\in\mathcal{D}\}} R^*(g_T(\delta)) = \frac{\Gamma(n+\frac{1}{2})}{\Gamma(n+1)\sqrt{2}}\sqrt{\sum_{i=1}^{n} x_i^2} \qquad (25)$$

which is the admissible estimator for

θ (with respect to (21)). A measure of the degree (with respect to δ^*) to which the estimator $\hat{\theta}$ can be used as a proxy for the true value of θ is given by

$$\eta_{\hat{\theta}} = \frac{R_\theta(\delta^*) - I_\theta(d^*)}{R_\theta(\hat{\theta}) - I_\theta(d^*)} 100\% = \frac{1 + \frac{\Gamma(n+1/2)}{\Gamma(n)\sqrt{n}}}{2} 100\% \quad (26)$$

where

$$R_\theta(\delta) = E_\theta\{I_\theta(\delta)\} \quad (27)$$

is a risk function of decision rule δ; d^* is an optimal decision, i.e.,

$$d^* = \arg \inf_{\{d: d \in \mathcal{D}\}} I_\theta(d) = \Theta . \quad (28)$$

It can be shown that

$$\eta_{\hat{\theta}} = \frac{R_\theta(\delta^*) - I_\theta(d^*)}{R_\theta(\hat{\theta}) - I_\theta(d^*)} 100\% = \frac{\Gamma^2(n+\frac{1}{2})}{\Gamma^2(n)\, n} 100\% \quad (29)$$

and $\eta_{\delta^*} = 100$ %. For example, suppose that $n = 1$. Then

$$\eta_{\hat{\theta}} = 94.31\% ; \quad \eta_{\hat{\theta}} = 78.54\% . \quad (30)$$

4.4. Optimal Confidence Interval

Confidence interval for the parameter θ may be determined from knowledge of the distribution of $V = T/\theta^2$. It follows from (10) that the probability density function of V is given by

$$\varphi_n(v) = \frac{1}{\Gamma(n) 2^n} v^{n-1} e^{-\frac{v}{2}} dv, \quad v \in (0, \infty). \quad (31)$$

This implies

$$\varphi_T^*(\theta) = \frac{T^n}{\Gamma(n) 2^{n-1}} \left(\frac{1}{\theta^2}\right)^{n+\frac{1}{2}} e^{-\frac{T}{2\theta^2}} d\theta, \quad \theta \in (0, \infty) \quad (32)$$

(the probability density function of θ generated by the invariant coupling technique). Let the symbol I denote a confidence interval $(\underline{\theta}(T) \equiv a, \overline{\theta}(T) \equiv b)$ and γ the confidence coefficient. Then a confidence interval I_{ML}, which is to have minimum length and confidence coefficient γ, can be obtained by minimizing

$$\phi(a, b, \lambda) = b - a + \lambda \left(\int_a^b \varphi_T^*(\theta) d\theta - \gamma \right)$$

$$= b - a + \lambda \left(\int_{T/b^2}^{T/a^2} \varphi_n(v) dv - \gamma \right) \quad (33)$$

with respect to a, b, λ where λ is a Lagrange multiplier. The resulting conditions for a and b, determining I_{ML}, can be written as follows:

$$I_{ML}: \quad \varphi_n\left(\frac{T}{a^2}\right) \frac{1}{a^3} = \varphi_n\left(\frac{T}{b^2}\right) \frac{1}{b^3}, \quad \int_{T/b^2}^{T/a^2} \varphi_n(v) dv = \gamma . \quad (34)$$

For example, assume a restriction to interval of the form

$$I = \left(a = \sqrt{\frac{T}{a_1}}, \; b = \sqrt{\frac{T}{b_1}}\right). \quad (35)$$

This implies (from (34))

$$I_{ML}: \quad \varphi_{n+\frac{3}{2}}(a_1) = \varphi_{n+\frac{3}{2}}(b_1), \quad \int_{b_1}^{a_1} \varphi_n(v) dv = \gamma . \quad (36)$$

5. SETTING OF DETECTION THRESHOLD

The totality of all possible receiver inputs when noise alone is present is called "Population N"; similarly, the collection of all receiver inputs when signal plus noise is present is called "Population SN". The observer must judge from which population the receiver input came. Let us assume that receiver input is either

$$X_N \sim f_\theta(x) = \frac{x}{\theta^2} e^{-x^2/2\theta^2}, \; x \in (0, \infty) \; (\theta = \theta_N) \quad (37)$$

or

$$X_{SN} \sim f_\theta(x) = \frac{x}{\theta^2} e^{-x^2/2\theta^2}, \; x \in (0, \infty) \; (\theta = \theta_{SN}) \quad (38)$$

where $\theta_{SN} > \theta_N$. For the detection of signal in noise, we will adopt the Neyman-Pearson philosophy of hypothesis testing. Then the probability of a false alarm (alarm when $\theta = \theta_N$) is usually designated by α and is given by

$$P_\theta^{FA}(h) = \int_h^\infty \frac{x}{\theta_N^2} e^{-x^2/2\theta_N^2} dx = \alpha \quad (39)$$

where h is a detection threshold. The probability of a false dismissal (dismissal when $\theta = \theta_{SN}$) is designated by β and is given by

$$P_\theta^{FD}(h) = \int_0^h \frac{x}{\theta_{SN}^2} e^{-x^2/2\theta_{SN}^2} dx = \beta . \quad (40)$$

As a rule, both parameters θ_N and θ_{SN} are unknown. Suppose that there is a sample $X^n = (X_1, \ldots, X_n)$ of observations of X_N. Then, using the invariant coupling technique, we have (at a given level of α) a detection threshold

$$h^* = \arg \left(E_T\{P_V^{FA}(q_T(h))\} = \alpha \right)$$

$$= \left[(\alpha^{-1/n} - 1) \sum_{i=1}^n X_i^2 \right]^{1/2} . \quad (41)$$

REFERENCES

[1] Nechval, N. A., Modern Statistical Methods of Operations Research (RCAEI, Riga, 1982).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A FAST RECURSIVE APPROACH TO FIR SYSTEM IDENTIFICATION

Peter Strobach

SIEMENS Information Systems Laboratory
ZT ZTI INF 121, Otto-Hahn-Ring 6
D-8000 München 83, West Germany

ABSTRACT: Recently developed algorithms for least squares (LS) estimation of autoregressive (AR) models are extended in this paper so as to facilitate the identification of finite impulse response (FIR) models. The algorithms presented here are based on the new algebraic method of generalized residual energies (GREs) which received attention recently in the derivation of covariance ladder algorithms with excellent numerical properties, hence our new FIR system identification algorithms share the computational efficiency and good numerical behaviour of the latter. It is emphasized that our FIR system identification algorithms can also be used for joint process estimation since both estimation problems are algorithmically equivalent. Two versions are presented - one for the true LS recursive tracking and the other for identifying models with slowly time-varying parameters. This study can ultimately lead to a robust fixed-point implementation of FIR system identification.

1. INTRODUCTION

The FIR-filter is frequently used in context with the identification of time-varying systems. Fig. 1.1 is a block diagram of a FIR system identification model. Both the input and the output of the unknown system are accessible. It is desired to model the unknown system as a FIR-filter with a sufficiently large order p.

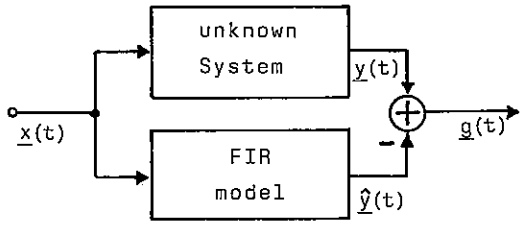


Fig. 1.1: FIR system identification model.

The approach used here is to determine the FIR model parameters such that the LS cost function

$$G(t) = \underline{g}^T(t)\underline{g}(t) \quad (1.1)$$

$$\underline{g}(t) = [g_0(t), g_1(t), \ldots, g_L(t)]^T \quad (1.2)$$

is minimized. Efficient recursive algorithms can be developed when the covariance matrix $W_{i,j}(t)$ of the input process $\underline{x}(t)$ and the cross correlation vector $\underline{v}(t)$ of $\underline{x}(t)$ and $\underline{y}(t)$ are known.

$$W_{i,j}(t) = \underline{x}^T(t-i)\underline{x}(t-j) \quad (1.3)$$

$$0 \leq i,j \leq p$$

$$v_j(t) = \underline{y}^T(t)\underline{x}(t-j) \quad (1.4)$$

where $\underline{x}(t) = [x(t), x(t-1), \ldots, x(t-L)]^T$ (1.5)

$\underline{y}(t) = [y(t), y(t-1), \ldots, y(t-L)]^T$ (1.6)

FIR system identification algorithms have previously been developed using either a tapped-delay-line based approach /3,4/ or the more promising normalized LS ladder approach /5/. Both methods require the direct computation of residuals for updating the FIR system parameters. Such procedures are not optimum in the sense of minimum round-off error sensitivity and dynamic range requirements as pointed out in /7,10/.
The FIR system identification problem can be treated as a part of the (more general) joint process estimation problem. Therefore in the following we concentrate our attention on the derivation of two joint process ladder algorithms based on the numerically improved method of GREs, while the FIR system identification problem is discussed as a special application of these new algorithms. Fig. 1.2 shows the application of a joint process ladder algorithm so as to facilitate FIR system identification. The AR-input of the joint process ladder algorithm must be connected to the input of the unknown system. The output of the unknown system drives the joint-input of the joint ladder algorithm.

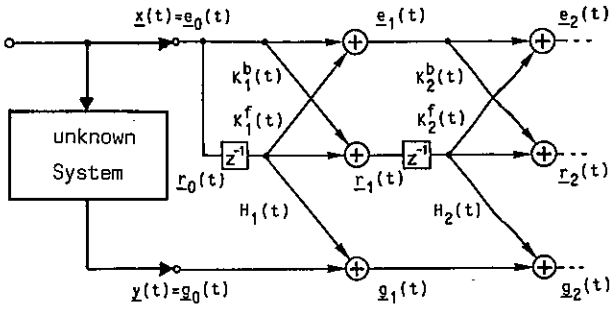


Fig. 1.2: FIR system identification using a joint process ladder algorithm.

The AR-part of the joint process ladder algorithm is required to calculate the adaptive Gram-Schmidt transformation of the system input $\underline{x}(t)$, while the FIR system parameters are estimated by the joint-part. In the stationary case the FIR system parameters are simply obtained as the response of the joint-part, when the AR-part is driven by a unit pulse.

2. THE NEW JOINT RECURSIONS

The joint ladder form shown in fig. 1.2 satisfies the following vector order recursions

AR-part:

$$\underline{e}_0(t) = \underline{r}_0(t) = \underline{x}(t) \tag{2.1a}$$

$$\underline{e}_m(t) = \underline{e}_{m-1}(t) + K_m^f(t)\underline{r}_{m-1}(t-1) \tag{2.1b}$$

$$\underline{r}_m(t) = \underline{r}_{m-1}(t-1) + K_m^b(t)\underline{e}_{m-1}(t) \tag{2.1c}$$

joint-part:

$$\underline{g}_0(t) = \underline{y}(t) \tag{2.2a}$$

$$\underline{g}_m(t) = \underline{g}_{m-1}(t) + H_m(t)\underline{r}_{m-1}(t-1) \tag{2.2b}$$

Where $\underline{e}_m(t)$ and $\underline{g}_m(t)$ are the forward residual vectors of the AR-part and the joint-part, respectively. $\underline{r}_m(t)$ is the backward residual vector. Efficient procedures for estimation of the AR reflection coefficients $K_m^f(t)$ and $K_m^b(t)$ are the subject of another paper /11/. We are interested in the estimation of the joint reflection coefficient $H_m(t)$. Minimizing the LS cost function of the joint-part

$$G_m(t) = \underline{g}_m^T(t)\underline{g}_m(t) \tag{2.3}$$

yields the desired rule for adjusting the joint reflection coefficient

$$H_m(t) = -\frac{\underline{g}_{m-1}^T(t)\underline{r}_{m-1}(t-1)}{\underline{r}_{m-1}^T(t-1)\underline{r}_{m-1}(t-1)} \tag{2.4}$$

We are seeking for order recursions of the nominator and the denominator inner products. Following this way, we can formally state the inner products of subsequent residual vectors

$$\underline{g}_m^T(t-i)\underline{e}_m(t-j) = \underline{g}_{m-1}^T(t-i)\underline{e}_{m-1}(t-j) + \tag{2.5a}$$
$$+ H_m(t-i)\underline{r}_{m-1}^T(t-1-i)\underline{e}_{m-1}(t-j) +$$
$$+ K_m^f(t-j)\underline{g}_{m-1}^T(t-i)\underline{r}_{m-1}(t-1-j) +$$
$$+ H_m(t-i)K_m^f(t-j)\underline{r}_{m-1}^T(t-1-i)\underline{r}_{m-1}(t-1-j)$$

$$\underline{g}_m^T(t-i)\underline{r}_m(t-1-j) = \underline{g}_{m-1}^T(t-i)\underline{r}_{m-1}(t-2-j) + \tag{2.5b}$$
$$+ H_m(t-i)\underline{r}_{m-1}^T(t-1-i)\underline{r}_{m-1}(t-2-j) +$$
$$+ K_M^b(t-1-j)\underline{g}_{m-1}^T(t-i)\underline{e}_{m-1}(t-1-j) +$$
$$+ H_m(t-i)K_m^b(t-1-j)\underline{r}_{m-1}^T(t-1-i)\underline{e}_{m-1}(t-1-j)$$

According to the method of generalized residual energies the following GREs are introduced

$$D_{m,i,j}(t) = \underline{g}_m^T(t-i)\underline{r}_m(t-1-j) \tag{2.6a}$$

$$A_{m,i,j}(t) = \underline{g}_m^T(t-i)\underline{e}_m(t-j) \tag{2.6b}$$

$$R_{m,i,j}(t) = \underline{r}_m^T(t-1-i)\underline{r}_m(t-1-j) \tag{2.6c}$$

$$C_{m,i,j}(t) = \underline{e}_m^T(t-i)\underline{r}_m(t-1-j) \tag{2.6d}$$

Using these definitions we can ultimately state the desired order recursions of the joint-GREs $D_{m,i,j}(t)$ and $A_{m,i,j}(t)$

$$D_{m,i,j}(t) = D_{m-1,i,j+1}(t) + \tag{2.7a}$$
$$+ H_m(t-i)R_{m-1,i,j+1}(t) + K_m^b(t-1-j)A_{m-1,i,j+1}(t) +$$
$$+ H_m(t-i)K_m^b(t-1-j)C_{m-1,j+1,i}(t)$$

$$A_{m,i,j}(t) = A_{m-1,i,j}(t) + \tag{2.7b}$$
$$+ H_m(t-i)C_{m-1,j,i}(t) + K_m^f(t-j)D_{m-1,i,j}(t) +$$
$$+ H_m(t-i)K_m^f(t-j)R_{m-1,i,j}(t)$$

The order recursions of $R_{m,i,j}(t)$ and $C_{m,i,j}(t)$ can be gathered from /11/. Regarding the definitions (2.6a,c) we conclude that the joint reflection coefficient $H_m(t)$ can also be expressed in terms of GREs

$$H_m(t) = -D_{m-1,0,0}(t)/R_{m-1,0,0}(t) \tag{2.8}$$

Combining the relations (2.7a,b) and (2.8) together with the AR-recursions given in paper /11/ already establishes a closed recursion for the true LS estimation of $H_m(t)$. As $\underline{g}_0(t-i)=\underline{y}(t-i)$ and $\underline{e}_0(t-j)=\underline{r}_0(t-j)=\underline{x}(t-j)$ we constitute the initialization scheme of the joint recursions from the cross covariance vector $\underline{v}(t)$

$$D_{0,0,j}(t) = v_{j+1}(t) \tag{2.9a}$$

$$A_{0,0,j}(t) = v_j(t) \tag{2.9b}$$

Note that only the first row of the GREs $A_{m,i,j}(t)$ and $D_{m,i,j}(t)$ will be initialized by (2.9a,b). Indeed, evaluating the expressions (2.7a,b) it becomes evident, that only the first row vectors $A_{m,0,j}(t)$ and $D_{m,0,j}(t)$ will be sufficient to aid in the order recursions of the joint-part.

Thus (2.7a,b) reduces to

$$D_{m,0,j}(t) = D_{m-1,0,j+1}(t) + \tag{2.10a}$$
$$+ H_m(t)R_{m-1,0,j+1}(t) + K_m^b(t-1-j)A_{m-1,0,j+1}(t) +$$
$$+ H_m(t)K_m^b(t-1-j)C_{m-1,j+1,0}(t)$$

$$A_{m,0,j}(t) = A_{m-1,0,j}(t) + \tag{2.10b}$$
$$+ H_m(t)C_{m-1,j,0}(t) + K_m^f(t)D_{m-1,0,j}(t) +$$
$$+ H_m(t)K_m^f(t-j)R_{m-1,0,j}(t)$$

3. SUMMARY OF FINAL ALGORITHMS

This section contains the summary of the joint process ladder algorithms developed in this paper. First we state the true LS joint process ladder algorithm. The derivation of the AR-recursions can be found in /7,11/. We use the following simplified notations for convenience

$$E_{m,0,j}(t) = E_j(t) \qquad A_{m,0,j}(t) = A_j(t)$$

$$R_{m,0,j}(t) = R_j(t) \qquad D_{m,0,j}(t) = D_j(t)$$

$$C_{m,0,j}(t) = C_j(t)$$

$$C_{m,j,0}(t) = C_j^*(t)$$

FOR j=0,1,...,p-1 initialize:

$$\begin{aligned}
&E_j(t) = W_{0,j}(t) && E_j(t-1) = W_{1,j}(t)\\
&R_j(t) = W_{1,j+1}(t) && R_j(t-1) = W_{2,j+1}(t)\\
&C_j(t) = W_{0,j+1}(t) && C_j(t-1) = W_{1,j+1}(t)\\
&C_j^*(t) = W_{1,j}(t) && C_j^*(t-1) = W_{2,j}(t)\\
&A_j(t) = v_j(t)\\
&D_j(t) = v_{j+1}(t)
\end{aligned}$$

$$K_1^f(t) = -C_0(t)/R_0(t) \qquad K_1^b(t) = -C_0(t)/E_0(t)$$

$$H_1(t) = -D_0(t)/R_0(t)$$

FOR m=1,2,...,p-1

FOR j=0,1,...,p-m-1

$$\begin{aligned}
E_j(t) &= E_j(t) + K_m^f(t)C_j^*(t) + K_m^f(t-j)C_j(t) +\\
&\quad + K_m^f(t)K_m^f(t-j)R_j(t)
\end{aligned}$$

$$\begin{aligned}
R_j(t) &= R_j(t-1) + K_m^b(t-1)C_j(t-1) + K_m^b(t-1-j)\\
&\quad C_j^*(t-1) + K_m^b(t-1)K_m^b(t-1-j)E_j(t-1)
\end{aligned}$$

$$\begin{aligned}
C_j(t) &= C_{j+1}(t) + K_m^f(t)R_{j+1}(t) + K_m^b(t-1-j)\\
&\quad E_{j+1}(t) + K_m^f(t)K_m^b(t-1-j)C_{j+1}^*(t)
\end{aligned}$$

$$\begin{aligned}
D_j(t) &= D_{j+1}(t) + H_m(t)R_{j+1}(t) + K_m^b(t-1-j)\\
&\quad A_{j+1}(t) + H_m(t)K_m^b(t-1-j)C_{j+1}^*(t)
\end{aligned}$$

$$\begin{aligned}
A_j(t) &= A_j(t) + H_m(t)C_j^*(t) + K_m^f(t-j)D_j(t) +\\
&\quad + H_m(t)K_m^f(t-j)R_j(t)
\end{aligned}$$

$$C_0^*(t) = C_0(t)$$

FOR j=0,1,...,p-m-2

$$\begin{aligned}
C_{j+1}^*(t) &= C_j^*(t-1) + K_m^f(t-1-j)R_j(t-1) +\\
&\quad + K_m^b(t-1)E_j(t-1) + K_m^f(t-1-j)\\
&\quad K_m^b(t-1)C_j(t-1)
\end{aligned}$$

$$K_{m+1}^f(t) = -C_0(t)/R_0(t) \qquad K_{m+1}^b(t) = -C_0(t)/E_0(t)$$

$$H_{m+1}(t) = -D_0(t)/R_0(t)$$

Table 3.1: True LS joint process ladder algorithm. $W_{i,j}(t)$ is the covariance matrix of the stimulation $\underline{x}(t)$. $v_j(t)$ is the cross covariance of $\underline{y}(t)$ and $\underline{x}(t)$.

It is seen that the true LS joint process ladder algorithm given in table 3.1 is efficient, requiring $O(p^2)$ computations per recursion. However, the implementation of this algorithm with the present days hardware can be troublesome, since the storage required will also grow with $O(p^2)$. The storage amount can be reduced to $O(p)$ if the input process $\underline{x}(t)$ is assumed to be wide sense stationary.This assumptionyields a Toeplitz structure of the GREs and results in two basic approximations which greatly facilitate the development of a low-storage joint process ladder algorithm. We have found, that the following stationarity assumptions do not seriously impair the performance of our joint process ladder algorithm.

APPROXIMATION 1 (constant reflection coefficients)

In case of a slowly time-varying stimulation $\underline{x}(t)$ the AR reflection coefficients can be assumed to be piecewise constant.

$$K_m^f(t-i) = K_m^f(t) \qquad 0 \leqslant i,j \leqslant p \qquad (3.1a)$$

$$K_m^b(t-j) = K_m^b(t) \qquad (3.1b)$$

APPROXIMATION 2 (Toeplitz structure of GREs)

In case of a stationary stimulation $\underline{x}(t)$ the GREs obtain a Toeplitz structure.

$$E_{m,i+1,j+1}(t) = E_{m,i,j}(t) \qquad (3.2a)$$

$$R_{m,i+1,j+1}(t) = R_{m,i,j}(t) \qquad (3.2b)$$

$$C_{m,i+1,j+1}(t) = C_{m,i,j}(t) \qquad (3.2c)$$

The incorporation of (3.1a,b) together with (3.2a,b,c) gives the following low-storage joint process ladder algorithm.

FOR j=0,1,...,p-1 initialize:

$$\begin{aligned}
&E_j = W_{0,j} && C_j = W_{0,j+1} && A_j = v_j\\
&R_j = W_{1,j+1} && C_j^* = W_{1,j} && D_j = v_{j+1}
\end{aligned}$$

$$K_1^f = -C_0/R_0 \qquad K_1^b = -C_0/E_0 \qquad H_1 = -D_0/R_0$$

FOR m=1,2,...,p-1

FOR j=0,1,...,p-m-1

$$\begin{aligned}
E_j &= E_j + K_m^f(C_j + C_j^*) + K_m^f K_m^f R_j\\
R_j &= R_j + K_m^b(C_j + C_j^*) + K_m^b K_m^b E_j\\
C_j &= C_{j+1} + K_m^f R_{j+1} + K_m^b E_{j+1} + K_m^f K_m^b C_{j+1}^*\\
D_j &= D_{j+1} + H_m R_{j+1} + K_m^b A_{j+1} + H_m K_m^b C_{j+1}^*\\
A_j &= A_j + H_m C_j^* + K_m^f D_j + H_m K_m^f R_j
\end{aligned}$$

$$C_0^* = C_0$$

FOR j=0,1,...,p-m-2

$$C_{j+1}^* = C_j^* + K_m^f R_j + K_m^b E_j + K_m^f K_m^b C_j$$

$$K_{m+1}^f = -C_0/R_0 \qquad K_{m+1}^b = -C_0/E_0 \qquad H_{m+1} = -D_0/R_0$$

Table 3.2: Approximate joint ladder algorithm. The time index has been omitted.

4. CONCLUSIONS

Two efficient and numerically robust joint process ladder algorithms have been presented and their most important application, namely, the identification of FIR models has been discussed. Similar to the AR-ladder algorithm introduced by a second paper /11/ time and order recursions have also been seperated in our new joint process ladder algorithms so as to facilitate

- mixed precision computations
- arbitrary recursive windowing /1,8/
- block processing computing the update recursions only each M-th time step. (M = parameter update rate)

The problem in our new approach focusses on the derivation of the pure order recursive true LS joint ladder recursions initialized from the input/output cross covariance vector. This derivation succeeded by two additional GREs incorporated in our formalism. Moreover, a second low-storage algorithm with only a modest decrease in performance, compared to the true LS solution, has been developed by two straightforward approximations.

ACKNOWLEDGEMENT

The author wishes to thank W. Ptacek for his discussions and comments on the new joint process ladder algorithms.

REFERENCES

/1/ T.P. Barnwell, "Recursive windowing for generating autocorrelation coefficients for LPC analysis", IEEE Trans. on ASSP, ASSP-29(5), pp. 1062-1066, 1981.

/2/ D.T.L. Lee, "Canonical ladder form realizations and fast estimation algorithms", Ph.D. dissertation, Stanford Univ., Stanford, CA, 1980.

/3/ S.L. Marple, "Efficient least squares FIR system identification", IEEE Trans. on ASSP, ASSP-29(1), pp. 62-73, 1981.

/4/ S.L. Marple, "Fast algorithms for linear prediction and system identification filters with linear phase", IEEE Trans. on ASSP, ASSP-30(6), pp. 942-953, 1982.

/5/ B. Porat and T. Kailath, "Normalized lattice algorithms for least-squares FIR system identification", IEEE Trans. on ASSP, ASSP-31(1), pp. 122-128, 1983.

/6/ P. Strobach, "Schnelle adaptive Algorithmen zur ordnungsrekursiven Kleinste-Quadrate-Schätzung autoregressiver Parameter", Ph.D. dissertation, Bundeswehr University Munich, Munich, West Germany, 1985.

/7/ P. Strobach, "Pure order recursive least squares ladder algorithms", IEEE Trans. on ASSP, August issue, 1986.

/8/ P. Strobach, "Efficient covariance ladder algorithms for finite arithmetic applications", submitted to SIGNAL PROCESSING, 1986.

/9/ P. Strobach, "Robust least squares covariance ladder algorithms for vector autoregressive processes", IEEE Trans. on Automatic Control, under review, 1986.

/10/ P. Strobach, "New forms of least squares lattice algorithms and a comparison of their round-off error characteristics", Proc. Int. Conf. on ASSP, Tokyo, 1986.

/11/ P. Strobach, "Efficient and robust covariance ladder algorithms for linear prediction", this volume, 1986.

/12/ P. Strobach, "A new class of least squares covariance ladder estimation algorithms", Proc. 19th Annual Asilomar Conf. on Circuits, Systems and Computers, Montery, CA, 1986.

/13/ P. Strobach, "A new approach to the least squares ladder estimation algorithm", Proc. IASTED Int. Symposium on Applied Signal Processing and Digital Filtering, pp. 37-40, Paris, 1985.

/14/ P. Strobach and U. Appel, "Ein Signalprozessor mit Wirt-Gast-Kopplung über gemeinsame Speicherbereiche", Proc. 8. GI-NTG Fachtagung Architektur und Betrieb von Rechensystemen, Springer-Verlag, Karlsruhe, pp. 61-72, 1984.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N5.2

VALIDATION OF MEASUREMENTS AND DETECTION OF SENSORS' FAILURES IN CONTROL SYSTEMS

M. STAROSWIECKI, M. HAMAD

Centre d'Automatique de Lille
Université des Sciences et Techniques de Lille Flandres-Artois
Bâtiment P2
59655 Villeneuve d'Ascq Cédex

The minivar residual approach is used to generate statical and dynamical relationships applying to the measured variables of a control system. Using this approach a degree of coherence is associated to each sensor and the signature which corresponds to its failure is defined. Multiple failures are also considered.

Keywords : Measurement, sensor validation, failure detection, analytical redundancy.

1. INTRODUCTION

In the hierarchical structure of a control system, orders are elaborated at each level from informations which are provided from lower ones. At the physical level (i.e the controled process) informations are issued from measurement devices including sensors and processing systems.

Faulty instruments lead to generate false informations which are used in upper levels. Thus these levels elaborate orders and statements which do not correspond to the real state of the process. Accordingly, it is necessary to validate these informations before their treatement at different hierarchical levels |1|.

The validation procedure consists to check the coherence between measurements themselves |2|, |3|, and between measurements and real values corresponding to a given operation point. Such a procedure is based upon the existence of either hardware or analytical redundancy. Analytical redundancy uses the system's model in order to generate statical or dynamical relationships linking the values of the measured variables. These relationships generate residuals whose supervision allows the validation of the informations issued from the physical level through the sensors.

In this communication, we present an approach which generates residuals with minimal variance. The coherence degree of a variable is defined and used to localize failures by means of a signature which is independant of the kind of the failure.

2. RESIDUAL GENERATION

Let us consider a control system described by the following invariant model :

$$x(k+1) = A\,x(k) + B\,u(k) + \eta(k) \qquad (1)$$

$$y(k) = C\,x(k) + \varepsilon(k) \qquad (2)$$

(1) is the state equation ; (2) is the measurement equation with the following notations : x, y and u are state, measurement and control vectors. n, m and q are their dimensions. A, B, C are matrices with corresponding dimensions.

2.1. Parity space approach

The parity space approach |4| is based only on the model of the sensors provided by the measurement equation (2). It exploits the excess information due to the physical redundancy ($m > n$) to elaborate relatinships among the outputs of the sensors. The vector defined by the residuals of these relationships is called the parity vector.

$$\forall k \qquad p(k) = \Omega\,y(k) = \Omega\,\varepsilon(k)$$

$$(\Omega \text{ such that } \Omega C = 0) \qquad (3)$$

This method cannot be applied in the absence of physical redundancy ($m \leq n$).

The generalized parity space approach |5| creates excess information via the use of the analytical redundancy relationships which exists in the system model (1) and (2). Considering an observation window of length l, one obtains :

$$\begin{vmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+l) \end{vmatrix} = \begin{vmatrix} C \\ CA \\ \vdots \\ CA^{l} \end{vmatrix} x(k) + \begin{vmatrix} 0 & & & 0 \\ CB & & & \\ \vdots & & & \\ CA^{l-1}B & \cdots & CB & 0 \end{vmatrix} \begin{vmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+l) \end{vmatrix}$$

$$+ \begin{vmatrix} 0 & & & 0 \\ C & & & \\ \vdots & & & \\ CA^{l-1} & \cdots & C & 0 \end{vmatrix} \begin{vmatrix} \eta(k) \\ \eta(k+1) \\ \vdots \\ \eta(k+l) \end{vmatrix} + \begin{vmatrix} \varepsilon(k) \\ \varepsilon(k+1) \\ \vdots \\ \varepsilon(k+l) \end{vmatrix} \qquad (4)$$

which is written under condensed form :

$$Y(k,l) = O_l x(k) + C_l u(k,l) + D_l \eta(k,l) + \varepsilon(k,l) \qquad (5)$$

The elimination of the state vector leads to the generalized parity vector, defined by :

$$p(k,l) = \Omega Y(k,l) - \Omega C_l U(k,l) = \Omega D_l \eta(k,l) + \varepsilon(k,l) \qquad (6)$$

(Ω is such that $\Omega O_l = 0$)

The failure detection problem consists to check whether the parity vector, or the generalized one, results only from noises as indicated by equation (3) or (6) or does not. For that, each of its components is compared to a given threshold whose value results from the solution of a decision problem in which H_o is the no failure hypothesis while H_1 is the failure hypothesis [6]. It is clear that the performance of the optimal decision is decreasing with the variance of the parity vector in the H_o case.

2.2. Minivar residual approach

If we assume the system is stable at known operating point, it is possible to generate a set of redundancy relationships whose residuals have minimal variance. We call this method Minivar Residual Approach MVRA. The equations (1) and (2) considered in the neighbourhood of the operating point (x^*,u) become :

$$X(k+1) = AX(k) + \eta(k) \qquad (7)$$

$$Y(k) = CX(k) + \varepsilon(k) \qquad (8)$$

$X(k) = x(k) - x^*(k)$, $\eta(k)$, $\varepsilon(k)$ are independant, Zero mean, random vectors, with covariance matrices Σ, Q and R.
$Y(k) = y(k) - Cx^*(k)$ is then a random vector Zero mean and covariance matrix $V = C\Sigma C' + R$.

The task of the MVRA is to transform the deviation vector (difference between measurement and theoretical vectors) into a new vector which is called minivar residual vector. This vector is obtained by solving a constrained quadratic minimisation problem.

a. Statical relationships

Consider the following parity vector :

$$P(k) = Y(k) = CX(k) + \varepsilon(k) \qquad (9)$$

P(k) is a random vector with Zero mean and covariance matrix V. Any linear combination of the components of P(k) can be used as a residual for the decision problem. We shall look for these which have minimal variance.

b. Dynamical relationships

The observation window of length l, applied to equations (7), (8) gives :

$$Y(k,l) = O_l X(k) + D_l \eta(k,l) + \varepsilon(k,l) \qquad (10)$$

The parity vector $P(k,l) = Y(k,l)$ has Zero mean and covariance matrix $V_l = O_l \Sigma O_l' + D_l \bar{Q} D_l' + \bar{R}$

with $\bar{Q} = \text{diag}\{Q,\ldots,Q\}$, $\bar{R} = \text{diag}\{R,\ldots,R\}$

We shall in this case apply the same approach than in the statical one.

c. Determination of the residuals

Let $r_i(k,l) = u_i' P(k,l)$ be the i^{th} generated residual. In both cases (statical or dynamical) the optimal residual generation problem is formulated as follows :

Find w_i $\quad i=1,\ldots,5$ such that :

$w_i' V_l w_i$ minimum

under the constraints $\quad w_i' w_i = 1$

$w_i' w_j = 0 \quad j \neq i$

This is a principal components analysis problem whose solution leads to retain the σ eigenvectors of the matrix V_l which correspond to the σ smallert eigenvalues [7]. The variance of the i^{th} residual is then equal to λ_i the i^{th} eigenvalue) and the number σ of the retained residuals is such that λ_σ has still an acceptable value. Let $R(k,l) = WY(k,l)$ be the vector of the r_i' s.

It is possible to show that when x^* is the least square estimate of x, the parity space (or generalized parity space) is a subspace of the minivar residual space (or generalized minivar residual space).

In the absence of failure, the residual vector is a random vector with known mean and covariance. When a failure occurs, these characteristics of the residual vector change.

The failure detection problem consists to decide whether the minivar residual vector lies in the noise area or not. The isolation

problem leads to the recognition of the failed sensor, and the diagnosis problem leads to the recognition of the kind of failure.

3. FAILURE DETECTION

As in the parity space approach |4|, |5|, the columns of the matrix W define m failure directions in the statical case (or m failure subspaces in the dynamical case) corresponding to the m sensors in the minivar residual space.

The use of a voting scheme supposes the 2 following conditions : detection condition : each sensor is included in at least one relationship ; isolation condition : $\forall (y_i, y_j)$ the set of the relationships in which y_i intervenes is not identical to the set of the relationships in which y_j intervenes.

3.1. Degree of coherence of a measurement

Let us consider a given sensor "y_j" and let "σ_j" be the number of relationships in which "y_j" intervenes. As we perform tests concerning all the relations, the set of the σ_j relationships including "y_j" will be split into two parts :

- The set of those whose residual is less than the decision threshold. Let σ_{jc} be their number.
- The set of those whose residual is greater than the decision threshold: their number is $\sigma_j - \sigma_{jc}$.

The *degree of coherence* of the measurement y_j is then defined by :

$$dj = \frac{\sigma_{jc}}{\sigma_j} \qquad (11)$$

3.2. Failure signature of the sensor

Let us suppose that a failure occurs on the i^{th} sensor. The degree of coherence of the measurements is given by :

$$j = 1,2,\ldots,m \quad d_j(i) = \begin{cases} 0 & \text{if } j = i \\ 1 - \dfrac{\sigma_{ji}}{\sigma_j} & \text{if } j \neq i \end{cases} \qquad (12)$$

where σ_{ji} is the number of relationships in which both y_j and y_i intervene.

The vector $D_i = (d_j(i),\ j = 1,\ldots m)$ is called *failure signature of the sensor* y_i.

It appears clearly that when no sensor is failed, all the degrees of coherence should be equal to "1". This is the *no-failure signature*.

Algorithm : For the sake of brievity, the idea of the algorithm is presented under the hypothesis that there does not exist any decision error (i.e r_i < threshold → no failed instrument intervene in the i^{th} relationship, $r_i >$ threshold → some failed instrument in the i^{th} relationship). At each instant, the degrees of coherence are comported for all the sensors. The resulting signature is compared to the typical ones : no failure, y_1 failed, y_2 failed, etc...

It is possible to show that, in this ideal case, the maximal number of sensors which can be supervised by the use of σ residuals is equal to $2^{\sigma}-1$.

3.3. Multiple failures

The case in which multiple failures occur simultaneously rises two kinds of problems :

. multiple failures may cancel each other so that we obtain the no failure signature (invisibility case)

. multiple failures can be detected, but exhibit a signature that is the same than an existing single failure signature.

The first case leads to a non detect of the failure, which the second case leads to a false isolation of the failed instrument.

The conditions under which no invisibility case can exist are linked with the independance of the columus of the matrix W.

4. CONCLUSION

The MVRA is used to generate statical and dynamical relationships applying to the measured variables of a control sytem. The degree of coherence of each measurement is computed on the basis of the result of a decision procedure applied to each of the relationships. The choice of the best relations is not a trivial one. In fact they should have :

- a small variance, in order to obtain a good decision threshold
- a short observation window, in order to accelerate the failure detection
- a weak effect of the failure of one sensor on the degrees of coherence of the others (see (11)).

REFERENCES

|1| Tylee, J.L., Hon, A.L.,"New concepts in Nuclear Power Plant Instrumentation and Control" 22th IEEE CDC, San Antonio, Texas, December 1983.

|2| Desai, M.N., Ray, A., "A Fault Detection and Isolation Methodology" 20th IEEE CDC, San Diego, California, December 1981, pp 1363-1369.

|3| Deckert, J.C., and all, "A signal Validation Methodologie for Nuclear power plants" presented at American Control Conference, Arkington, Virginia, June, 1982.

|4| Potter, J.E., Suman, M.C., "Thresholdless Redundancy Management with Array of Skewed Instruments" Integrity in Electronic Flight Control Systems, AGARDOGRAPH - 224, pp 15-1 to 15-25, 1977.

|5| Chow, E.Y., Willsky, A.S., "Analytical Redundancy and the Design of Robust Failure Detection Systems" IEEE Trans. on AC, Vol. AC-29, N° 7, July 1984.

|6| Arques, P.Y., "Décisions en Traitement du Signal" Masson, 1979.

|7| Seber, G.A.F., "Multivariate Observations" J. WILY and Sons, 1984.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AUTOREGRESSIVE ANALYSIS OF DIGITALLY SIMULATED NUCLEAR REACTOR NOISE

J.E. Hoogenboom, Ö. Ciftcioglu* and H. van Dam

Delft University of Technology and Interuniversity Reactor Institute,
Mekelweg 15
2629 JB Delft, The Netherlands.

A digital nuclear reactor simulator with software noise generators has been used to test the multivariate autoregressive (AR) modelling of noise signals. Some peculiarities of digital simulation in connection with AR-analysis are pointed out. The importance of a correct choice of the sampling time for application of AR analysis is demonstrated. Also the role of noise contribution ratio's in system identification is shown.
With appropriate measurement conditions correct results for system transfer functions and other quantities in comparison with their theoretical expectations can be obtained.

1. INTRODUCTION

Noise as seemingly undetermined fluctuations in measured signals is a well-known phenomenon in every physical system. Most often it is considered as a nuisance, which must be eliminated as far as possible to extract "true" information from the signals. However, if the noise is not introduced by the measurement equipment like sensors, transducers, amplifiers, etc., the noise in the system variables is influenced by the system dynamics and opens the possibility to extract information about the system dynamic behaviour.
Although system dynamics can efficiently be studied by exciting the system by well-chosen input signals, this may not always be possible in practice when the system is not available for experiments. This is mostly the case with nuclear power plants for economic reasons. Then it is very efficient to make use of the noise inherently present in the system under operating conditions without disturbing the system. Moreover, noise measurements with appropriate analysis methods can be very powerful and sensitive for detection of small changes in the signal characteristics, yielding noise analysis as a widespread method to analyze and monitor nuclear power plants.
If a reactor is considered as a system responding to one or more input signals, its transfer functions can be determined over the desired frequency band by measuring the frequency spectra of the input and output signals and their cross spectra. However, in noise studies on nuclear reactors the input signals are the physical noise sources present in the reactor and, in general, none of them can be measured directly. For such cases more advanced analysis methods have to be applied such as autoregressive modelling.

2. AUTOREGRESSIVE MODELLING

A noise signal can be described as a regression of its values at previous times and the addition of a white noise source. Following the description of Kleiss [1], the autoregressive model for the multivariate case with m signals can be written as follows

$$\underline{x}_t = \sum_{i=1}^{p} A_i \underline{x}_{t-i} + \underline{n}_t \qquad (1)$$

with $\underline{x}_t$ the signal vector at time t, $\underline{n}_t$ the vector with inherent noise sources, A_i mxn matrices comprising the AR-model parameters and p the model order. Time t is measured in units of the sampling time Δ. The matrices A_i and the covariance matrix $\Sigma = \langle \underline{n}_t \underline{n}_t^T \rangle$ are estimated from the matrix of cross correlation functions C_k using the multivariate Yule-Walker equations, which can be derived from Eq.(1)

$$C_k^T = \langle \underline{x}_t \underline{x}_{t+k}^T \rangle = \sum_{i=1}^{p} A_i C_{k-i}^T + \Sigma \delta_{k,0} \qquad (2)$$

*on leave from Technical University of Istanbul, Turkey.

using the recursive procedure for increasing model orders, described by Upadhyaya et al [2]. If we define the matrix G(f) of transfer functions from noise source vector $\underline{N}$ in the frequency domain to the signal vector $\underline{X}(f)$ by $\underline{X}(f)=G(f)\underline{N}(f)$, we obtain from Eq.(1) by discrete Fourier transformation

$$G^{-1}(f) = I - \sum_{i=1}^{p} A_i e^{-2\pi j f i \Delta} \qquad (3)$$

with I the unit matrix. The frequency spectra are obtained from

$$S(f) = \langle \underline{X}^*(f)\underline{X}^T(f) \rangle = G^*(f)\Sigma G^T(f) \qquad (4)$$

If the noise sources in the system are uncorrelated we have for the autospectrum of signal i

$$S_{ii}(f) = \sum_{k=1}^{m} G^*_{ik}(f)\Sigma_{kk}G_{ik}(f) \qquad (5)$$

Each term in the summation of the RHS is the contribution to the autospectrum of signal i from noise source k. Then the noise contribution ratio (NCR) is defined as

$$NCR_{ik}(f) = G^*_{ik}(f)G_{ik}(f)\Sigma_{kk}/S_{ii}(f) \qquad (6)$$

It is also possible [1] to derive the open-loop transfer functions $H_{ij}(f)$ between signals i and j. The noise contribution ratio's are a helpful means for system identification. If a NCR is very small over a certain frequency range, the accompanying open-loop transfer function cannot be determined well for that frequency range.

3. NUCLEAR REACTOR SIMULATOR

As nuclear reactors are complex dynamical systems with several feedback loops, there is a lack of experimental verification of the results obtained from an AR-analysis. Therefore, a digital simulation of a simplified nuclear reactor system was used to test the AR-analysis. The simulator model contains the kinetics from neutron multiplication by fission chains for the simple case of a position independent description (point reactor kinetics). The model allows for the description of six groups of delayed neutrons, which are not directly released in a fission, but by decay of certain fission product nuclei formed at a fission. The kinetic equations for the number of neutrons n(t) and the delayed neutron precursors $C_i(t)$ of the i-th group read

$$\frac{dn}{dt} = \frac{\rho(t)-\beta}{\Lambda} n(t) + \sum_{i=1}^{6} \lambda_i C_i(t)$$

$$\frac{dC_i}{dt} = \frac{\beta_i}{\Lambda} n(t) - \lambda_i C_i(t) \qquad i=1,\ldots,6$$

with all symbols having their usual meaning [3]. Note that the equations are not linear with respect to the reactivity $\rho(t)$ which is a measure for the departure from equilibrium of the fission chain reactions in the reactor. For theoretical calculations of transfer functions for noise signals the equations have to be linearized.

In addition to the reactor kinetics equations feedback loops are added describing the effects of temperature of fuel and coolant on reactivity. All together the simulator is based on 10 coupled algebraic or first order differential equations.

In the present study three output signals are considered: reactor power P, which is proportional to the number of neutrons n in the reactor, the fuel temperature T_f and the coolant temperature T_c. Input signals are three noise sources chosen in accordance with realistic reactor systems, namely reactivity noise due to control rod vibrations, noise in the inlet temperature of the coolant and noise in the coolant flow velocity, acting on the fuel temperature. The noise sources were realized using independent pseudo random number generators, providing white noise and applying appropriate digital filters to obtain the desired frequency spectra.

The processes are simulated in real time using a computer clock and analog signals for the selected output variables are available from a digital to analog converter, which makes it possible to apply the usual methods of analog noise analyses.

4. DIGITAL FILTERING

In the simulated system, various digital low-pass filters were implemented as result of physical considerations. These are essentially used to shape relevant white noise sources for the simulation of appropriate driving noise sources exciting the system as input signals. The detailed description of the filters is presented elsewhere [4] and is only briefly summarized here:

- first-order AR filtering for the simulation of fluctuations in the coolant flow,
- second-order filtering with bilinear transformation for the simulation of a vibrating control rod with damping.
- fifth-order Chebyshev filtering for the simulation of the inlet temperature fluctuations assumed to be band-limited white.

In order to investigate the effect of the filtering technique used in AR analysis, a transversal low-pass filter was designed and implemented as an alternative to the Chebyshev filter. To obtain comparable frequency characteristics, the number of weighting coefficients

was chosen to be 32. Since the filter is in moving average form, approximation of such a system with AR modelling requires relatively high model orders. Although exact comparison is rather difficult to carry out due to statistical variations, for this particular case an appreciable increase in model order relative to that obtained with Chebyshev filtering was observed.

5. FEATURES OF DIGITAL SIMULATION FROM THE VIEW-POINT OF AR ANALYSIS

For an AR model it follows from Eq.(1) that between any pair of input and output signals there is a time delay which is at least equal to the sampling time Δ. The delay introduced into the model is justified by virtue of the causality condition in physical systems. In real systems due to its continuous time character, there is always a certain elapse of time necessary for information transmission between input and output signals so that the completion of the total signal transmission can better be expressed in terms of time constants.
From the view-point of AR analysis of data obtained from a digitally simulated system the following observations are essential.

- As the discrete representation of a differential equation describing a (sub)system normally provides instantaneous response, a delay has to be introduced deliberately in the algorithm. At least a delay of one sampling time is necessary between any pair of signals in a system.
- In the digital simulation of a closed-loop system a delay will naturally takes place in the algorithm, because the feedback variable cannot be determined when it is to be added to the input signal of the loop. This delay can be used as a substitute for the deliberate delay mentioned above if one of the measured signals is obtained from the feedback loop.
- If there are several measured signals in a closed-loop, there is a certain freedom where to insert the necessary delays. However, their position in the loop must be chosen carefully in order to meet all requirements without introducing unnecessary delays.

The implementation of the above mentioned observations is illustrated in the example shown in Fig. 1 where N is the driving noise source, X_1 and X_2 signals used in AR analysis and z^{-1} a delay element.

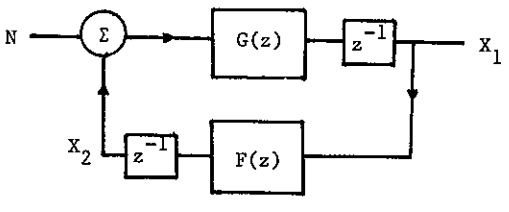


Fig. 1 Schematic representation of a digitally simulated linear system with feedback

The investigations revealed that, if the delays are not placed correctly in the simulation, a deterioration of system identification results as the noise source covariance matrix does not take the diagonal form, indicating false correlations between the sources.
Although there is no general rule for the selection of optimum sampling time for AR analysis of analog data, it can be stated that for the analog data obtained from a digitally simulated system, the sampling time should be less than or equal to the time step used in the digital computations so that instantaneous information transmission is avoided.

6. RESULTS OF AR ANALYSIS

The cumulative noise contribution ratio's (NCR's) of the coolant temperature noise signal are shown in Fig.2 for a sampling time Δ=10 ms. As can be seen from this figure the functions agree well with the theoretical values calculated from the known system parameters. The NCR's for the three noise sources do not add up to 1 exactly for low frequencies as a result of some correlation between the noise sources found in the AR analysis. This may also be the reason for the deviations of the frequency spectrum of the coolant temperature fluctuations from the theoretical computations as shown in Fig. 3.

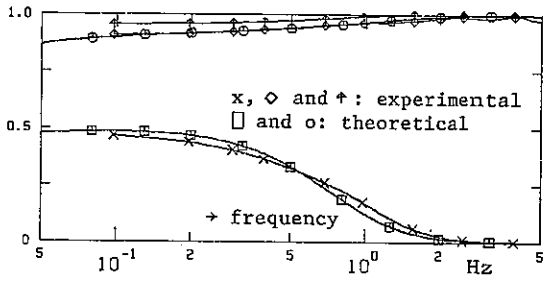


Fig. 2. Experimental and theoretical cumulative NCR's for coolant temperature signal

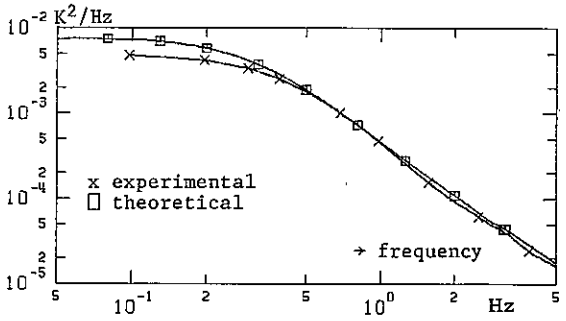


Fig.3. Experimental and theoretical coolant temperature fluctuation frequency spectra

Fig. 4 shows the open-loop transfer function from reactor power to fuel temperature fluctuations, which compares well with theory. Fig. 5 shows the open-loop transfer function from coolant temperature to fuel temperature fluctuations. Due to the very low NCR from the intrinsic reactor power noise source to the fuel temperature signal shown in Fig. 6 (between lines denoted by O and Δ), this transfer function cannot be obtained accurately.
Fig. 6 also shows the sum of the NCR's for the fuel temperature signal measured with 6, 20 and 40 ms sampling time. The basic time step in the digital simulation was 5 ms. This figure clearly demonstrates that longer the sampling time stronger the correlation found between the noise sources from AR analysis.

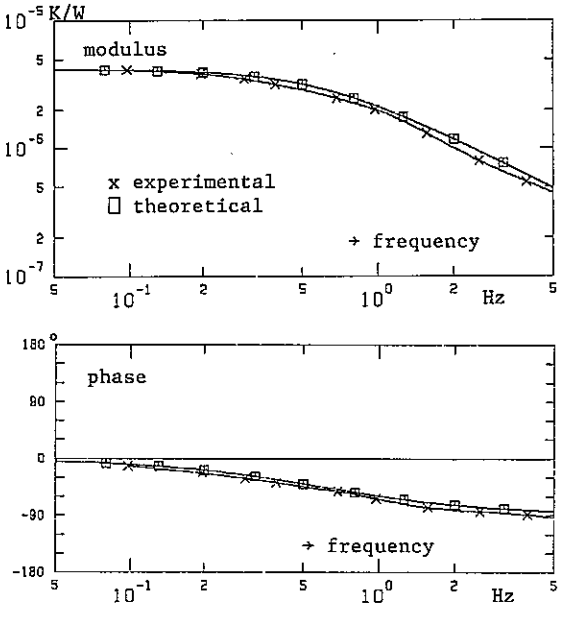


Fig. 4. Experimental and theoretical power-to-fuel temperature open-loop transfer functions

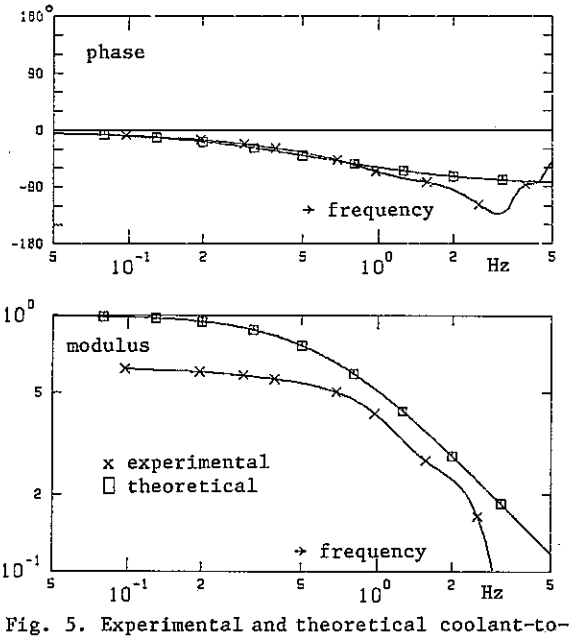


Fig. 5. Experimental and theoretical coolant-to-fuel temperature open-loop transfer functions

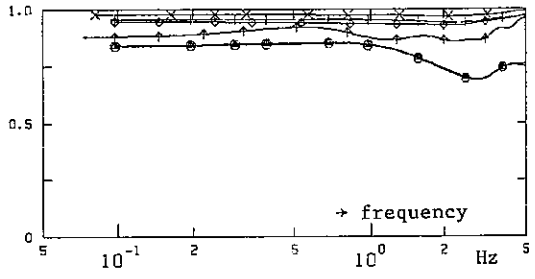


Fig. 6. Cumulative NCR's (symbols o, Δ and +) and sum of NCR's for different sampling times (x: 6 ms; +: 10 ms; ◇ : 20 ms and ↑ : 40 ms)

7. CONCLUSIONS

Correct multivariate autoregressive model of a system can be determined from its noise output signals if the sampling time is chosen short enough. If the sampling time is not chosen correctly, correlation between the inherent noise sources of the several signals will be found, which need not be the case in reality. AR-modelling provides correct open-loop transfer functions between two measured signals if the inherent noise source for the first signal contributes significantly to the spectrum of the second signal. The NCR is therefore a very helpful means to judge system identifiability. If AR-analysis is to be applied to signals generated by digital simulation care must be taken in designing the digital simulator and delays between every pair of measured signals should be included.

REFERENCES

[1] Kleiss, E.B.J., On the Determination of Boiling Water Reactor Characteristics by Noise Analysis, Thesis Delft University of Technology (Delftse Universitaire Pers, Delft, 1983).
[2] Upadhyaya, B.R., Kitamura, M. and Kerlin, T.W., Ann. Nucl. En. 7 (1980) 1.
[3] Lewins, J., Nuclear Reactor Kinetics and Control (Pergamon Press, Oxford, 1978).
[4] Ciftcioglu, Ö., Synthesizing Noise-like Data Through Simulation, IRI-131-85-06 (Interuniversitair Reactor Instituut, Delft, 1985).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

GENERATION OF THE FAST 2D DISCRETE FOURIER TRANSFORM

K. Yeung, O. Rath and K. R. Rao

Department of Electrical Engineering
The University of Texas at Arlington
Arlington, Texas 76019

An algorithm is developed for the generation of the radix-2 2D-Discrete Fourier Transform (DFT). The computational complexity is shown and compared with the repeated use of the 1D-DFT algorithm. The algorithm has been tested by Fortran programs.

INTRODUCTION

Since the mid sixties, when the fast Fourier transform (FFT) was discovered by Cooley and Tukey [1] a lot of FFT algorithms have been developed. During that period particular interest has been evinced on the development of algorithms for the 1D-case only.

The presentation here is not an extension of the fast 1D-case. The computational complexity is shown to be less than what it would be if the 2D-DFT were implemented by repeated use of the fast 1D-algorithm.

The cue for the development of this algorithm has been taken from Besslich [2-3] wherein he develops the 2D-(WHT_h) (Hadamard order Walsh Hadamard transform) of the data hierarchically by arranging the 2D-data matrix of order say NxN into a vector of the size N^2x1 and treating it like the 1D-case of order N^2.

THE 2D-WHT_h

The 2D-(WHT_h) for 2D data can be written as

$$[Q(n)] = [H_h(n)][P(n)][H_h(n)]^T \qquad (1)$$

where,
$[Q(n)]$ = the transformed data matrix
$[P(n)]$ = the original 2D data matrix
$[H_h(n)]$ = the (WHT_h) matrix

The dimension of the matrices are NxN where $N=2^n$ and the normalizing factor viz. $1/N^2$ has been dropped in (1).

Besslich has shown that if

$$[P(n)] = \begin{bmatrix} [P_0(n-1)] & [P_0(n-1)] \\ [P_2(n-1)] & [P_3(n-1)] \end{bmatrix}$$

and $[Q_i(n-1)]$ is the 2D-(WHT_h) of the corresponding $[P_i(n-1)]$ then $[Q(n)]$ can be written as,

$$[Q(n)] = \begin{bmatrix} [Q_0(n-1)]+[Q_1(n-1)]+[Q_2(n-1)]+[Q_3(n-1)] & [Q_0(n-1)]-[Q_1(n-1)]+[Q_2(n-1)]-[Q_3(n-1)] \\ [Q_0(n-1)]+[Q_1(n-1)]-[Q_2(n-1)]-[Q_3(n-1)] & [Q_0(n-1)]-[Q_1(n-1)]-[Q_2(n-1)]+[Q_3(n-1)] \end{bmatrix} \qquad (2)$$

STAGGERED (WHT_h)

We digress here to discuss a method [2] to obtain the (WHT_h) of a 2D-block iteratively. The transform coefficients of each 2x2 sub-block are calculated first. From there we calculate the (WHT_h) of the 4x4 subblocks and so on until we have calculated the (WHT_h) of the whole block of size NxN. In order to be able to relate (2) with the above discussion let us define a few notations;
Suppose for a matrix,

$$[A(1)] = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

we define a companion vector $\underline{a}(2) = [a_{11}a_{12}a_{21}a_{22}]^T$ and similarly for any matrix of the dimension NxN,

$$[A(n)] = \begin{bmatrix} [B_0(n-1)] & [B_1(n-1)] \\ [B_2(n-1)] & [B_3(n-1)] \end{bmatrix}$$

the companion vector is defined as

$$\underline{a}(2n) = [\underline{b}_0^T(2(n-1))\ \underline{b}_1^T(2(n-1))\ \underline{b}_2^T(2(n-1))\ \underline{b}_3^T(2(n-1))]^T$$

where the notations are defined as follows: if [A(n)] is a matrix of the order NxN then its companion vector is represented by the lower case 'a' and the number '2n' within the parenthesis represents the number of elements it contains viz. 2^{2n}.

As an example the formation of a companion vector of a matrix [A(3)] of the order 8x8 has been shown by Fig. 2.1 from which we get the companion vector for [A(3)] as:

$[a_{11},a_{12},a_{21},a_{22},a_{13},a_{14},a_{23},a_{24},a_{31},a_{32},a_{41},$
$a_{42},a_{33},a_{34},a_{43},a_{44},a_{15},a_{16},a_{25},a_{26},a_{17},a_{18},$
$a_{27},a_{28},a_{35},a_{36},a_{45},a_{46},a_{37},a_{38},a_{47},a_{48},a_{51},$
$a_{52},a_{61},a_{62},a_{53},a_{54},a_{63},a_{64},a_{71},a_{72},a_{81},a_{82},$
$a_{73},a_{74},a_{83},a_{84},a_{55},a_{56},a_{65},a_{66},a_{57},a_{58},a_{67},$
$a_{68},a_{75},a_{76},a_{85},a_{86},a_{77},a_{78},a_{87},a_{88}]$

Defining such companion vectors for [Q(n)], [Q´(n-1)] and [Q(n-1)],(2) can be written as $q(2n) = q_0'(2n-2)^T\ q_1'(2n-2)^T q_2'(2n-2)^T\ q_3'(2n-2)^T]^T =$

$$\begin{bmatrix} [I(2n-2)] & [I(2n-2)] & [I(2n-2)] & [I(2n-2)] \\ [I(2n-2)] & -[I(2n-2)] & [I(2n-2)] & -[I(2n-2)] \\ [I(2n-2)] & [I(2n-2)] & -[I(2n-2)] & -[I(2n-2)] \\ [I(2n-2)] & -[I(2n-2)] & -[I(2n-2)] & [I(2n-2)] \end{bmatrix} \cdot$$

$[q_0'(2n-2)^T\ q_1'(2n-2)^T\ q_2'(2n-2)^T\ q_3'(2n-2)^T]^T$

which implies,

$$q(2n) = [[H_h(2)] \times [I(2(n-1))]] \cdot [q_0(2(n-1))^T\ q_1(2(n-1))^T\ q_2(2(n-1))^T\ q_3(2(n-1))^T]^T \quad (3)$$

Equation (3) can be used to compute the 2D-(WHT_h) of a block if its four subblocks are available without having to go through the 'Inverse (WHT_h)' of the subblocks.

$$[A(3)] = \left[\begin{array}{cc|cc|cccc} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & a_{18} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{27} & a_{28} \\ \hline a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} & a_{37} & a_{38} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} & a_{47} & a_{48} \\ \hline a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} & a_{57} & a_{58} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} & a_{67} & a_{68} \\ a_{71} & a_{72} & a_{73} & a_{74} & a_{75} & a_{76} & a_{77} & a_{78} \\ a_{81} & a_{82} & a_{83} & a_{84} & a_{85} & a_{86} & a_{87} & a_{88} \end{array}\right]$$

Fig. 2.1 The systematic formation of the companion vector of an 8x8 matrix [A(3)]

Expanding (3) we get,

$$q(2n) = [[H_h(2)] \otimes [I(2n-2)]] \cdot (\mathrm{diag}[[H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)]]) \ldots (\mathrm{diag}[[H_h(2)], [H_h(2)] \ldots [H_h(2)]])\ p(2n) \quad (4)$$

where $p(2n) = [p_{11}p_{12}p_{21}p_{22}p_{13}p_{14}p_{23} \cdots p_{NN}]^T$ is the companion vector of the data matrix [P(n)].

NUMBER OF ADDITIONS

Analyzing (4) we get the number of additions as $\alpha_{recursive} = 2N^2 \log_2 N$ where 'α' stands for the number of additions. This is the same number which we would have got by repeated use of the 1D-(WHT_h) formula.

THE 2D-DFT

The 2D-DFT can be written as $[X(n)] = [F(n)][P(n)][F(n)]^T$ (5)

where,
[X(n)] = the transformed data matrix
[P(n)] = the original 2D-data
$[F(n)] = [\omega_N^{uj}]$ u,j = 0,1,2,...,N-1
$\omega_N = \exp\{-2\pi i/N\}$, $i = \sqrt{-1}$

Note:
(i) Here also like the (WHT_h) case the normalizing factor $1/N^2$ has been neglected.
(ii) $[F(n)]=[F(n)]^T$ from the properties of the DFT. If we rewrite

$[F(n)] = [I(n)]_{BRO}[F(n)]_{BRO}$

where,
$[I(n)]_{BRO}$ = rows (columns) of [I(n)] in bit reversed order
$[F(n)]_{BRO}$ = the [F(n)] matrix with the rows in bit reversed order

then (5) can be written as,

$$[X(n)] = [F(n)]^T_{BRO}[I(n)]^T_{BRO}[P(n)][F(n)]_{BRO} = [F(n)]^T_{BRO}[\tilde{P}(n)][F(n)]_{BRO} \quad (6)$$

where

$[\tilde{P}(n)] = [I(n)]^T_{BRO}[P(n)][I(n)]_{BRO} = [I(n)]_{BRO}[P(n)][I(n)]_{BRO}$

The recursive formula for the DFT is given by, $[F(n)]_{BRO} =$

$$\begin{bmatrix} [F(n-1)]_{BRO} & 0 \\ 0 & [F(n-1)]_{BRO} \end{bmatrix} \begin{bmatrix} [I(n-1)] & 0 \\ [0] & [W(n-1)] \end{bmatrix} \cdot \begin{bmatrix} [I(n-1)] & [I(n-1)] \\ [I(n-1)] & -[I(n-1)] \end{bmatrix} \quad (7)$$

where $[W(n-1)] = (\text{diag}[1 \quad w_N \quad w_N^2 \cdots w_N^{N/2-1}])$;
$w_N = \exp\{-2\pi i/N\}$

Substituting the recursive DFT formula in (6) we get,
$[X(n)] =$

$$\begin{bmatrix} [I(n-1)] & [I(n-1)] \\ [I(n-1)] & -[I(n-1)] \end{bmatrix} \begin{bmatrix} [I(n-1)] & 0 \\ 0 & [W(n-1)] \end{bmatrix} \cdot$$

$$\begin{bmatrix} [F(n-1)]^T_{BRO} & 0 \\ 0 & [F(n-1)]^T_{BRO} \end{bmatrix} \begin{bmatrix} [\tilde{P}_0(n-1)] & [\tilde{P}_1(n-1)] \\ [\tilde{P}_2(n-1)] & [\tilde{P}_3(n-1)] \end{bmatrix} \cdot$$

$$\begin{bmatrix} [F(n-1)]_{BRO} & 0 \\ 0 & [F(n-1)]_{BRO} \end{bmatrix} \begin{bmatrix} [I(n-1)] & 0 \\ 0 & [W(n-1)] \end{bmatrix} \cdot$$

$$\begin{bmatrix} [I(n-1)] & [I(n-1)] \\ [I(n-1)] & -[I(n-1)] \end{bmatrix}$$

which can be further written as

$$[X(n)] = \begin{bmatrix} [P] & [Q] \\ [R] & [S] \end{bmatrix} \tag{8}$$

where

$[P] = [X_0(n-1)]+[X_1(n-1)][W(n-1)]+[W(n-1)]$
$[X_2(n-1)]+[W(n-1)][X_3(n-1)][W(n-1)]$

$[Q] = [X_0(n-1)]-[X_1(n-1)][W(n-1)]+[W(n-1)]$
$[X_2(n-1)]-[W(n-1)][X_3(n-1)][W(n-1)]$

$[R] = [X_0(n-1)]+[X_1(n-1)][W(n-1)]-[W(n-1)]$
$[X_2(n-1)]-[W(n-1)][X_3(n-1)][W(n-1)]$

$[S] = [X_0(n-1)]-[X_1(n-1)][W(n-1)]-[W(n-1)]$
$[X_2(n-1)]+[W(n-1)][X_3(n-1)][W(n-1)]$

where the $[X_i(n-1)]$'s are the DFTs of the corresponding $[\tilde{P}_i(n-1)]$s.

Just as in the case of 2D-(WHT_h), (8) can be written in terms of the companion vectors corresponding to $[X(n)]$ and $[X(n-1)]$. However in this case it is a bit involved because from (8) we see that except for $[X_0(n-1)]$ all other DFTs of the data subblocks i.e. the $[X_i(n-1)]$s are multiplied on either or both sides by the matrix $[W(n-1)]$. A look at (8) tells us that as in the case of the 2D-(WHT_h), here also there will be an equation similar to (3) but with an additional interjected matrix to account for the multiplications by $[W(n-1)]$. A little thought will also reveal that this additional matrix will be a diagonal one.

From (8) we had seen that $[X_0(n-1)]$ does not get multiplied with any matrix, $[X_1(n-1)]$ is postmultiplied by $[W(n-1)]$, $[X_2(n-1)]$ is premultiplied by $[W(n-1)]$ and $[X_3(n-1)]$ is both post and premultiplied by $[W(n-1)]$. Therefore while writing (8) in terms of companion vectors the subscripts R(right), L(left) and B(both sides) have been used in the subblocks of the interjected matrix to indicate the kind of multiplication it corresponds. We have

$$\underline{x}(2n) = [[H_h(2)] \textcircled{x} [I(2(n-1))]] \cdot (\text{diag}[[I(2n-2)], [W_R(2n-2)], [W_L(2n-2)], [W_B(2n-2)]][\underline{x}_0(2n-2)^T \; \underline{x}_1(2n-2)^T \; \underline{x}_2(2n-2)^T \; \underline{x}_3(2n-2)^T]^T \tag{9}$$

where the $\underline{x}_i(2(n-1))$s are the companion vectors of the matrices $[X_i(n-1)]$s and $[W_R(2(n-1))]$, $[W_L(2(n-1))]$, $[W_B(2(n-1))]$ are the left hand side premultipliers for $\underline{x}_1(2(n-1))$, $\underline{x}_2(2(n-1))$, $\underline{x}_3(2(n-1))$ corresponding to the matrix products $[X_1(n-1)][W(n-1)]$, $[W(n-1)][X_2(n-1)]$ and $[W(n-1)][X_3(n-1)][W(n-1)]$ respectively.

Let us now derive the general formulae for $[W_R]$, $[W_L]$ and $[W_B]$.

For $[W_R]$ from the definition of $[W(n-1)]$ and expressing $[X(n-1)]$ into four subblocks we can write
$[X(n-1)][W(n-1)] =$

$$\begin{bmatrix} [X_0'(n-2)] & [X_1'(n-2)] \\ [X_2'(n-2)] & [X_3'(n-2)] \end{bmatrix} \begin{bmatrix} [W(n-2)]^{1/2} & 0 \\ 0 & w_4[W_4(n-2)]^{1/2} \end{bmatrix}$$

where $[W(n-2)]^{1/2} = [\{w_{ii}\}^{1/2}]$; $i = 0,1,2\ldots 2^{n-2}$ and $w_4 = \exp\{-i2\pi/4\} = -i$.

From the above equation we can write,
$[X(n-1)][W(n-1)] =$

$$\begin{bmatrix} [X_0'(n-2)][W(n-2)]^{1/2} \\ [X_2'(n-2)][W(n-2)]^{1/2} \end{bmatrix} \begin{bmatrix} w_4[X_1'(n-2)][W(n-2)]^{1/2} \\ w_4[X_3'(n-2)][W(n-2)]^{1/2} \end{bmatrix}$$

Expressing the above equation as the product of a matrix and the corresponding companion vectors we have,

$$[W_R(2n-2)] \cdot \underline{x}(2n-2) = (\text{diag}[[W_R(2n-4)], w_4[W_R(2n-4)], [W_R(2n-4)], w_4[W_R(2n-4)]]) \cdot [\underline{x}_0'(2n-4)^T \; \underline{x}_1'(2n-4)^T \; \underline{x}_2'(2n-4)^T \; \underline{x}_3'(2n-4)^T]^T$$

From this we get the recursive formula for $[W_R(2(n-1))]$. Proceeding on the lines of the derivation of $[W_R]$ we have,

$$[W_L(2(n-1))] = (\text{diag}[[W_L(2n-4)]^{1/2}, [W_L(2n-4)]^{1/2}, w_4[W_L(2n-4)]^{1/2}, w_4[W_L(2n-4)^{1/2}]])$$

$[W_B(2(n-1))] = (\text{diag}[[W_B(2n-4)]^{1/2}, w_4[W_B(2n-4)]^{1/2}, w_4[W_B(2n-4)]^{1/2}, -[W_B(2n-4)]^{1/2}])$

and $[W_R(0)] = [W_L(0)] = [W_B(0)] = 1$

The general formula given by (9) can be further expanded to give,

$$\underline{x}(2n)= [[H_h(2)] \otimes [I(2(n-1))]] \cdot (\text{diag}[[I(2n-2)], [W_R(2n-2)][W_L(2n-2)][W_B(2n-2)]]) \cdot (\text{diag}[[H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)], [H_h(2)] \otimes [I(2n-4)]]) \cdot (\text{diag}[[[I(2n-4)], [W_R(2n-4)], [W_L(2n-4)], [W_B(2n-4)]], [[I(2n-4)], [W_R(2n-4)], [W_L(2n-4)], [W_B(2n-4)]], [[I(2n-4)], [W_R(2n-4)], [W_L(2n-4)], [W_B(2n-4)]], [[I(2n-4)], [W_R(2n-4)], [W_L(2n-4)], [W_B(2n-4)]]]) \ldots\ldots (\text{diag}[[H_h(2)], [H_h(2)], \ldots [H_h(2)]]\tilde{\underline{p}}(2n) \quad (10)$$

where $\tilde{\underline{p}}(2n)$ is the companion vector corresponding to the $[\tilde{P}(n)]$ matrix.

Based on the above discussion the flowgraph for the 2D-DFT for n=2 is shown in Fig. 2.2.

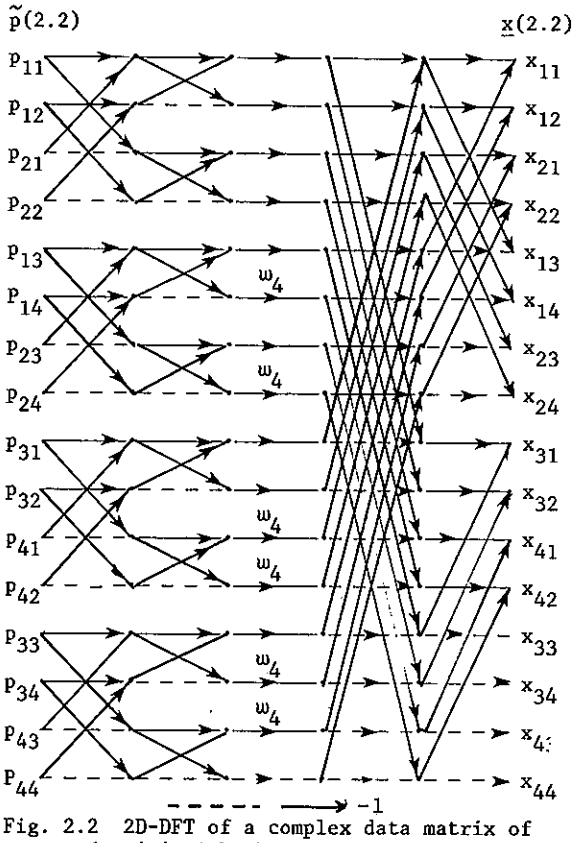


Fig. 2.2 2D-DFT of a complex data matrix of order 4x4 with the data and transform coefficients in the zig-zag format

NUMBER OF ADDITIONS AND MULTIPLICATIONS REQUIRED FOR A 2D-DFT OF N X N COMPLEX DATA

Additions. The number of additions remain the same as in the case of the staggered 2D-(WHT_h) because the additional interjected matrix in this case viz.

$(\text{diag}[[I(2n-2)], [W_R(2n-2)]], [W_L(2n-2)], [W_B(2n-2)]])$

consists of only the diagonal elements.

Multiplications. Analyzing (10) we get the general formula for the total number of multiplications as $3(N/4)[N(\log_2 N-2)+2]$.

The number of complex computations required for the generation of the 2D-DFT have been enlisted in Table I for different values of n both for the fast 2D-DFT case and repeated use of the 1D algorithm.

TABLE I. COMPARISON OF THE NUMBER OF COMPLEX ADDITIONS AND MULTIPLICATIONS FOR THE REPEATED USE OF THE FAST 1D ALGORITHM AND THE FAST 2D ALGORITHM (FOR COMPLEX INPUT DATA)

		FAST 2D		REPEATED FAST 1D	
n	N	Adds	Mults	Adds	Mults
2	4	64	6	64	8
3	8	384	60	384	80
4	16	2048	408	2048	544
5	32	10240	2352	10240	3136
6	64	49152	12384	49152	16512

CONCLUSION

Definitely the fast 2D algorithm gives us computational savings as compared to the repeated use of the 1D algorithm by 25% as can be seen from Table I. This is advantageous as far as transform processing of images are concerned where we are especially interested in the 2D case. For example choosing small subblocks may result in loss of correlation between adjacent blocks and give rise to distinct edges. In such cases if we have a computationally efficient fast 2D algorithm, going in for the computation of the 2D-DFT of a subblock of higher size is easier.

REFERENCES

1. J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," Math. Comput., vol. 19, pp. 297-301, Apr. 1965
2. Ph. W. Besslich, "Fast in place processing of pictorial data," in R. M. Haralick (Ed.), Pictorial Data Analysis, Springer Verlag, Berlin, pp. 43-68, 1983.
3. ____, "Hierarchical generation of 2D data structures," DAGM-Symp. GRAF/Osterveich, Oct. 1984.
4. N. Ahmed, K.R. Rao, and A.L. Abdussattar, "BIFORE or Hadamard Transform," IEEE Trans. Audio and Electroacoust. vol. AU-19, pp. 225-234, Sept. 1971.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ALGORITHM FOR HIGH-RESOLUTION DETECTION AND EQUALIZATION

G. Dietmar Achilles

University of Kaiserslautern
Electrical Engineering Department
D-6750 Kaiserslautern, Federal Republic of Germany

An FFT algorithm has been developed that is capable of resolving overlapping multiple echoes of a known signal. It is not sensitive to noise in the sense that ill-conditioning would occur. Preliminary results in testing the applicability to detection have been obtained.

1. PRINCIPLE OF THE METHOD

Aproblem which typically occurs in detection and equalization is that we are given a superposition

$$y(t) = \sum_k r_k\, x(t-t_k) \tag{1}$$

of echoes of a known signal, $x(t)$, and wish to determine the reflection coefficients, r_k, and the delays, t_k. The problem may be reduced to determining the r_k only, if the range of relative delays is subdivided by a sufficiently fine grid into 2M equidistant intervals, yielding approximately

$$t_k = n_k T\ , \quad n_k \text{ integer} \tag{2}$$

Equation (1) may then be rewritten in the form

$$y(t) = \sum_{k=0}^{2M-1} a_k\, x(t-kT) \tag{3}$$

with

$$a_k = \begin{cases} r_k & \text{for } k=n_k \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

With $Y(f)$ and $X(f)$ denoting the Fourier transforms of $y(t)$ and $x(t)$, respectively, we have from (3):

$$Y(f) = \sum_{k=0}^{2M-1} a_k\, X(f)\, e^{-j2\pi fkT} \tag{5}$$

Sampling $y(t)$ at frequency $1/T$ and performing DFT yields

$$\tilde{Y}(f) = \sum_{n=0}^{N-1} y(nT)\, e^{-j2\pi fnT} \tag{6}$$

Similarly, we have

$$\tilde{X}(f) = \sum_{n=0}^{N-1} x(nT)\, e^{-j2\pi fnT} \tag{7}$$

The relationship between $\tilde{X}(f)$ and $\tilde{Y}(f)$ may be obtained as follows:

$$\begin{aligned} \tilde{Y}(f) &= \sum_{\lambda=-\infty}^{\infty} Y(f-\lambda/T) = \\ &= \sum_{\lambda=-\infty}^{\infty} \sum_{k=0}^{2M-1} a_k\, X(f-\lambda/T)\, e^{-j2\pi(f-\lambda/T)kT} \\ &= \sum_{k=0}^{2M-1} a_k e^{-j2\pi fkT} \sum_{\lambda=-\infty}^{\infty} X(f-\lambda/T) \\ &= \sum_{k=0}^{2M-1} a_k e^{-j2\pi fkT}\, \tilde{X}(f) \end{aligned}$$

Thus we have

$$\tilde{Y}(f)/\tilde{X}(f) = \sum_{k=0}^{2M-1} a_k\, e^{-j2\pi fkT}\ , \tag{8}$$

which can be solved for the coefficients a_k by suitably sampling in frequency and applying inverse discrete Fourier transform. The specific algorithm to be developed depends on the type of its application. We first consider in some more detail the case of detection of superimposed echoes in radar or sonar systems.

2. APPLICATION TO DETECTION

2.1 Algorithm

We assume signals x(t) having triangular envelope, as they would ideally occur at the matched filter output of a radar or sonar receiver in response to rectangular pulses of duration Θ. Since we are primarily interested in echoes delayed by $t_k \leq \Theta$, we set

$$2MT = \Theta \tag{9}$$

For convenience, let us first consider signals without carrier. Fourier transform of a triangular pulse of duration 2Θ and unit peak value gives

$$X(f) = \Theta\{\frac{\sin \pi f\Theta}{\pi f\Theta}\}^2 \tag{10}$$

The DFT of samples x(nT) then yields (see for example [1]):

$$\tilde{X}(f) = \frac{1}{\Theta} \sum_{\lambda=-\infty}^{\infty} \{\frac{\sin\pi(f-\lambda/T)\Theta}{\pi(f-\lambda/T)}\}^2$$

$$= \frac{\sin^2 2\pi fMT}{2MT} \sum_{\lambda=-\infty}^{\infty} \frac{1}{\{\pi(f-\lambda/T)\}^2}$$

$$= \frac{T}{2M}\{\frac{\sin 2\pi MT}{\sin \pi fT}\}^2 \tag{11}$$

By substituting these results into (8) we obtain

$$\tilde{Y}(f) = \frac{T}{2M}\{\frac{\sin 2\pi fMT}{\sin\pi fT}\}^2 \sum_{k=0}^{2M-1} a_k e^{-j2\pi kT} \tag{12}$$

Choosing now N to be a multiple integer of 4M

$$N = I4M \ , \quad I \text{ integer} \tag{13}$$

and frequency sampling points

$$f_m = \frac{I(2M+1)}{NT} = \frac{2m+1}{4MT} \tag{14}$$

we are able to turn the sum in (12) into a DFT:

$$\sum_{k=0}^{2M-1} a_k e^{-j2\pi fkT} =$$

$$= \sum_{k=0}^{2M-1} a_k e^{-j\pi k/(2M)} e^{-j2\pi km/(2M)} \tag{15}$$

Thus, equation (12) can be solved for the reflection coefficients a_k by an inverse DFT. Since

$$\sin(2m+1)\pi/2 = \pm 1$$

will be squared in (12), we finally obtain:

$$a_k = \tag{16}$$

$$= \frac{e^{j\pi k/(2M)}}{T} \sum_{m=0}^{2M-1} \tilde{Y}\{\frac{2m+1}{4MT}\}\sin^2\frac{\pi(2m+1)}{4M} e^{j2\pi km/(2M)}$$

2.2 Experimental Results

The above considered model is of course far from realistic conditions in a typical radar or sonar environment. In order to answer the question whether or not the algorithm or some modification of it would be applicable to relevant detection problems, a thorough simulation under more realistic conditions including ground clutter and MTI processing, target scattering, noise, and bandwidth limitation has to becarried out. Preliminary results have obtained concerning the equidistant sampling of time and thus range axis, presence of noise, and random carrier phase [2].

Testing the algorithm with signals superimposed by white Gaussian noise has shown that noise does not present a specific problem in the sense that ill-conditioning would occur. Obviously, the algorithm is not particularly sensitive to noise, which was to be expected, since division by small numbers, which is typical for most deconvolution methods, does not have to be performed in equation (16).

Targets being located between sampling points in range can be detected without problem. It can be seen from fig.1, where five such targets have been chosen that refinement of the sampling grid results in focussing on the target echoes.
Finally, the method was apllied to pulses with random carrier phase. We observe a fluctuation of the detected reflection coefficients,

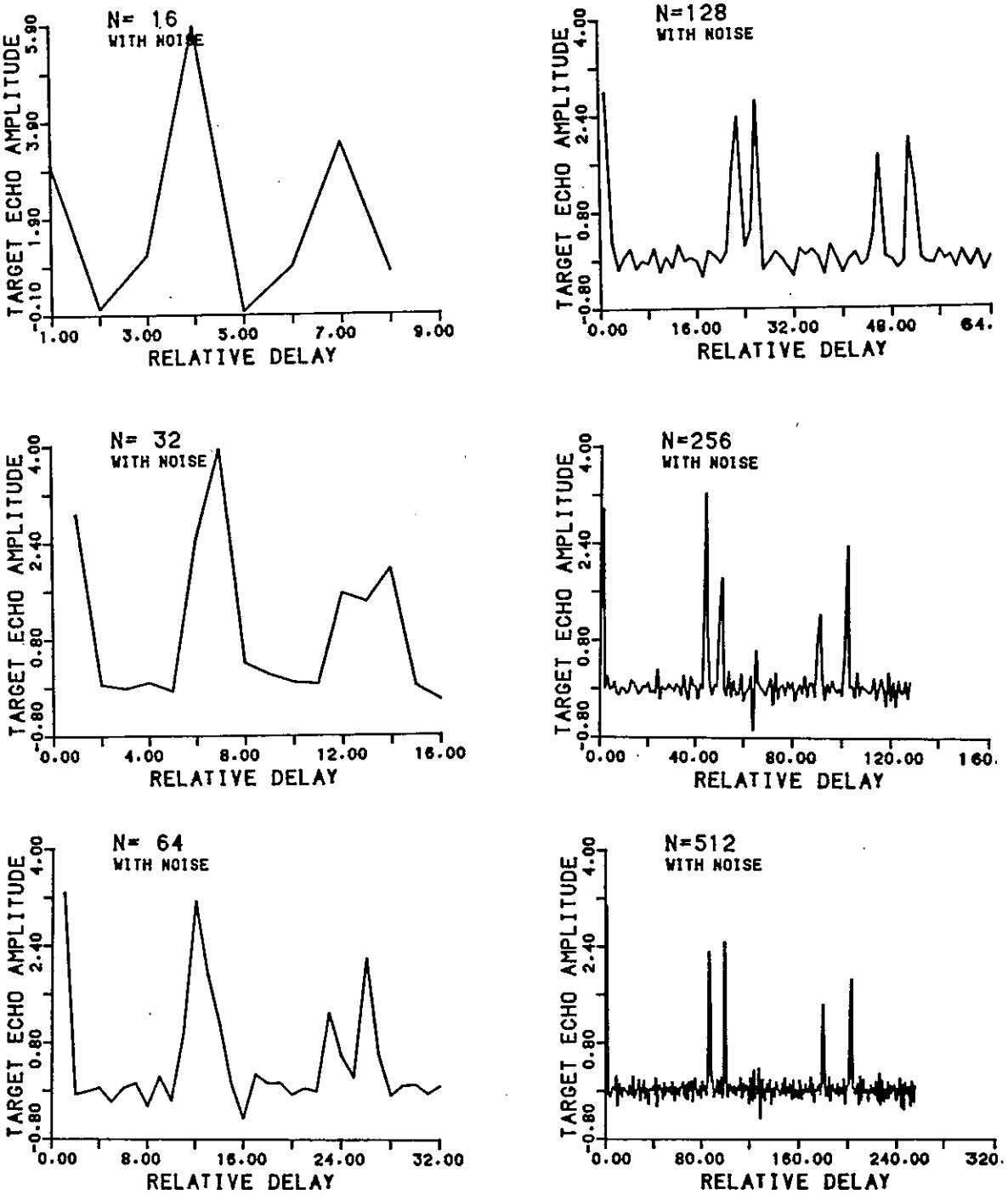


Figure 1: Detection of five targets located between sampling points (after [2])
Reflection coefficients: $a_1=3$, $a_2=4$, $a_3=3$, $a_4=2$, $a_5=1$

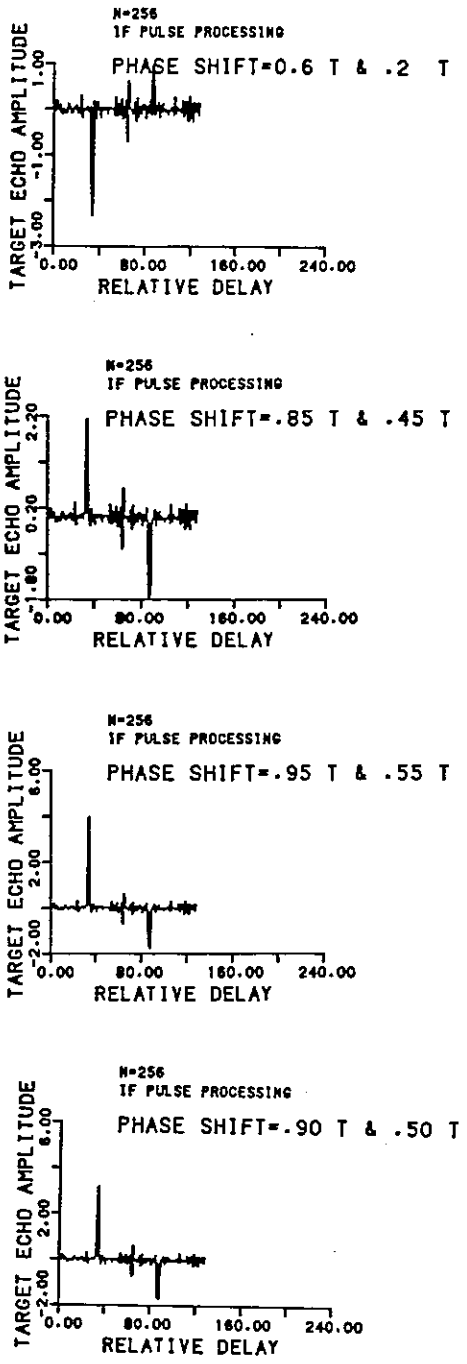


Figure 2: Fluctuation of detected reflection for two echo pulses with different phase angles (after [2])

depending on the phase angle, as shown in figure 2. On the average, however, the target is detected.

3. APPLICATION TO EQUALIZATION

Application of the above principle to a zero-forcing transversal equalizer would of course result in solving the respective simultaneous equations by DFT, which does not present anything new. We are more interested in the other type of equalizer that is aimed at restoring the undistorted channel with optimum noise performance. A channel distortion that may be removed by a transversal equalizer then produces a signal described by equation (3), where however, all of the coefficients a_k will be nonzero, in general. The a_k can then be determined in the same way as above and from these the equalizer weights may be easily computed.

4. CONCLUSION

These preliminary experimental results encourage to continue with testing the algorithm under more realistic conditions with respect to detection of radar and sonar signals.

ACKNOWLEDGEMENT

The author is indebted to Mr. R. Windecker for discussions.

REFERENCES

[1] Achilles, D., Die Fourier-Transformation in der Signalverarbeitung (Springer, Berlin Heidelberg-New York, 1985, second edition)

[2] Ismail, T.H., An FFT method of digital radar signal processing to enhance range resolution, M.S. Thesis, University of Petroleum and Minerals, Dhahran, 1984

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N5.6

USE OF PARAMETRIC METHODS TO DETECT MICROPROCESSOR'S FAILURES

J-L GASSER - W. YE - P. CSILLAG

ENSEEIHT/GAPSE 2, rue Camichel 31071 TOULOUSE Cedex (FRANCE)

ABSTRACT : In this paper, we present some results about random testing of microprocessors. Six microprocessors EF 6809 (Thomson-Efcis) were submitted to electronic (β^-) radiations, and we analysed the moment of occurrence and the nature of some failures. These observations allowed us to elaborate a new method for the signature analysis of microprocessors. This method is called the Rademacher expansion of histograms, and seems to be very efficient for detecting some kinds of failures. After a short introduction to the VLSI testing problem, we present our random testing method and we describe in 2.the failures we observed during the irradiation. In 3., we review the analysis methods which seem to be interesting and discuss them. In 4. we present some results and in 5. we discuss the limitations of the method and present possible applications.

KEYWORDS : Random testing/ Microprocessor/ Electronic radiations/ Intermittent failures/ Signature analysis/ AR modelisation/ Rademacher-Walsh transform.

1 - Introduction to the testing method

It becomes more and more difficult to test the VLSI components because the number of gates is ever increasing and the internal architecture is more and more complex /1/ /2/. The exhaustive testing was a good method for testing some LSI components, but some problems appear with sequential logic. With the VLSI, it was necessary to build some new and efficient testing methods /3/. We choose random testing, which has already given good results /4/ /5/, and developed a new testing approach.

Our study was started by the CNES (Centre National des Etudes Spatiales, Toulouse, France), dept QPE/FT (Qualité Produit Environnement/ Fiabilité Test). The hardware implementation and the first results are presented in /6/. In this study, the input sequence is similar to a white noise; all the opcodes of the microprocessor (MP) are applied with the same probability, and we analyse the MP response by the use of signal processing methods.

We studied the Thomson-Efcis MP EF6809. An automaton generates all the opcodes of the MP using an M-sequence. The response of the MP was furnished by the the data and address busses, which were stored for analysis. We did not use interrupt capabilities of the MP, and more generally we did not test the control bus.

A first approach showed that it was necessary to select some relevant signals to make a coherent analysis. In this paper, we limit ourselves to some results on the Data Bus (DB) during a "write" cycle, which reflects internal registers of the MP.

2 - Description of the experiment Observed failures

We carried out this experiment in collaboration with the CERT (Centre d'Etudes et de Recherches de Toulouse, France), dept DRTS (Département d'Etudes et de Recherche en Technologie Spatiale). We submitted six MP EF6809 in ceramic package, with the same date code, to β^- radiations. These resulted in threshold changes in the gates; so, the experience simulated the artificial ageing of the components. The results we obtained /7/ were similar to those presented by many authors /8/. We did not stop the irradiation after these observations and we analysed the degradation of the MP behavior. The nature of the failures was the same for each MP, and we brought out some weaknesses of the MP.

Errors are often imputable to a single bit, occasionally to several bits, but never to all the bits. An error rarely appears alone, so we can observe batches of them. This characteristic of the random testing have already been observed by some authors /9/, and can be very interesting for signal processing methods. Much errors appear on the bit 4; they are often the same kind, "single stuck at 0" errors.

3 - Analysis methods

3.1 Response of the MP

The output sequence appeared to be weakly autocorelated, with nonuniform distribution. We modelised the DB as a signal and obtained some results /10/; but some failures localised on the lowest significant bits were not detected. The irradiation experiment shows that failures are often localised on a single bit; that is why we elaborated a method for analysing every bit of DB. We called this method the "Rademacher Expansion of Histogram"; it has been developed in a special case /11/, but we generalised its application.

3.2 Histogram analysis

We can make a histogram with the numerical values appearing on the DB. In order to analyse this histogram, we built an estimator which describes the behaviour of each bit. The output is a variable distributed over $[0,2^n-1]$, and $B=b_{n-1},..b_0$ is its binary representation. Let a failure process transforming each b_k into b'_k be defined by:

a) $b'_k=1 \quad \forall b_k$ with probability $P_1(k)$; "single stuck at 1".
b) $b'_k=0 \quad \forall b_k$ with probability $P_0(k)$; "single stuck at 0".
c) $b'_k=b_k$ with probability $P(k)=1-P_1(k)-P_0(k)$

d) The failure process is independent of b_k and each b_k is independent of each other.

We define:

$$P[b_k=0] = \frac{1+\lambda_k}{2} \qquad P[b_k=1] = \frac{1-\lambda_k}{2} \qquad (1)$$

when λ_k is the distribution parameter, uniform distribution is obtained with $\lambda_k=0$.

We define a failure parameter called α_k :

$$\alpha_k = P_0(k)(1-\lambda_k) - P_1(k)(1+\lambda_k) \qquad (2)$$

One can show that /14/:

$$\hat{\alpha}_k = -\hat{\lambda}_k + \sum_{m=0}^{2^{n-1}} \hat{p}(m)\,Wal(2^{n-k},m2^{-n}) \qquad (3)$$

$$E[\hat{\alpha}_k] = 0 \qquad k=0,...n-1$$

$$Var[\hat{\alpha}_k] = \frac{1}{N}\,(1-(\lambda_k+\alpha_k)^2)$$

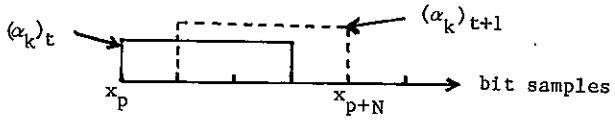
where $\hat{p}(m)$ is the probability of the value m in the histogram, N is the number of samples, and Wal is the Walsh-Rademacher transform. (3) also provides an estimator of λ_k.

In relation to the binomial rule, we can compute the confidence interval of α_k under some constraints: the choice of the number of samples N, the false alarm probability (FAP), the value of λ_k. We built a test using a Neyman-Pearson approach:
- Hypothesis H_0 : there is no failure and the FAP is a chosen value, e.g. 5% or 1%.
- Hypothesis H_1 : there is a failure, and α_{kF} is the value of the failure parameter defined in 2. The non detection probability (NDP) is then known.

The NDP is a continuous variable which decreases as α_k increases. There is a value α_{kFC} whose NDP has the same value as the FAP, this case could be called "certain detection". We choose N, FAP, and compute the reliability interval $[1^-,1^+]$. We estimate α_k, and the test is the following:
- if $\hat{\alpha}_k \in [1^-,1^+]$, MP is good.
- if $\hat{\alpha}_k > 1^+$, MP presents "single stuck at 0" failures.
- if $\hat{\alpha}_k < 1^-$, MP presents "single stuck at 1" failures.

3.3 AR modelisation.

The histogram analysis is a "static" method, and we want to estimate the spectral properties of the α_k. We made an AR modelisation of the α_k. We computed α_k with a given window lenght N; by sliding this window each new sample, we built a new signal which reflects the time behavior of :



The autoregressive coefficients of the AR model were estimated by using Morf's algorithm /12/. The power spectral density and the Linear Prediction Error (LPE) were computed according to /15/. We know that the mean value of α_k is a relevant parameter, and that for little failures we can observe some mean value changes. A very simple method for detecting this kind of nonstationarities, based on the level crossing analysis of the LPE has given good results /13/. So, we analysed the level and the behavior of LPE.

Many other methods are available for detecting nonstationarities, but we tried to use methods which use no prohibitive computing time.

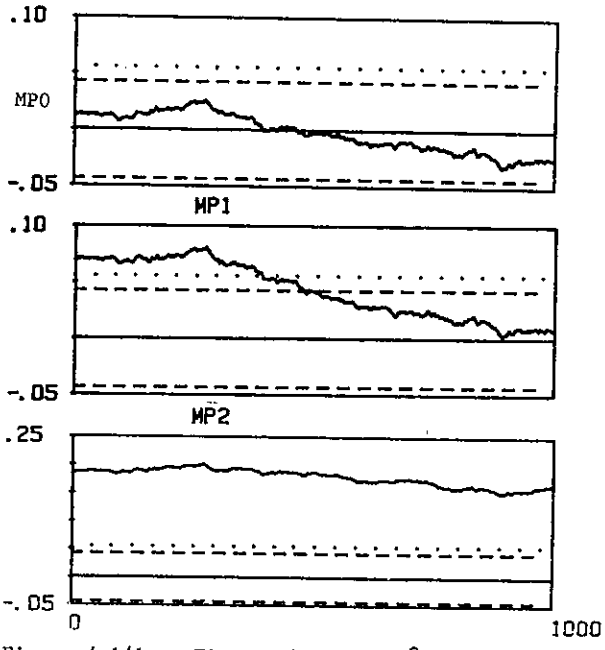


Figure 4.1/1 : Time evolution of $\hat{\alpha}_4$ with N = 2048
Confidence intervals : ...PFA =1% ---PFA =5%

4 - Results

We only give results about bit 4, in three cases:
the MP is right (MP0); the MP presents "little failures" (MP1), e.g. the number of failures is low; the MP presents more failures than MP1 (MP2). Over N=4096 samples, the "stuck at 0" probabilities are 2,2% and 8,2% for MP1 and MP2 respectively.

4.1 Histogram analysis

We estimate the density probability parameter $\hat{\lambda}_4$ =0.090. We choose a number of samples N as a power of 2, so N=16,32,...2048. The confidence intervals for several number of samples and the estimations of the values of the certain detection are given in the following:

N	FAP	$P_0/P_1=0$	$P_1/P_0=0$	1^-	1^+
256	0.1%	40%	32%	-.201	.206
512	0.1%	26%	21%	-.145	.144
1024	1%	17%	14%	-.80	.079
2048	5%	8.5%	7%	-.043	.043

- with N=16,...128, and FAP=0.1%, the variance of $\hat{\alpha}_4$ is too high ; the false alarms of MP0 were numerous.
- with 256 < N < 512 and FAP=0.1, the results began to be interesting. We observed no false alarm of MP0, even during long observation. Some failures of MP1 were detected, corresponding to the maxima of the number of failures in the window. But some others were not detected. We observed many failures with MP2.
- with N=1024 or N=2048, and FAP = 5%, we did not observe any false alarm of MP0.

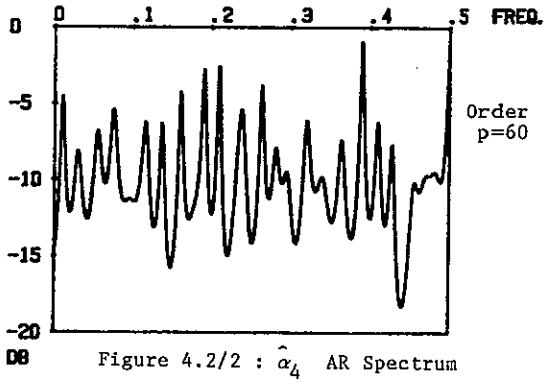


Figure 4.2/2 : $\hat{\alpha}_4$ AR Spectrum

But MP1 presented several failures, and MP2 was always signaled as bad (see figure 4.1/1).

With a greater number of samples, it becomes possible to estimate α_4 with good precision, and to observe the degradation of the MP.

4.2 AR modelisation

A first analysis proved it was necessary to take every 16th sample of the signal. In the following we discuss the sampled signal. We analysed the spectral properties of the signal using an AR modelisation: an order 60 or 70 was necessary to bring out many frequencies (see figure 4.2/2). The signal to noise ratio was about 10 dB.

But we did not observe significant frequency changes or power changes between the right MP and a MP which presented failures. After computing a lot of samples, we established that the signal could not be stationnary; either the spectral properties were time dependent, or the spectral components were too various and it was difficult to make a good estimation of the poles.

We did not observe significant changes of the LPE in each case (see figure 4.2/3). We can only say that in the case of "stuck at 0" failures, the mean value of LPE is significantly higher than value 0. This phenomenon is probably caused by local changes of the mean value of $\hat{\alpha}_4$ (see figure 4.2/3).

It was not possible to make an efficient analysis of α_4 with other window values because of the hardware limitations.

5 - Conclusion

We presented an original method for testing microprocessors. It seems that considering the output bits of the component as a numerical value gives results only when a high weighted bit presents failures; we presented a method which indicates the

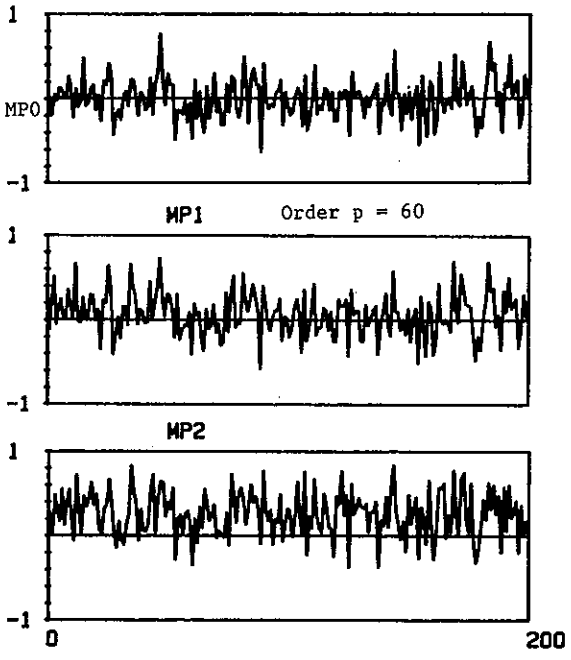


Figure 4.2/3 : $\hat{\alpha}_4$ Linear Prediction Error

behavior of each bit. The Rademacher expansion of histogram is a sensitive method for detecting a large class of failures. The AR modelisation of the parameter does not give significant results, either because of the nonstationnarity of or because of the high order necessary to estimate the spectrum.

It could be difficult to carry out theoritical estimations about the accuracy of our method, but it is possible to establish it experimentally.

The testing method can be easily extended to other VLSI components. We can consider the use of them for on-line testing of logic systems. It is interesting for testing microprocessors submitted to spatial conditions. In our experiment, we obtained a good estimation of the vulnerability of the MP to β^- radiations, and we brought out some weaknesses in the MP which could not be detected by conventionnal testing methods /7/.

Acknowledgment

The authors wish thank D. FALGUERE and J. BOURRIAU from the CERT of Toulouse for helping them to carry out the irradiation experiment.

This work is a part of J.L. GASSER's and W. YE's Doctoral Thesis

REFERENCES

/1/ D.P. SLEWIOREK - L. KWOK-WOON LAI "Testing of digital systems", Proc. of IEEE, vol.69, N°10, October 1981.

/2/ T.W. WILLIAMS - K.P. PARKER "Design for testability- a survey", Proc. of IEEE, vol N°71, January 1983.

/3/ Projet pilote SURF, bilan et perspectives, January 1982.

/4/ X. FEDI "Contribution à l'étude expérimentale du test aléatoire de microprocesseurs 8 bits", Thèse DI, INP Grenoble, 1983.

/5/ P. THEVENOD-FOSSE "Test aléatoire de microprocesseurs- Application au 6800", Doctorat-ès-Sciences Grenoble, 1983.

/6/ M. FORNOFF "Application des méthodes de traitement du signal au test aléatoire de microprocesseurs", Thèse DI, INP Toulouse, 1983.

/7/ J.L. GASSER "Détection et localisation de pannes de microprocesseurs par des méthodes de traitement du signal", Doctorat de l'INP Toulouse, to be published in September 1986.

/8/ IEEE Trans. on nuclear science, many papers have been presented in the December rewiews since 1977.

/9/ R. DAVID - P. THEVENOD-FOSSE "Un outil pour l'étude du test aléatoire de la partie opérative de microprocesseurs", Journées IRIA, Projet pilote SURF, January 1980.

/10/ W. YE "Etude de la pertinence de la modélisation paramétrique des signaux pour le test aléatoire des microprocesseurs", Doctorat de l'INP Toulouse, to be published in September 1986.

/11/ F.CASTANIE - D. DUBE "Automated test of digitizing systems by Rademacher expansion of histograms", Acta IMEKO 1982, Berlin WEST, Vol V/II.

/12/ MORF M. "Eficient solution of covariance equations for linear prediction" IEEE, Trans. on Acoustic, Speech and Signal Processing, ASSP Vol.25, N°5, Oct.77, pp.429-433.

/13/ E. DAYMIER - F. CASTANIE "Analyse d'une méthode de détection des sauts de moyenne et de variance", 10e Colloque sur le Traitement du Signal et ses Applications, May 1985, Nice, France.

/14/ J.L. GASSER "Utilisation des fonctions de Walsh-Rademacher pour l'analyse d'histogrammes", note interne GAPSE, ENSEEIHT, April 1986.

/15/ S.M. KAY - S.L. MARPLE "Spectrum analysis - A modern perspective" Proc. of IEEE, vol.69, N°11, 1981.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

N5.8

OPTIMAL QUADRATIC SYSTEMS FOR DETECTION AND ESTIMATION

B. PICINBONO and P. DUVAUT

Laboratoire des Signaux et Systèmes*
ESE - Plateau du Moulon, 91190 Gif-sur-Yvette, FRANCE.

The problem of optimal quadratic systems for detection and estimation without any Gaussian assumption is considered. Usually quadratic systems are used for the detection of stochastic signals in Gaussian noise. Using the deflection criterion, it is shown that the optimal systems for detection can be obtained from a linear equation using fourth order moments of the noise. Solutions of such an equation are given in the case of fourth order white noise. Singular detection for quadratic systems is also discussed.

1. INTRODUCTION

Quadratic systems are very often used in signal detection or estimation problems. For example, the optimal receiver for testing two Gaussian distributions with zeromean value and different covariance matrices is a quadratic system in the form $\underline{x}^T M \underline{x}$, where M is a symmetric matrix. But such quadratic systems can be used in many cases other than Gaussian detection problems, and the aim of this paper is to give a general description of their applications in detection and estimation problems without the Gaussian assumption.

A quadratic system calculating $S(\underline{x})=\underline{x}^T M \underline{x}$ is defined by the matrix M and the very general class of problems considered in the following is to find an optimal matrix M_o such that the system $S(\underline{x})$ associated to this matrix is optimal in some particular sense. For example, it is interesting to find the matrix M giving an optimal result for the detection of a random signal (not necessarily Gaussian) in a Gaussian noise. Using the deflection criterion [1], which is a particular method of classification [2] or distance criterion [3], the optimal matrix was obtained in [4]. The result is the matrix form of the Eckardt filter [5] obtained by other methods long before. The deflection criterion is also a contrast criterion, as discussed in [6]. More recently, quadratic systems with Gaussian assumption have also been studied from a complexity aspect [7].

But optimal quadratic systems have been obtained only in the Gaussian case, because it is necessary to use high order moments of $\underline{x}$, and these moments have a simple form in the Gaussian case. An attempt to suppress the Gaussian assumption has been presented in [8], but without general results.

The purpose of this paper is to show that some optimum quadratic systems for detection and estimation can be calculated, even without the Gaussian assumption. Moreover, these are always obtained by the solution of a system of linear equations for which many well-known and effective techniques can be applied.

2. OPTIMAL SYSTEMS FOR DETECTION

Let us consider an observation vector $\underline{x}$ which is a real random vector of R^N. Some statistical properties of $\underline{x}$ are known under the two possible hypotheses H_o (noise only) and H_1 (signal plus noise). Under H_o we assume that $\underline{x}$ is zeromean, $E_o(\underline{x}) = \underline{0}$, and introducing the components x_i, or x(i), of $\underline{x}$, we define the four first moments by

$$C(i,j) \triangleq E_o(x_i x_j) \qquad (2\text{-}1)$$

$$B(i,j,k) \triangleq E_o(x_i x_j x_k) \qquad (2\text{-}2)$$

$$A(i,j,k,l) \triangleq E_o(x_i x_j x_k x_l) - C(i,j)C(k,l), \qquad (2\text{-}3)$$

where E_o means the expectation value under H_o. These functions clearly have obvious symmetries obtained by permutations between the components x_i, and are discussed below. Moreover, the quantities C(i,j) are the matrix elements of the covariance matrix C of $\underline{x}$ defined by $E_o[\underline{x}\,\underline{x}^T]$. Under H_1 we assume that

$$E_1(\underline{x}) = \underline{s} \qquad (2\text{-}4)$$

$$E_1(\underline{x}\,\underline{x}^T) = C_1 + \underline{s}\,\underline{s}^T , \qquad (2\text{-}5)$$

which means that C_1 is the covariance of $\underline{x}$ under H_1.

To any symmetric NxN matrix M, we associate the function of the observation, or the statistic, defined by

$$S(\underline{x}) \triangleq \underline{x}^T M\underline{x} - Tr(C\,M) \quad , \qquad (2\text{-}6)$$

where the trace or the constant term is introduced to ensure that $S(\underline{x})$ is zeromean under H_o. Our aim is to find the optimal matrix M such that the contrast C(S) is maximum. This contrast is defined by

$$C(S) \triangleq \frac{[E_1(S)-E_o(S)]^2}{V_o(S)} = \frac{E_1^2(S)}{E_o(S^2)} \quad , \qquad (2\text{-}7)$$

where $V_o(S)$ is the variance of S under H_o. The second equality is clearly a consequence of the fact that S is zeromean under H_o.

It is necessary to justify briefly the use of the contrast criterion in detection problems. A more detailed discussion can be found in [6]. At first we can notice that the contrast C(S) is exactly the deflection criterion introduced in [2] and used in [4] to find the Eckardt filter. It is also sometimes called a distance between H_o and H_1 [3]. Secondly, the contrast has a strong statistical meaning, because it can be found that the statistic giving the maximum contrast is precisely the likelihood ratio $L(\underline{x})$, which is a sufficient statistic for the test between two simple hypotheses H_o and H_1. In other words the statistic (2-6) giving the contrast maximum is the best quadratic approximation of the absolute optimal statistic which is the likelihood ratio [6].

After these preliminary considerations we can present the calculation of the optimal matrix M_o giving the maximum value of C(S).

Using (2-5) and (2-6) we easily find

$$E_1(S) = Tr(M\Gamma) \quad , \qquad (2\text{-}8)$$

where

$$\Gamma \triangleq C_1 - C + \underline{s}\,\underline{s}^T \quad . \qquad (2\text{-}9)$$

In many practical problems it is assumed that under H_1 the observation is a sum of the noise and an uncorrelated signal. The consequence is that $C_1 = C + C_s$, and then

$$\Gamma = C_s + \underline{s}\,\underline{s}^T \quad , \qquad (2\text{-}10)$$

where C_s is the covariance matrix of the random signal and $\underline{s}$ its expectation value.

Using (2-1), (2-2), (2-3) and the definition (2-6), the denominator of (2-7) becomes

$$V_o = E_o(S^2) = \sum_{i,j,k,l} A(i,j,k,l)M(i,j)M(k,l). \qquad (2\text{-}11)$$

In order to find the optimum matrix M_o it is worth writing this variance as a scalar product of vectors. For this purpose we consider any matrix M as a vector, the multiplication by a scalar λ giving λM. Let us now take two such vectors M and M' and introduce

$$\langle M, M' \rangle \triangleq \sum_{i,j,k,l} A(i,j,k,l)M(i,j)M'(k,l). \qquad (2\text{-}12)$$

This expression satisfies all the necessary requirements in order to define a scalar product. In particular the square of M is positive because it is the variance of S defined by (2-11). This variance could be equal to zero, but in this case that would mean that $S(\underline{x})$ is almost surely equal to zero, or with (2-6)

$$\underline{x}^T M\,\underline{x} \overset{a.s.}{=} Tr(CM) \quad . \qquad (2\text{-}13)$$

That also means that under H_o the random vector $\underline{x}$ belongs to a subset of R^N defined by (2-13), and we exclude in principle this possibility in our discussion. Then if the square of M is equal to zero, M=0.

Let us now consider the numerator of C(S) appearing in (2-7) and defined by (2-8). Our aim is to write $E_1(S)$ in the form

$$E_1(S) = \langle M, M_o \rangle \quad . \qquad (2\text{-}14)$$

Using (2-8) and the definition (2-12) we easily find that M_o is obtained by the linear equation

$$\sum_{k,l} A(i,j,k,l)M_o(k,l) = \Gamma(i,j) \quad , \qquad (2\text{-}15)$$

As a consequence, the contrast becomes

$$C(S) = \frac{\langle M, M_o \rangle^2}{\langle M, M \rangle} \quad , \qquad (2\text{-}16)$$

and we deduce from the Schwarz inequality that its maximum value is obtained for the matrix M_o defined by (2-15).

This maximum is the square of M_o which can also be written in the form

$$d_o^2 \triangleq \langle M_o, M_o \rangle = \sum_{i,j} M_o(i,j)\Gamma(i,j) = Tr[M_o\Gamma]. \qquad (2\text{-}17)$$

It is important to note that because of all the symmetries of the elements appearing in S, the matrix M_o is symmetric, which was assumed at the very beginning.

In conclusion, the optimum filter $S_o(\underline{x})$ for the detection is defined by (2-6) where M is deduced from (2-15). Moreover, the corresponding contrast, which is the maximum possible for a quadratic system, is given by (2-15). The main interest of this result is that no Gaussian assumption at all has been introduced, and even in this very general case the optimum statistic is calculated by a linear system.

We will now present some consequences of this result.

As indicated above, the solution M_o of (2-15) is a symmetric matrix. Indeed, the

functions A and C defined by (2-1) and (2-3) have very strong symmetry properties. In particular C(i,j)=C(j,i) but the symmetry of A is a little more complicated. We see in (2-3) that A is invariant if we permute i and j or k and l or even the pair (i,j) with the pair (k,l).

Furthermore it appears that the third order moment B defined by (2-2) does not play any role in the solution, which is a consequence of the quadratic structure of the receiver.

On the other hand the linear equation (2-15) can be considered as a normal equation defining an optimal matrix M_o by

$$M_o = A^{-1} \Gamma , \qquad (2\text{-}18)$$

where now A is a matrix depending on four indices.

Let us now consider the **Gaussian case** for which we have

$$A(i,j,k,l) = C(i,k)C(j,l) + C(i,l)C(j,k). \qquad (2\text{-}19)$$

Using this relationship in (2-15), we get $2\, C M_o C = \Gamma$, or

$$M_o = \frac{1}{2} C^{-1} \Gamma C^{-1} , \qquad (2\text{-}20)$$

which is the result obtained in [4] and also discussed in [9]. As a matter of fact it is the matrix form of the Eckardt filter [5]. The corresponding maximum contrast is given by (2-17) which becomes

$$d_o^2 = \frac{1}{2} \mathrm{Tr}[(C^{-1} \Gamma)]^2 . \qquad (2\text{-}21)$$

At the end of this section it is worth discussing the relationship between infinite contrast and singular detection.

Singular detection has been extensively studied, particularly in the Gaussian case, and it is not necessary here to indicate all the references corresponding to this problem. We have only to recall that singular detection means the possibility to reach simultaneously a false alarm probability equal to zero and a detection probability equal to one. In the case of the detection of a deterministic signal $\underline{s}$ in a Gaussian noise $N(\underline{o}, C)$ it is well-known that singular detection appears if $d^2 = \underline{s}^T C^{-1} \underline{s}$ becomes infinite. We are then interested in knowing if an infinite value of the maximum contrast means that the detection problem is singular. We will show that this is often the case, but this condition alone does not guarantee singular detection.

Indeed, the fact that d^2 becomes infinite means that there exists at least one matrix M for which the variance $V_o(S)$ appearing in (2-7) is equal to zero because the numerator of (2-7) is always assumed to be finite. If this variance is zero, then for this matrix and under H_o , equation (2-13) is valid. This means that there exists a statistic $S_o(\underline{x})$ defined by (2-6) and (2-13) almost surely equal to zero under H_o. Using this statistic as decision function where H_o is decided if $S_o(\underline{x})=0$ and H_1 if $S_o(\underline{x}) \neq 0$, the detection problem becomes singular if $S_o(\underline{x})$ is almost surely different from zero under H_1, or

$$\Pr[S_o(\underline{x}) \neq 0 \mid H_1] = 1 . \qquad (2\text{-}22)$$

Then an infinite contrast is a necessary condition of singular detection but not a sufficient condition: we also have to check that a statistic exists such that (2-22) is satisfied.

3. QUADRATIC SYSTEMS FOR DETECTION IN FOURTH ORDER WHITE NOISE

A large number of papers dealing with detection problems in non Gaussian noise assume that this noise is white, or more precisely that the components x_i of the observation vector are i.i.d. random variables. This greatly simplifies the calculation of the likelihood ratio because the statistical properties of the noise are completely defined by univariate distribution. In fact, the concept of white noise is not at all precise and there is a large number of different kinds of white noise [10].

As we are using moments only up to the fourth order, we will introduce the concept of fourth order white noise. Let us consider a sequence of zeromean i.i.d. random variables x_i with first order moments $m_k = E(x_i^k)$. The moments defined by (2-1), (2-2) and (2-3) then become

$$C(i,j) = m_2 \, \delta[i,j] \qquad (3\text{-}1)$$

$$B(i,j,k) = m_3 \, \delta[i,j,k] \qquad (3\text{-}2)$$

$$A(i,j,k,l) = m_2^2 \{\delta[i,k]\,\delta[j,l]+\delta[i,l]\,\delta[j,k]\} + (m_4 - 3m_2^2)\, \delta[i,j,k,l] . \qquad (3\text{-}3)$$

The symbols $\delta[\ ]$ are extensions of the Kronecker delta symbols and are equal to one if all the indices are equal and zero in the other cases. Expression (3-3) is also given in [8].

We will say that a noise is fourth order white if its first moments are given by (3-1), (3-2) and (3-3), and no particular assumption is introduced on the higher order moments. It is in some sense an extension of the concept of second order white noise where only (3-1) is introduced.

Let us now suppose that we have to detect a signal defined by Γ in a fourth order white noise by using a quadratic system. The optimal system is defined by equation (2-15) where A is given by (3-3). We then easily obtain

$$2m_2^2 M_o(i,j) + \delta[i,j]\{(m_4-3m_2^2)M_o(i,i)\} = \Gamma(i,j) \qquad (3\text{-}4)$$

The solution of this equation is very simple and we get for $i \neq j$

$$M_o(i,j) = \frac{1}{2m_2^2}\,\Gamma(i,j) \qquad (3\text{-}5)$$

and for $i=j$,

$$M_o(i,i) = \frac{1}{m_4-m_2^2}\,\Gamma(i,i)\ . \qquad (3\text{-}6)$$

The performance of the optimal system is characterized by the value of its contrast which is defined by (2-17) which gives

$$d_o^2 = \frac{1}{2m_2^2} \sum_{i\neq j} \Gamma^2(i,j) + \frac{\Gamma^2(i,i)}{m_4-m_2^2}\ . \qquad (3\text{-}7)$$

First let us consider the case of a fourth order Gaussian white noise, also called "Gaussian-like noise" in [8]. This means that $m_1=m_3=0$ and $m_4=3m_2^2$, which are the values of the four first moments of a normal distribution. Using the previous equations, we find

$$M_o(i,j) = \frac{1}{2m_2^2}\,\Gamma(i,j)\ , \qquad (3\text{-}8)$$

which of course is a particular value of (2-20). Moreover, (2-21) takes the form

$$d_G^2 = \frac{1}{2m_2^2} \sum_{i,j} \Gamma^2(i,j)\ . \qquad (3\text{-}9)$$

Now let us consider (3-7). From the Schwarz inequality we have $m_4 \geq m_2^2$. If $m_4 \to m_2^2$, d^2 becomes infinite, a situation which will be discussed later. We can now verify that $d_o^2 > d_G^2$ if $m_4 < 3m_2^2$ and $d_o^2 < d_G^2$ if $m_4 > 3m_2^2$.

Let us now discuss the possibility of singular detection in a fourth order white noise. As indicated above, a necessary condition for this is that d^2 become infinite, and we see in (3-7) that this is possible only if $m_4=m_2^2$. To check if this effectively gives a singular detection, we also have to consider (2-22). From the Schwarz inequality it appears that the condition $m_4=m_2^2$ for a random variable x gives

$$x^2 - m_2 \overset{a.s.}{=} 0\ . \qquad (3\text{-}10)$$

This means that x takes only the values $\pm\sqrt{m_2}$ with the same probabilities. As all the random variables x_i satisfy (3-10), the most general statistic such that $S_o(x) \overset{a.s.}{=} 0$ under H_o is

$$S_o(x) = \sum_i \alpha_i (x_i^2 - m_2)\ , \qquad (3\text{-}11)$$

where the coefficients α_i are real and at least one of them is non zero.

The singular detection appears if (2-22) is satisfied.

This is especially true if under H_1 the x_i's are continuous random variables, and then the detection problem is singular. But, on the other hand, if under H_1 the x_i's take the value $\pm\sqrt{m_2}$ with finite probability, it becomes possible to have a zero false alarm probability and a detection probability smaller than one simultaneously, which is not strictly a situation of singular detection.

*Laboratoire du CNRS et de l'ESE associé à l'Université de Paris-Sud.

REFERENCES

[1] J.L. LAWSON, and G.E. UHLENBECK, Threshold signals, New York, McGraw-Hill, 1950.

[2] W.A. GARDNER, "A unifying view of second-order measures of quality for signal classification", IEEE Trans. Comm., COM.28, pp.807-816, June 1985.

[3] H.V. POOR, "Robust decision using a distance criterion", IEEE Trans. Inf. Theor. IT 26, pp.575-587, Sept. 1980.

[4] C.R. BAKER, "Optimum quadratic detection of a random vector in Gaussian noise", IEEE Trans. Comm., COM.14, pp.802-805, Dec. 1966.

[5] C. ECKARDT, "Optimal rectifier system for the detection of steady signals", Techn. Rep. S.10, Ref 52-11, Scrips Institute of Oceanography, U. Calif., March 1952.

[6] B. PICINBONO and P. DUVAUT, "Detection and contrast", to be published, Internal L2S Report, 1985.

[7] H.V. POOR and C.I. CHANG, "A reduced complexity quadratic structure for the detection of stochastic signals", J. Acoust. Soc. Amer., 78, pp.1652-1657, Nov. 1985.

[8] W.A. GARDNER, "Structurally constrained receivers for signal detection and estimation", IEEE Trans. Comm., COM.24, pp.578-592, June 1976.

[9] W. GARDNER, "Anomalous behaviour of receiver output SNR as a predictor of signal detection performance exemplified for quadratic receivers and incoherent fading Gaussian channels", IEEE Trans. Inf. Theor., IT-25, pp.743-745, Nov. 1979.

[10] B. PICINBONO, "White noises", in Signal Processing Theories and Application, H.W. Schüssler Ed., EUSIPCO 83, pp.13-16, 1983.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON TRACKING PROPERTIES OF LOCALIZED ESTIMATORS

Maciej NIEDŹWIECKI

Technical University of Gdansk, Institute of Computer Science
ul. Majakowskiego 11/12, Gdansk, Poland

The tracking properties of the localized estimators (weighted least squares or least squares with data windowing) applied to non-stationary system indentification are considered. The concept of the frequency characteristics associated with the localized estimators is introduced and used to derive useful conclusions on the influence of the window shape on tracking properties of the corresponding estimator.

1. INTRODUCTION

Let us consider the problem of identification of a non-stationary dynamic stochastic system described by the following difference equation

$$y(i)=\sum_{j=1}^{p} a_j(i)u(i-j)+n(i) = \alpha^T(i)s(i)+n(i) \qquad (1)$$

In (1) $\{u(i)\}$ is the input sequence (the stationary Gaussian process with exponentially decaying covariance function) and $\{n(i)\}$ is the white noise disturbance; $\alpha(i)=[a_1(i),\ldots,a_p(i)]^T$ and $s(i)=u(i-1),\ldots,u(i-p)]^T$ denote the system parameter and state vectors, respectively.

1.1. Localized estimators

If the system was time-invariant the method of least squares (LS) could be used for the purpose of its identification. For non-stationary systems the LS estimation scheme is usually modified by introducing the special weighting (window) sequence $\{w(i)\}$ allowing the estimation scheme to be localized i.e. time-oriented. Without any loss of generality we can restrict our attention to sequences of the form

$$0\leqslant w(0)\leqslant\ldots\leqslant w(k-1)\leqslant w(k) = 1\geqslant w(k+1)\geqslant\ldots\geqslant 0 \qquad (2)$$

where $k\geqslant 0$. Depending on the application the value of k is set either to 0, resulting in the "one-sided" windows (suitable e.g. for the prediction or control purposes) or to some positive value, resulting in the "two-sided" windows (more suitable for the predictive coding or spectral estimation purposes).

The applied window may be finite or infinite duration; the only requirement is that $\sum_{i=0}^{\infty}w(i)<\infty$, i.e. that the effective length of the sequence $w(i)$ is finite.
There are two ways of utilizing the sequence $\{w(i)\}$ for the purpose of estimation of $\alpha(i)$. First, one can attempt to minimize the weighted sum of squares. Assuming, for convenience, that the infinite observation history of u(i) and y(i) is available at the instant t, the weighted least squares (WLS) estimator $\hat{\alpha}_1(t)$ of $\alpha(t)$ is given by

$$\hat{\alpha}_1(t)=\arg\min_{\alpha}\sum_{i=0}^{\infty}w_1(i)(y(t-i)-\alpha^Ts(t-i))^2 = \left(\sum_{i=0}^{\infty}w_1(i)s(t-i)s^T(t-i)\right)^{-1}\cdot\left(\sum_{i=0}^{\infty}w_1(i)y(t-i)s(t-i)\right) \qquad (3)$$

The second method results when the weighting sequence is applied directly to the data. Let $y_w(t-i)=w_2(i)y(t-i)$, $u_w(t-i)=w_2(i)u(t-i)$ and $s_w(t-i)=[u_w(t-i-1),\ldots,u_w(t-i-p)]^T$. Applying the ordinary least squares procedure to the modified ("windowed") data segment one arrives at the following least squares estimator with data windowing (LSW)

$$\hat{\alpha}_2(t)=\arg\min_{\alpha}\sum_{i=0}^{\infty}(y_w(t-i)-\alpha^Ts_w(t-i))^2 = \left(\sum_{i=0}^{\infty}s_w(t-i)s_w^T(t-i)\right)^{-1}\cdot\left(\sum_{i=0}^{\infty}y_w(t-i)s_w(t-i)\right) \qquad (4)$$

1.2. Relationship between the WLS and LSW estimators

Assuming that the effective length of the weighting sequence $\{w_2(i)\}$ is much greater than the number of identified coefficients one has $w_2(i)\cong w_2(i-1)\cong\ldots\cong w_2(i-p)$ and consequently

$$s_w(t-i)\cong w_2(i)s(t-i) \tag{5}$$

Combining (4) and (5) one arrives at the following approximate relationship

$$\hat{\alpha}_2(t)\cong\Big(\sum_{i=0}^{\infty}w_2^2(i)s(t-i)s^T(t-i)\Big)^{-1}\cdot \cdot\Big(\sum_{i=0}^{\infty}w_2^2(i)y(t-i)s(t-i)\Big)$$

according to which the LSW estimator obtained for the data window $\{w_2(i)\}$ is asymptotically equivalent to the WSL estimator corresponding to the weighting sequence $\{w_2^2(i)\}$. The basic properties of the LSW estimator can be therefore deduced from the corresponding properties of the WLS estimator.

2. TRACKING PROPERTIES OF THE LOCALIZED ESTIMATORS

2.1. Static characteristics

It was shown in [1] that if the parameter vector does not change with time all localized estimators corresponding to the weighting sequences of the same "equivalent length" l_∞ yield the same value of the mean square estimation error - irrespective of the form of the applied windows:

$$V\cong\frac{\rho_o S_o^{-1}}{l_\infty} \tag{6}$$

where $V=\mathrm{cov}(\hat{\alpha}(t))$, $\rho_o=E[n^2(i)]$, $S_o=\mathrm{cov}(s(i))$ and l_∞ denotes the equivalent number of observations

$$l_\infty=\sum_{i=0}^{\infty}(\tilde{w}(i))^2 \tag{7}$$

By $\tilde{w}(i)=w_1(i)/k_\infty=w_2^2(i)/k_\infty$ in (7) we denote the window coefficients normalized with respect to the effective number of observations

$$k_\infty=\sum_{i=0}^{\infty}w_1(i)=\sum_{i=0}^{\infty}w_2^2(i) \tag{8}$$

2.2. Dynamic characteristics

A Associate impulse responses

The window shape may be important factor as far as the tracking capabilities of the localized estimators are concerned. Actually, assuming that the parameter trajectory is uniformly bounded it is possible to show that

$$E[\hat{\alpha}_1(t)]=\sum_{i=0}^{\infty}\tilde{w}(i)\alpha(t-i)+0(\frac{1}{k_\infty}) \tag{9}$$

By modifying the derivation of (9) it is possible to prove that the same relationship holds for the LSW estimafors - the pretty obvious result in the light of the asymptotic equivalence of the WLS and LSW estimators as discussed in section 1.2.
According to (9) the expected path of the parameter estimates can be approximately viewed as an output of a linear causal time-invariant filter of the impulse response $\{\tilde{w}(i)\}$, excited by the process $\{\alpha(i)\}$. The sequence $\{\tilde{w}(i)\}$ may be therefore called the impulse responses associated with the WLS (LSW) estimators.

B Associate frequency characteristics

Following the lines of [2] the frequency response $\tilde{W}(\omega)$ associated with the WLS (LSW) estimator can be defined as the discrete Fourier transform of the impulse response $\{\tilde{w}(t)\}$

$$\tilde{W}(\omega)=\sum_{t=0}^{\infty}\tilde{w}(t)e^{-i\omega t}=A(\omega)e^{i\varphi(\omega)} \tag{10}$$

where $A(\omega)=|\tilde{W}(\omega)|$ and $\varphi(\omega)=\arg W(\omega)$ are the amplitude and phase responses, respectively.
It is straightforward to show that if the weighting (window) sequence satisfies (2), (10) is the frequency response of a low-pass filter. Furthemore, defining the estimation bandwidth B_η of the WLS (LSW) estimator as the η dB bandwidth of the associate filter, i.e. setting $B_\eta=\omega_\eta$ where ω_η denotes the cutoff frequency

$$-20\log A(\omega_\eta)=\eta$$

one can show that irrespective of the value of l_∞ it holds (c.f. [2]).

$$\frac{B_\eta}{v}\cong\mathrm{const} \tag{11}$$

where

$$v=\mathrm{tr}\Big[\inf_t\ \mathrm{cov}(\hat{\alpha}(t))\Big]=\mathrm{tr}[V]$$

(11) is in fact nothing but the precise statement of one of the basic principles of the non-stationary system identification using the localized estimators, namely the principle saying that the choice of the window width is always a matter of compromise between the

"acceptable estimation accuracy" (reflected by v) and "acceptable tracking ability" of the corresponding estimation algorithm (reflected by B_η).

C Importance of the window shape

One of the main advantages of the associate frequency characteristics is that they allow to compare the tracking abilities of the corresponding estimation algorithms. In order to make the problem well-posed one should restrict the analysis to the windows of the same equivalent width, i.e. to windows yielding the same estimation accuracy in the stationary case. It is straightforward to show that for all windows characterized by the same value of l_∞ the total energy of the energy spectrum $|\tilde{W}(\omega)|^2$ is also the same [2] - it is the distribution of this energy over the different frequency bands that depends on the window shape. The problem of determining the influence of the window shape on the properties of the WLS estimators was discused in [2], [3]. It was shown there that the problem may be considered from the two different standpoints. Depending on whether one is interested in following the actual parameter value (the parameter tracking problem) or in matching the system parameter trajectory as a whole (the parameter matching problem) certain window features may be considered either as the advantageous or disadvantageous ones see e.g. [3] for more details. We note that most of the practical applications fall into one of these two categories. Obviously, the general conclusions of [2] and [3] must be valid for the LSW estimators, as well. Some additional points will be discussed in the next section.

3. DATA WINDOWING IN AUTOREGRESSIVE SYSTEM IDENTIFICATION

3.1. Autocorrelation method

Let us consider the autoregressive (AR) process, i.e. process governed by the following equation

$$y(i)=\sum_{j=1}^{p} b_j(i)y(i-j)+n(i) = \beta^T(i)z(i)+n(i) \qquad (12)$$

where $\beta(i)=[b_1(i),\ldots,b_p(i)]^T$, $z(i)=[y(i-1),\ldots,y(i-p)]$ and $\{n(i)\}$ denotes, as before, the white noise sequence. Both the WLS and LSW methods can be utilized for the purpose of estimation of $\beta(i)$. In practice the LSW estimator is used in a slightly modified (so-called autocorrelation) form:

$$\hat{\beta}(t) = \begin{bmatrix} \hat{R}_o(t) & \hat{R}_1(t) & \ldots & \hat{R}_{p-1}(t) \\ \hat{R}_1(t) & \hat{R}_o(t) & \ldots & \hat{R}_{p-2}(t) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \hat{R}_{p-1}(t) & \hat{R}_{p-2}(t) & \ldots & \hat{R}_o(t) \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_1(t) \\ \hat{R}_2(t) \\ \cdot \\ \cdot \\ \cdot \\ \hat{R}_p(t) \end{bmatrix}$$

where

$$\hat{R}_k(t)=\sum_{i=0}^{\infty} y_w(t-i)y_w(t-i-k) \qquad (13)$$

or the large values of the effective window widths the autocorrelation estimator is a negligible modification of the LSW estimator described earier.
The main advantage of the autocorrelation estimate is that it can be evaluated by means of the efficient Levinson-Durbin algorithm.

3.2. Extension of the obtained results to AR processes

According to [1] the static characteristics of the localized estimators such as the effective or equivalent numbers of observations remain valid for AR processes.
The main difficulty generalizing the dynamic characteristics lies in the fact that, unlike for the system governed by (1), the covariance matrix of the state vector $z(i)$ depends on the actual value of the parameter vector $\beta(i)$. Although no simple extension of the obtained results can be expected in such a case, it is clear that some general effects of the window shape on the tracking properties of the autocorrelation estimator should be retained. In particular, it seems that one could risk explaining certain properties of the autocorrelation estimator using the concepts introduced in section 2.

Example

We will comment on the results of the experimental study on adaptive lattice analysis of speech reported by Makhoul and Cosell [4].
The authors consider such hardware implementation of the LPC vocoder which requires estimating the AR coefficients every time sample. In order to reduce the computational load the Barnwell's [5] infinite multi-pole data windows

$$w^{(n)}(i)=\binom{i+n-1}{n-1}\theta^i, 0<\theta<1,\ n\geqslant 1 \qquad (14)$$

were used insted of the usually applied finite windows (the Hamming window is a standard choice in LPC applications). The name "multi-pole windows" comes from the fact that the sequence of weights (14) (as well as certain quantities depending on it, like (13)) can be computed recursively using the n-pole filter $W^{(n)}\theta(z)=1/(1-0z^{-1})^n$.
The main conclusions of [4] are as follows:

1. The human listener prefers windows of a certain equivalent width l_∞ ($\cong$20 ms).

2. For the fixed value of l_∞ the higher-order exponential windows yield the considerably higher speech quality than the first-order one.

Since delay in estimating the "true" parameter trajectory is not a negative phenomenon in LPC systems (the synthesized speech waveform may be the time-shifted version of the original speech signal) this is a typical example of a parameter-matching problem. It is easy to argue that in such a case the maximum degree of linearity of the phase response (for linear phase the phase and group delays are constant) and the maximum degree of concentration of the amplitude response are the most desirable properties of the associate filter.
Figure 1 shows the amplitude and phase characteristics corresponding to the single-pole and two-pole exponential data windows. It is evident from comparison of these plots that the two-pole window is more suitable for the purpose of predictive coding than its single-pole counterpart; the improvement becomes still more significant for the higher-order windows.

Remark

It is the well-known fact that - as far as the autocorrelation method is concerned - the windowing techniques are utilized even if the analysed signal is stationary or locally-stationary (window-width-stationary). This is to reduce the distortions of the LPC coefficients due to the discontinuities in the data at the ends of the observation interval. Depending on the data at the ends of the interval this distortion in the coefficients estimated by the autocorrelation method may or may not be significant. We note that the single-pole exponential window is "one-sided", i.e. tapered at one end, while the multi-pole windows are "two-sided", i.e. tapered at both ends. The use of the multi-pole windows in LPC is therefore advisable for at least two reasons: to improve tracking (matching) properties of the autocorrelation method and to reduce distortions which would appear in the case of utilizing the single-pole window.

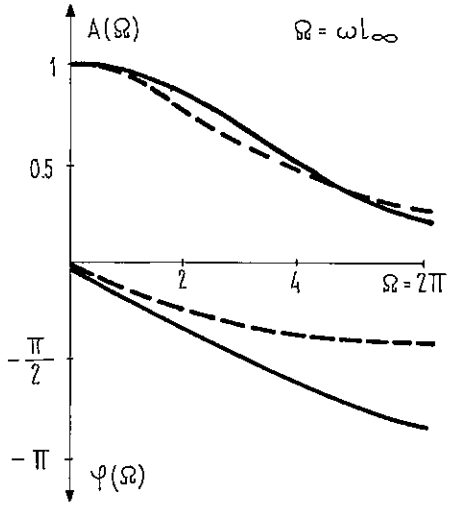


Fig. 1 Comparison of the amplitude and phase responses corresponding to the one-pole (broken line) and two-pole (solid line) exponential data windows.

REFERENCES

[1] Niedźwiecki, M., On the localized estimators and generalized Akaike's criteria, IEEE Trans.Automat.Contr., 11 (1984) 970.

[2] Niedźwiecki, M., On time and frequency characteristics of weighted least squares estimators applied to nonstationary system identification, Proc. 24th IEEE Conf.on Decision and Contr. Fort Lauderdale, FL (1985).

[3] Niedźwiecki, M., Optimization of the window shape in weighted least squares identification of a class of nonstationary systems, Proc. 7th Conf. on Analysis and Optimization of Systems, Antibes, France (1986).

[4] Makhoul, J.I. and Cosell, L.K., Adaptive lattice analysis of speech, IEEE Trans.Circ.and Syst.6 (1981) 494.

[5] Barnwell, T., Recursive autocorrelation computation for LPC analysis Proc. IEEE Conf.ASSP, Hartford, CT (1977).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CODING FOR COMMUNICATION THROUGH MULTIPATH CHANNELS AND APPLICATION TO UNDERWATER CASE

G. HAKIZIMANA, G. JOURDAIN, G. LOUBET

CEPHAG, UA CNRS 346, INPG/IEG, BP 46, 38402 Saint Martin d'Hères Cedex, France

Some codes adapted to communication through multipath channels are presented here. Their periodic and aperiodic correlation parameters and communication properties are given. An underwater acoustic communication experiment has been conducted with these codes. The obtained performances are given in term of error probability. Several reception schemes are presented and compared.

1. INTRODUCTION

Communications through natural environments and particularly the underwater channel are generally disturbed by multipath effects : fading and intersymbol interference. In such a dispersive channel, it is difficult to achieve high data rates. Several techniques to combat the multipath have been proposed update : mainly, equalization and coding spread-spectrum techniques. The latter has shown to be much robust and more efficient, especially when the signal-to-noise ratio is small. This is due to the redundancy introduced by the coding.

Choosing a code involves trade-offs between the communication link reliability and the complexity of the hardware and software required to encode and decode the information.

In the first part of this paper, after recalling the communication model, we present briefly some of well adapted codes especially coset pseudo orthogonal code, Kasami and Boehmer codes. A more complete description will be found in the authors' papers [4,7,2,3]. These codes have been used up to now essentially in ionospheric propagation [4,2].

In a second part, we have conducted an underwater communication experiment using these various codes. Their performances are presented in term of error probability. Several reception schemes are presented and compared.

2. CHANNEL AND COMMUNICATION MODEL

2.1. Channel model

We are concerned here with the following multipath channel model.

Let s(t) be the transmitted signal, r(t) the received one, and b(t) the additive noise.

$$r(t) = \sum_{l=0}^{L-1} a_l \, s(t-t_l) \, e^{j\theta_l} + b(t) \qquad (1)$$

The channel consists of L paths of strenghts a_l, delays t_l and phases θ_l . b(t) is assumed to be white, gaussian and zero-mean.

2.2. Communication system

To encode the data, a N length codeword is associated with k information bits. The codeword is a binary sequence (+1,-1) belonging to a set (code) of K sequences, all with the same length, the same bandwidth, energy and duration T.

The optimal receiver for a single-path channel with additive white gaussian noise is well known and consists of a bank of correlators ; the largest output indicates the received codeword (Fig. 1).

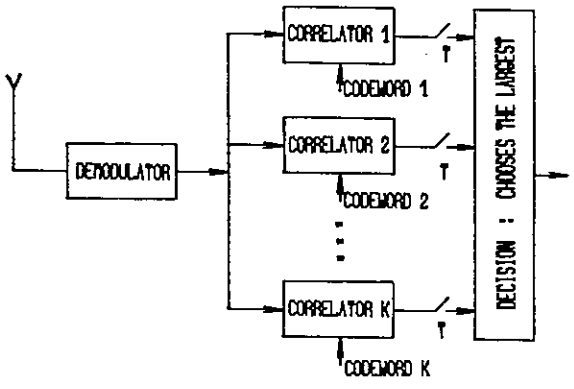


Figure 1 : Receiver

When multipath occurs, this receiver leads to wrong decisions : the desired output is disturbed by partial correlations due to the delayed

signals. Thus, keeping this receiver involves sets of sequences with low autocorrelation sidelobes and also small peak cross correlation magnitude for any delay so that the interference of the delayed codeword versions would be minimized. Hence, the choice of a code will be in a large amount based on its correlation properties.

3. GENERAL PROPERTIES OF CODES

3.1. Periodic and aperiodic correlation

The receiver in Fig. 1 uses aperiodic correlation. But, in order to select optimal codes in the sense defined above, the first step is the use of periodic correlation because it is analytically calculable. The second step will consist in numerical selection of the codes that keep good aperiodic properties among those which had good periodic ones [5].

Let S be a code or set of K sequences of length or period N ; X and Y two sequences in S.

Let $X = \{x_1,\dots,x_N\}$ and $Y = \{y_1,\dots,y_N\}$

The configuration of X (or Y) obtained after l cyclic shifts to the left will be called the phase l of X (or Y).

The periodic correlation between X and Y, Γ^P and the aperiodic one Γ are defined as

$$\Gamma^P_{X,Y}(l) = \sum_{i=1}^{N} x_i\, y_{i+l} \quad ; \quad \Gamma_{X,Y}(l) = \sum_{i=1}^{N-l} x_i\, y_{i+l}$$

For the periodic correlation, the parameters that characterize S are

$$\Gamma^P_a = \operatorname*{Max}_{X\in S} \{ |\Gamma^P_{X,X}(l)| , \quad 1 \leqslant l \leqslant N-1 \} \qquad (2)$$

$$\Gamma^P_i = \operatorname*{Max}_{X,Y\in S} \{ |\Gamma^P_{X,Y}(l)| , \quad 0 \leqslant l \leqslant N-1 \} \qquad (3)$$

$$\Gamma^P_{Max} = \mathrm{Max}\{ \Gamma^P_a, \Gamma^P_i \} \qquad (4)$$

In the aperiodic case, the parameters are Γ_a, Γ_i, Γ_{Max} and are defined in the same way, replacing in (2), (3) and (4) Γ^P by Γ.

Let define R as : $R = \Gamma_{Max} / N \qquad (5)$

3.2. Efficiency

An other important parameter of a code is its rate or efficiency α_R.

$$\alpha_R = k/N \quad (6), \quad \text{where } k = \mathrm{Log}_2 K \qquad (7)$$

K is the total number of N length codewords and k is the number of information bits. Then, the code choice will be based on R and α_R. A good code will provide high α_R and low R.

4. CODES PRESENTATION

4.1. Coset pseudo-orthogonal codes

These codes are described by F. Chavand [4] and are constructed as Gold codes [7].

They consist of 2^k sequences of length $2^k - 1$. It is shown that for these codes

$$\begin{cases} \Gamma^P_{Max} = 2^{\left[\frac{k+2}{2}\right]} \pm 1 \\ \alpha_R = k / 2^k - 1 \end{cases} \qquad (8)$$

where [u] denotes the integer part of the number u.

These codes have been elaborated in order to be "pseudo-orthognal" which means

$$|\Gamma_{XY}(0)| = 1 \qquad \forall\ X \neq Y \qquad (9)$$

4.2. Kasami Code (small set)

This code is described by Sarwate and Pursley [7]. Its construction is based on a pseudo-noise (PN) sequence of length $2^k - 1$ with k even only. It consists of $2^{k/2}$ sequences and achieves

$$\begin{cases} \Gamma^P_{Max} = 2^{k/2} + 1 \\ \alpha_R = (k/2) / 2^{k/2} \end{cases} \qquad (10)$$

4.3. Boehmer code

The two previous codes are linear combinations of PN-sequences and so are of length $2^k - 1$. They are easily generated by binary shift registers. The situation is quite different with Boehmer code [2,3]. Here, the code length N is a prime number.

In power residue classification of a prime, one represents the prime N as $N = ns + 1$.

The N-1 elements $\{1,2,\dots N-1\}$ can be classified in n residue classes of s members each.

Afterwards, in order to make a sequence, one assigns the elements of q classes ($q \leqslant n-1$) to the +1 value (or -1) and the remainders to -1 (or +1).

4.4. Chosen lengths

Chavand [4] has pointed out the 31 length coset code "V45C75 Rot0" of 32 sequences. Its parameters are : $\Gamma^P_{max} = 9$, $\Gamma_{max} = 11$, $R = 11/31$ and $\alpha_R = 5/31$. The author has shown that this code achieves the best trade-off between R and α_R in comparison with the other lengths.

The selected Kasami code is 15 length and is constructed from the primitive polynomial $X^4 + X^3 + 1$ (31 in octal polynomial notation). It consists of 4 sequences and is characterized by $\Gamma^P_{max} = 5$, $\Gamma_{max} = 5$, $R = 5/15$, $\alpha_R = 2/15$.

There exists only one phase, called reference phase, with k '1' as k first bits for a PN-sequence of 2^k-1 length. For this Kasami small set, k=4 and we used the generating PN-sequence in the phase corresponding to the reference shifted one time to the left. Hence, Γ_{max} was numerically obtained with the phases (5,12,1,4) for the 4 sequences.

For the Boehmer code, the selected lenght is N = 13 = 4.3+1. There are 4 classes of 3 elements each. We formed 4 sequences corresponding to the following residue class selection :

$$(0,1,2)\ ,\ (0,1,3)\ ,\ (0,2,3)\ ,\ (1,3)$$

When these sequences are shifted in order to be respectively in the phases 0,3,3,9, we obtain a Boehmer code with the following parameter $\Gamma^P_{max} = 5$, $\Gamma_{max} = 5$, R = 5/13, $\alpha_R = 2/13$.

All the selected codes are optimal with respect to the lower bounds for periodic correlation parameters of binary sequences sets [7]. Finally, we remark that the 3 codes have the same R and α_R.

In a correlation scheme, the longer is the sequence the best is the protection against noise. Thus, the selected length for the pseudo-orthogonal code will provide a good protection against noise. On the other side, the Kasami and Boehmer code will provide simplifications of required hardware and software because of their small number of sequences. The following table resumes the important parameters.

	N	K	R	α_R
Chavand	31	32	0.35	0.16
Kasami	15	4	0.33	0.13
Boehmer	13	4	0.38	0.15

Table 1

5. THE UNDERWATER COMMUNICATION LINK

In order to test the properties of these codes, we have conducted an experiment of underwater communication in a lake. The receiver was 13.5 m far from the emitter. Both were 4 m deep.

The signals were BPSK modulated with a 5 Khz carrier. The BPSK signal occupied 1.25 Khz (half width of the main lobe). The noise was generated by a white gaussian zero-mean noise generator and hence we could control Signal-to-Noise Ratio. Two SNR have been tested : 8dB and -2dB. They were measured at the receiver input. The receiver signals were recorded on a magnetic tape and processed in laboratory.

5.1 Channel identification

The channel identification shows that there are 3 paths whose strenghts and delays are stable. The mean relative strengths of the other paths to the main one are : 0.7, 0.4, and the mean delays are : 1.6 ms, 3.2 ms. This means that interference exists up to 4 binary digits of each codeword. On the other hand, we do not give phases values because we assume that these are unknown or random (uniformly distributed over $(0,2\Pi)$).

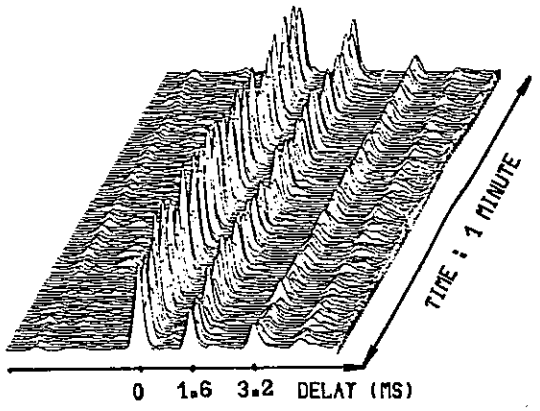


Figure 2
Square envelope of the channel impulse response.

5.2 Preprocessing of the received signals.

We performed a non-coherent reception scheme. It is constructed with complex demodulation, matched filter and envelope squarer. It is well known that this is the optimal processing of a signal whose phase is unknown or random.

6. RECEIVERS

We are interested in three kinds of receivers which correspond to different presumed channel knowledge.

6.1. Single-path receiver

This receiver works only with the first signal arrival. It would be optimal if one deals with a single-path channel.

6.2. Main path receiver

Here we assume there must be 2 paths and the delay between them is assumed known. The receiver performs preprocessing of § 5.2 for each path and chooses the main energy output.

6.3. Path energy combining receiver

In this third case, we suppose that the path energies and delays are known, the paths are resolvable [8] and the phases are uniformly distributed over $(0,2\Pi)$. Then, if the sample of the k^{th} output envelope at delay t_l is x_{kl}, the optimal combining of the samples is given in [8]

$$w_k = \sum_{l=0}^{L-1} \log_e I_o\left(\frac{2a_l x_{kl}}{N_o}\right) \qquad (11)$$

where I_o is a bessel function and N_o is the channel noise power density.

With approximations on I_o, this receiver is easily implemented with transversal filters after the preprocessing of Section 5.

7. RESULTS

We have sent 10^4 codewords for each code. The three receivers of Section 6 have been numerically implemented. The beginning of codewords has been identified by correlation.

7.1. Error probability

The obtained error probabilities are given in Table 2. The error probability at 8 db is less than 10^{-4} because no error was done over all transmitted codewords. For receiver 1, the error probability is in good agreement with theoretical results given in [4]. The difference between the 3 receivers has been exhibited only for the first code. A little improvement was got by using all the channel paths. Surely, this is due to the fact that the second and third paths are small relatively to the first one.

SNR	- 2 db			8 db
RECEIVERS	1	2	3	1
PSEUDO (CHAVAND)	10^{-1}	$8\ 10^{-2}$	$4\ 10^{-2}$	$< 10^{-4}$
KASAMI	$1.9\ 10^{-1}$			$< 10^{-4}$
BOEHMER	$2\ 10^{-1}$			$< 10^{-4}$

Table 2 : Error probabilities

7.2. Data rate

The full data rate was 1250 bits/s but the real information rate was only $1250 \times \alpha_R \simeq 188$ bits/sec. Such data rate meets general specifications of underwater communications [1].

For comparison, we note that with a -2 db SNR and a 1.25 Khz bandwidth, the maximum bit rate is given by the Shannon formula which leads to 885 bits/sec.

8. CONCLUSION

This first experiment shows that a coded underwater communication is feasible. The major information here lies upon the fact that for a relatively high SNR ($\simeq 8$ db) all the selected codes are good. So, it will be more interesting to use short codes as Kasami or Boehmer rather than long ones because of hardware or software simplifications. On the other hand, at low SNR ($\simeq 0$) only long codes are acceptable.

An other kind of code has been described by Yates and Holgate [9] and Potter [6]. This code is very promising because it has good communication properties and leads to a very simplified receiver.

ACKNOWLEDGEMENTS

This work has been partly supporte d by the Direction of the French Naval Constructions.

REFERENCES

[1] BAGGEROER A.B., Acoustic telemetry. An overview, IEEE J. of Ocean. Engin., OE-9, n° 4, 229-235, 1984.
[2] BOEHMER A.M., Binary pulse compression codes, IEEE Trans. Inf. Theory, IT-13, 156-167, 1967.
[3] CHAKRABARTI N.B. and TOMLINSON M., Design of sequences with specified autocorrelation and crosscorrelation, IEEE Trans. Comm., COM-24, 1246-1252, 1976.
[4] CHAVAND F., Transmission d'information dans les canaux multitrajets à caractéristiques aléatoires par codage pseudo-orthogonal. Application au canal ionosphérique, Thèse d'Etat Paris-Sud, ORSAY, 1981.
[5] HAKIZIMANA G., Etude de famille de séquences binaires, Rapport CEPHAG n° 44/85.
[6] POTTER J.M., Recursive code generation based on m-sequence, Electronic Letters, Vol. 16, n° 22, 858-859, 1980.
[7] SARWATE D. and PURSLEY M.B., Crosscorrelation properties of pseudo random and related sequences, Proc. IEEE, Vol. 68, n° 5, 593-619, 1980.
[8] TURIN G.L., Introduction to spead-spectrum antimultipath techniques and their application to urban digital radio, Proc. IEEE, Vol. 68, n° 3, 328-353, 1980.
[9] YATES K.W. and HOLGATE D.J., Code modulation of m-sequence, Electronic Letters, Vol. 15, 836-838, 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ALL DIGITAL IMPLEMENTATION OF A RECEIVER FOR BANDWIDTH EFFICIENT COMMUNICATION

M. Oerder, G. Ascheid, R. Häb, H. Meyr

Aachen Technical University (RWTH)
Templergraben 55, D-5100 Aachen, W. Germany

By means of the maximum likelihood principle, algorithms can be derived for optimal detection of the data in a given received signal as well as for the clock and carrier phase synchronization that is needed for the detection. The signal processing that is necessary to implement these algorithms is very complex, but the potential of digital integrated circuits make such signal processing feasible today. A 2 MBit/s prototype of such a receiver has been built. We describe the receiver, laying the emphasis on the clock and carrier phase synchronization.

1. INTRODUCTION

In this paper a fully digital receiver for bandwidth efficient digital modulation is described. We first explain the structure of the receiver. Then the development from a mathematical criterion of optimality to a digital circuit is described based on the example of the clock and carrier phase synchronization of the receiver. In section 3 the maximum likelihood estimation algorithm for clock and carrier phase synchronization is presented. Section 4 deals with the various restrictions that a practical realization imposes on the algorithm, and section 5 gives an overview of the actual hardware realization. The last sections deal briefly with the other parts of the receiver. The symbol detector is described in a companion paper [3].

2. RECEIVER STRUCTURE

We consider a general received signal in complex baseband representation

$$r(t) = s(t-\varepsilon_o T,\underline{a})\, e^{j\Theta_o} + n(t) \qquad (1)$$

with

- $s(t,\underline{a})$: transmitted signal, modulated by the symbol sequence $\underline{a}$
- T : symbol clock period
- ε_o : unknown clock phase
- Θ_o : unknown carrier phase
- $n(t)$: noise

For the detection of the symbol sequence $\underline{a}$ estimates of clock and carrier phases are needed. On the other hand, the optimal estimation of the synchronization parameters requires a knowledge of the symbols.

Since in general clock and carrier phase vary only very slowly, a decision directed receiver structure (fig. 1) can be used. For the symbol detection, the received signal is corrected by estimates $\tilde{\varepsilon}$ and $\tilde{\Theta}$ to yield

$$r_c(t) = r(t+\tilde{\varepsilon}T)\, e^{-j\tilde{\Theta}}$$

$$= s(t-\varepsilon_o T+\tilde{\varepsilon}T,\underline{a})\, e^{j(\Theta_o-\tilde{\Theta})} + \tilde{n}(t)$$

$$\approx s(t,\underline{a}) + \tilde{n}(t) \qquad (2)$$

where $\tilde{n}(t)$ is a noise signal with the same statistical properties as $n(t)$. The estimates $\tilde{\varepsilon}$ and $\tilde{\Theta}$ are generated by a prediction filter from estimates $\hat{\varepsilon}$ and $\hat{\Theta}$ that are estimated by using the already detected part of the symbol sequence $\underline{a}$.

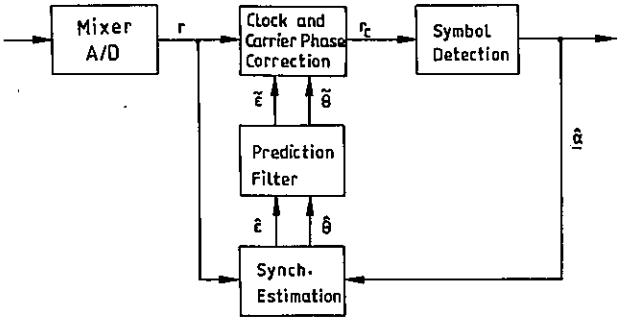


Fig. 1: Decision-directed receiver structure

The advantage of this structure is that mixer and sampling oscillators can operate at fixed rates, and all the synchronization takes place fully digitally. Feedback from the digital part of the receiver to the analog stages to control oscillators is not necessary.

3. MAXIMUM LIKELIHOOD ESTIMATION OF CLOCK AND CARRIER PHASE

For the estimation of the synchronization

parameters the received signal is divided into segments with length MT and for each segment estimates $\hat{\varepsilon}$ and $\hat{\Theta}$ are computed. M must be small enough to allow the assumption of constant clock and carrier phase in each segment. On the other hand, the larger M is, the lower the rate is at which the estimates are generated and have to be processed by the prediction filter.

The maximum likelihood estimates of clock phase ε_o and carrier phase Θ_o result from maximizing the likelihood function

$$L(\varepsilon,\Theta) = \exp\left[-\frac{1}{N_o} \int_{mMT}^{(m+1)MT} \left| r(t)-s(t-\varepsilon T,\underline{\alpha})\, e^{j\Theta} \right|^2 dt \right] \quad (3)$$

To simplify the notation we let m=0 from now on. As suggested in the previous section, we use the detected sequence $\hat{\underline{\alpha}}$ instead of the unknown sequence $\underline{\alpha}$.

For signals with constant envelope we have

$$\int_0^{MT} \left| s(t-\varepsilon T,\hat{\underline{\alpha}})\, e^{j\Theta} \right|^2 dt + f(\varepsilon,\Theta) \quad (4)$$

Therefore, instead of $L(\varepsilon,\Theta)$, we can maximize the reduced loglikelihood function

$$\lambda(\varepsilon,\Theta) = \int_0^{MT} \left[r(t)\, [s(t-\varepsilon T,\hat{\underline{\alpha}})e^{j\Theta}]^* + r^*(t)\, s(t-\varepsilon T,\hat{\underline{\alpha}})e^{j\Theta} \right] dt \quad (5)$$

It can be shown [1] that by using the expressions

$$K_c(\varepsilon) = \mathrm{Re}\left\{ \int_0^{MT} r(t)\, s^*(t-\varepsilon T,\hat{\underline{\alpha}})\, dt \right\} \quad (6)$$

$$K_s(\varepsilon) = \mathrm{Im}\left\{ \int_0^{MT} r(t)\, s^*(t-\varepsilon T,\hat{\underline{\alpha}})\, dt \right\} \quad (7)$$

λ can be written as

$$\lambda(\varepsilon,\Theta) = K_s(\varepsilon)\sin\Theta + K_c(\varepsilon)\cos\Theta = \sqrt{K_s^2(\varepsilon)+K_c^2(\varepsilon)}\,\cos(\Theta-\arctan(K_s/K_c)) \quad (8)$$

Thus the clock phase can be estimated independently of the carrier phase by maximizing

$$K(\varepsilon) = K_s^2(\varepsilon) + K_c^2(\varepsilon) \quad (9)$$

and subsquently the estimate $\hat{\Theta}$ can be computed as

$$\hat{\Theta} = \arctan(K_s(\hat{\varepsilon})/K_c(\hat{\varepsilon})) \quad (10)$$

These are the optimal estimates (in the maximum likelihood sense) that can be obtained from a given signal segment without a-priori knowledge. In the receiver however the filtered estimates $\tilde{\varepsilon}$ and $\tilde{\Theta}$ are available. These are the best estimates given all the preceding signal segments, and not the one under present consideration. Because of the slow change of ε_o, the filtered estimate $\tilde{\varepsilon}$ can be used for computing the carrier phase estimate:

$$\hat{\Theta} = \arctan(K_s(\tilde{\varepsilon})/K_c(\tilde{\varepsilon})) \quad (11)$$

4. DIGITAL IMPLEMENTATION OF THE ESTIMATOR FOR CLOCK AND CARRIER PHASE

For the digital realization of the clock and carrier phase estimation described in the previous section, quantizations and simplifications must be made. They will now be described.

4.1. Resolution and range of the estimates

To enable fast acquisition of the clock phase, a zero tracking algorithm is not used to find the maximum of the likelihood function, but rather a parallel estimation algorithm [1] is used as follows: the loglikelihood function is computed for N_p values of ε, namely

$$\varepsilon = \frac{k}{N_p} \qquad k \text{ integer}, \quad \frac{-N_p}{2} \le k < \frac{N_p}{2} \quad (12)$$

So the estimate resolution is $1/N_p$ and the estimation range is -0.5...0.5. A resolution with N_p=16 was shown to be sufficient for the symbol detector [1]. A larger range is unnecessary, because a larger clock phase leads to the symbol sequence being detected correspondingly shifted by the symbol detector and thus results in a automatically shifted reference signal $s(t-\varepsilon T,\hat{\underline{\alpha}})$. So the resulting clock phase is again between -0.5 and 0.5.

For the carrier phase estimate $\hat{\Theta}$ a 6-bit resolution was found to be sufficient. The maximum quantization error is then 6^o. For high signal-to-noise ratio (SNR), such an error does not effect the symbol detection. For low SNR, the quantization error is reduced by the filtering of the estimates and negligible compared to the error variance induced by the noise.

4.2. Signal sampling and quantization

The quadrature components of the received signal are generated in a mixer and sampled at rate N/T.

$$r_I(n) = \mathrm{Re}\{ r(nN/T) \} \quad (13)$$

$$r_Q(n) = \mathrm{Im}\{ r(nN/T) \} \quad (14)$$

Expressions (6) and (7) are computed as

$$K_c(\varepsilon) \approx \sum_{n=0}^{NM-1} \left[r_I(n) s_I(n,\varepsilon,\hat{\underline{\alpha}}) + r_Q(n) s_Q(n,\varepsilon,\hat{\underline{\alpha}}) \right] \quad (15)$$

$$K_s(\varepsilon) \approx \sum_{n=0}^{NM-1} \left[r_I(n) s_Q(n,\varepsilon,\hat{\underline{\alpha}}) - r_Q(n) s_I(n,\varepsilon,\hat{\underline{\alpha}}) \right] \quad (16)$$

with $s_I(n,\varepsilon,\hat{\underline{\alpha}}) = \mathrm{Re}\{s(nN/T-\varepsilon T,\hat{\underline{\alpha}})\}$ (17)

$$s_Q(n,\varepsilon,\hat{\underline{\alpha}}) = \mathrm{Im}\{s(nN/T-\varepsilon T,\hat{\underline{\alpha}})\} \quad (18)$$

The error caused by this approximation depends on the sampling rate. N=4 was found to introduce only small errors, while for smaller N the error was too large.

A 3 to 4 bit resolution of the quadrature components of the received signal is sufficient so that the increase in estimate variance compared to the unquantized case is negligible. Because of practical considerations (available components), the receiver prototype was built having a 4 bit resolution.

4.3 Reference signal generation

The reference signals s_I, s_Q are generated through using the already detected symbols. For linear modulation schemes like MPSK (M-ary phase shift keying) or QAM (quadrature amplitude modulation) the transmitted signal can be written as

$$s(t,\underline{\alpha}) = \sum_k \alpha_k \, g(t-kT) \quad (19)$$

with α_k : transmitted symbol in interval k
$g(t)$: signal pulse
T : length of symbol interval

For unfiltered 8PSK modulation, the symbols are

$$\alpha_k = e^{jk\pi/4}, \quad 0 \le k \le 7 \quad (20)$$

and the signal pulses are rect-pulses of duration T

If the signal is filtered, $g_T(t)$ is of infinite duration. So the reference signal can be constructed correctly only if all symbols are known. This would mean intolerable delay and hardware complexity. Therefore the reference signal must be approximated. In the prototype receiver pulses are used that depend on two symbols only. It could be shown that estimation with these reference pulses is only slightly worse than estimation with exact reference pulses.

The fact that the reference signals do not have a constant envelope, as assumed in eq. (4), does not substantially impair the estimation either.

For CPM (continuous phase modulation), a similar way of generating the reference signals is possible.

5. HARDWARE REALIZATION

A hardware prototype was built to demonstrate the feasibility of the algorithms for high data rates. To achieve high speed, the processing was parallelized and pipelined as much as possible.

In fig. 2 a block diagram of the clock and carrier phase estimator is shown. The metric $K_s^2+K_c^2$ is computed in 16 parallel branches for $\varepsilon = -8/16 \ldots 7/16$. Each branch has a local reference signal generator, multipliers, adders and accumulators for the computation of K_s and K_c and a circuit for the computation of the sum of the squares. The multipliers have 4-bit inputs and 8-bit outputs and thus can easily be implemented even at high speeds. The accumulator outputs are reduced to 7 bits without a significant loss in accuracy.

The receiver prototype was built with standard TTL components. Each of the parallel branches occupies one half of a double Euroformat board (233mm x 160mm). By pipelining the computations we were able to obtain a processing time of 250 ns per sample of the received signal. With N=4

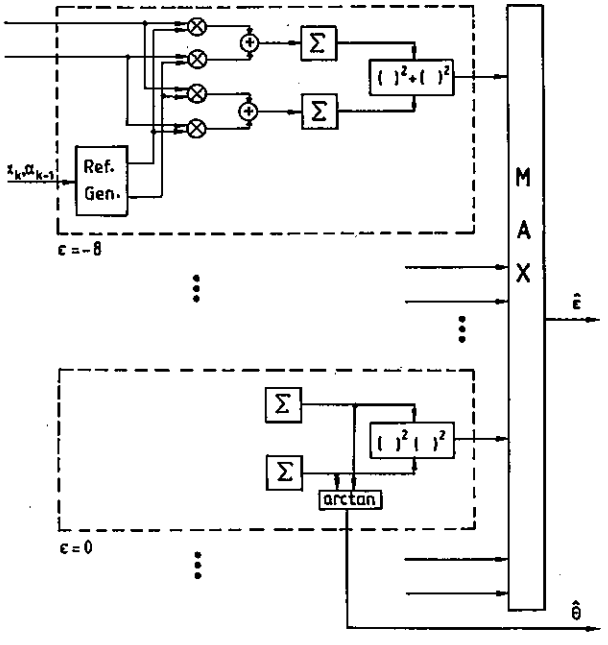


Fig. 2: Estimation of clock and carrier phase

this is a rate of 10^6 symbols per second. By

using faster components, this rate could of course be enlarged.

The center branch also contains the computation of $\hat{\theta}$ from K_s and K_c. A maximum selector determines $\hat{\varepsilon}$ from the results of the 16 branches.

6. FILTERING OF THE ESTIMATES

The estimates $\hat{\varepsilon}$, $\hat{\theta}$ emerge from the estimator at a rate 1/MT. This is a factor 1/MN slower than the rate at which the estimator itself must operate. So the filtering can be done by much slower circuitry. We chose a TMS320 signal processor for the task.

The estimates $\hat{\varepsilon}$, $\hat{\theta}$ are filtered by Kalman filters with stored coefficients. Thereby optimal estimates $\tilde{\varepsilon}$, $\tilde{\theta}$ can be determined. This is important in the acquisition phase. For $\hat{\varepsilon}$ a first order filter is used and for $\hat{\theta}$ a second order filter, which allows for a frequency offset between receiver and transmitter, is used.

7. CLOCK AND CARRIER PHASE CORRECTION AND SYMBOL DETECTION

According to eq. (2), the received signal r(t) must be filtered by a filter with transfer function

$$H(f) = e^{j2\pi\tilde{\varepsilon}Tf}\, e^{-j\tilde{\theta}} \qquad (21)$$

to yield the corrected signal $r_c(t)$.

The rotation $e^{-j\tilde{\theta}}$ can be performed separately by a memoryless transformation. The clock phase is corrected by a transversal filter with adjustable coefficients. For linear modulation as in eq. (19) this filter can also take over the task of a matched filter with transfer function $G^*(f)$. By switching the coefficients, in our prototype, 8 different transfer functions

$$H(f) = G^*(f)\, e^{-j2\pi\varepsilon_i Tf} \qquad \varepsilon_i = -4/16 \ldots 3/16$$

can be realized. The filter works at a sampling rate of 2/T which is above the Nyquist rate for usual (double sided) bandwidths of 1.2/T ... 1.6/T. At the output of the filter only one sample per symbol is passed on. Because of matched filtering and correct shift ε_i this sample is the best one for the symbol detection.

A special logic ensures that symbol clock delays of more than T/2 are correctly handled by choosing the appropriate samples at the filter output.

The symbol detector is designed for trellis coded 8PSK modulation. The code is rotationally invariant [2], so phase ambiguities do not impair the carrier phase synchronization. The symbols are decoded by a Viterbi decoder [3].

8. CONCLUSION

Using maximum likelihood technique algorithms for detection and synchronization suitable for digital implementation were derived. The structure of the receiver based on these algorithms is fundamentally different from the classical receiver structure. For example, clock and carrier synchronization is achieved with a maximum of 4 samples per symbol. This is far lower than would be possible by employing a digital PLL (which is a direct block by block conversion from the analog to the digital domain). To achieve the necessary computational throughput, the architecture of the receiver takes advantage of massive parallel processing and pipelining. The architecture is well suited for a VLSI realization which is presently under work.

REFERENCES

[1] G. Ascheid, H. Meyr, "Maximum likelihood detektion and synchronisation by parallel digital processing," IEEE GLOBECOM 84 conference record, Atlanta, vol. 3, pp. 1068-1072.

[2] M. Oerder, "Rotationally invariant trellis codes for MPSK modulation," IEEE ICC 85 conference record, Chicago, vol. 2, pp. 552-556.

[3] J. Stahl, H. Meyr, M. Oerder, "Implementation of a high speed Viterbi decoder," this volume

ACKNOWLEDGEMENT

The support of the Deutsche Forschungsgemeinschaft (DFG) under contract no. Me 651/4 is gratefully acknowledged.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DESIGN OF A DEMULTIPLEXER FOR A REGENERATIVE SATELLITE

Enrico DEL RE, Romano FANTACCI

Dipartimento di Ingegneria Elettronica, Università di Firenze,
Via S. Marta, 3 - 50139 Florence, Italy.
ph. (55) 4796285 tlx. 572460 UNIFI I

In this paper the design of a demultiplexer for the on-board interfacing of FDMA and TDMA links in a regenerative satellite is presented. In particular, we focus here only on per-channel structure based on the analytic signal method that allows a high modular and flexible implementation. A theoretical analysis and computer simulation are reported in order to evaluate the performance degradation due to the finite arithmetic implementation of the demultiplexer.

1. INTRODUCTION

The development of space communications has required a new generation of satellites. In the past, satellites have operated using analog modulation of the carrier, and accessing the satellite was accomplished with frequency-division multiple access (FDMA). The satellite simply translated the carrier frequency and retransmitted the signal in a wide beam covering a large geographic area. Today the new systems employ time-division multiple access (TDMA), new efficient modulation techniques, multiple beam antennas, and on-board processing for a higher system efficiency.

The on-board signal processing offers several advantages to satellite communication systems. A typical and interesting feature is the separation of the uplinks and downlinks, thus allowing their separate and independent optimization. For example, in many applications, such as mobile communication services, the use of uplink FDMA techniques (with the inherent low-cost earth stations) and downlink TDMA techniques (that can fully exploit the satellite transponder output power without intermodulation) is an attractive solution. However, the feasibility of this approach depends on efficient means of translating between the two multiple access formats on board the satellite. The on-board system implementation complexity (including the VLSI design) and power consumption are, of course, of primary concern.

The on-board processing system receives an input FDMA signal and supplies an output TDMA signal; therefore, it must accomplish the functions of the separation of each individual channel and of its demodulation. An appropriate name for the on-board processing system is the 'multicarrier demodulator' (MCD). Two main functions are implemented by a MCD: formerly the demultiplexing and consequently the demodulation. We focus here only on the digital demultiplexer (DEMUX), that is a signal processor for translating signals from FDM to TDM formats. The demodulation operation, following the DEMUX, can be performed by any of the available implementation of digital demodulators.

The DEMUX design has been carried out by assuming the QPSK modulation, because it is well known that this modulation technique permits the achievement of a good trade-off between bandwidth and performance degradation. However, the DEMUX design can also be carried out for different modulation techniques suitable for satellite communications such as, for example, the MSK modulation. For the digital DEMUX design two basic approaches exist: the block methods [1] and the non-block methods [2]. The block methods require a FFT block processor while this is not used for the non-block methods. We focus here only on the non-block methods that allow a high modular and flexible implementation structure. The DEMUX design has been carried out in order to reduce the overall implementation complexity and includes the finite precision design. It has been carried out in particular aiming at its possible implementation by means of custom VLSI digital circuits.

2. ANALYTIC SIGNAL APPROACH FOR THE DEMUX

We consider the analytic signal approach for the DEMUX implementation [2]. This approach is a per-channel method that avoids any digital product modulator and any block processor. It has the specific feature to relax the filter specifications, thus achieving a lower implementation complexity with respect to other per-channel approaches. Further, the analytic signal approach directly leads to a per-channel and high modular structure; this structure is directly matched to the per-channel implementation of the demodulators. Therefore, a certain degree of integration of the DEMUX and DEMOD functions is conceivable. Another advantage of the analytic signal approach is its high flexibility: differently from the other methods, in the case that some specific applications should benefit from the unequal channel bandwidth, the analytic signal structure could vary on demand the bandwidth

Work performed under European Space Agency Contract ESTEC 6096/84/NL/GM(SC).

assigned to each channel, simply by switching to a suitable new set of DEMUX parameters. The principle of operation of the analytic signal method is illustrated in [2] and will be briefly recalled in the following.

The structure of the DEMUX according to the analytic signal method is shown in Fig. 1 [2],[4]. The FDM input signal is sampled, according to the sampling theorem [3], at the high-rate frequency $f_u = 1/T_u$ (uplink) and processed in order to obtain N_c TDM digital signals, each sampled at the low-rate frequency $f_d = 1/T_d$ (downlink), N_c being the number of multiplexed channels. In Fig. 1, $H_i(fT_u)$, $H_i'(fT_u)$ represent the conjugate symmetric and antisymmetric parts, respectively, of the high-rate complex bandpass filter $\bar{H}_i(fT_u)$ which can be regarded as a frequency translated version of a low-pass prototype $\bar{H}(fT_u)$ such that [2]:

$$(1)\quad \bar{H}_i(fT_u)=H_i(fT_u)+jH'(fT_u)=\bar{H}[2\pi(f-iW-W/2)T_u]$$

where W is the channel spacing. In the same figure, $G_i(fT_d)$ and $G_i'(fT_d)$ represent the conjugate symmetric and antisymmetric parts, respectively, of the complex low-rate filter $\bar{G}_i(fT_d)$ which can be defined as [2]:

$$(2)\quad \bar{G}_i(fT_d)=G_i(fT_d)+jG_i'(fT_d)=\bar{G}\{[f-(-1)^i W/2]T_d\}$$

Thus, each filter $\bar{G}_i(fT_d)$ is related, according to eq. (2), to a low-pass prototype. It can be noted from eq. (2) that the number of different filters $G(fT_d)$ is actually two: one for the odd channels and the other for the even channels. Taking into account eqs. (2) and (3), we have in the frequency domain [2]:

$$(3)\quad X_i(fT_d)=S(fT_d+i\pi)[H_i(fT_d+i\pi)\cdot G_i(fT_d+i\pi) - H_i'(fT_d+i\pi)\cdot G_i'(fT_d+i\pi)]/N_c$$

according to the implementation structure shown in Fig. 1. It must be noted that a decimation factor equal to the number N_c of multiplexed channels is involved.

3. FINITE ARITHMETIC IMPLEMENTATION EFFECTS

The implementation of a digital signal processing system necessarily requires a finite arithmetic. Although it is possible to conceive and actually implement floating-point arithmetic for digital signal processing systems, however it is deemed that the fixed-point arithmetic implementation will still represent the more convenient solution in the near to medium term. Thus, we consider here only the effects of a fixed-point finite arithmetic implementation.

The error sources derived from the finite length of the digital registers are: a) quantization of the input signal; b) quantization of the filter coefficients; and c) rounding of the multiplication operations. For the first source of error the sampled input signal is quantized in amplitude in order to be represented by a set of numbers in binary form. We suppose that the input signal will be modeled as a random Gaussian signal. This assumption comes from the consideration that the input FDMA signal is the sum of several independent signals. Under this hypothesis the signal-to-quantization noise ratio SNR_q can be expressed in dB as [3]:

$$(4)\quad (SNR)_{dB} = 6.02\ b_q - 7.27\ dB$$

where b_q is the number of bits employed for the quantization of the input signal. Assuming that the dynamic range of the analog-to-digital converter (A/D) is in the range ± 1, we suppose that an Automatic Gain Control (AGC) is used in order to constrain the output signal (input to A/D) within the range ± 1. Further, we shall assume that the output signal from any filter is in the range ± 1. This can be guaranteed by a suitable scaling of the digital filter coefficients (included in the filter design and implementation).

For the second source of error the minimum wordlengths of the filter coefficients are determined by computer rounding in order to guarantee that they still verify the required filtering specifications. For the third source of error it must be observed that a FIR implementation is the most suitable one for the digital filters of Fig. 1. A FIR filter implemented by P multiplications each rounded to b_m bits produces an output noise error with mean power equal to that introduced by an output quantization to b_a bits, according to:

$$(5)\quad P\,\frac{2^{-2b_m}}{3} = \frac{2^{-2b_a}}{3}$$

Let us suppose to have determined (through analytic or simulation tools) the number of bits b_a required for the output signal quantization to achieve some specified performance, then the number of bits b_m for the multiplication roundings inside the filter is determined as:

$$(6)\quad b_m = b_a + \lceil (\log_2 P)/2 \rceil$$

where $\lceil x \rceil$ denotes the minimum integer greater than or equal to x.

The block diagram of the DEMUX according to the analytic signal approach and including the multiplication rounding model previously described is reported in Fig. 2. In this figure, S_i denotes the power of the input FDM signal assumed uniformly distributed among N_c channels, N_i is the mean power of the noise introduced in the uplink and N_q is the quantization noise power due to the input A/D conversion, both supposed white, Gaussian and uniformly distributed among the N_c channels. The term $S_t/2$ represents the power of the signals at the output of the filters $H_i(fT_u)$, $H_i'(fT_u)$ and also at the output of the filters $G(fT_d)$, $G'(fT_d)$ under the assumption that they are of the all-pass type. In the same figure, S_t is the power of the signal at the i-th output of the DEMUX, N_t is the overall noise power for each DEMUX output which will be defined in the following, N_{a_1} denotes the noise

power due to the finite arithmetic implementation of the filters $H_i(fT_u)$, $H_i^!(fT_u)$ due to the quantization of their outputs at b_{a1} bits. In the same way, N_{a2} represents the power of the noise introduced by the finite arithmetic implementation of the filters $G_i(fT_d)$, $G_i^!(fT_d)$.

In order to evaluate the signal-to-noise ratio SNR_t at each DEMUX output, in addition to the contributions previously considered, the effects of the decimation process must also be included. The decimation process gives rise to a noise contribution at each DEMUX output independent of the other disturbances with mean power given by [4]:

$$(7) \quad N_d = S_i \, \delta_2^2$$

where δ_2^2 is the maximum acceptable squared out-of-band ripple and assuming $N_c >> 1$. It can be noted that eq. (7) is derived according to a worst-case analysis because we have assumed the out-of-band ripple constant in the filtering bandwidth and equal to its maximum values δ_2.

Now, under the hypothesis that the filters $G_i(fT_d)$, $G_i^!(fT_d)$ are of the all-pass type, by setting $N_a = N_{a1} + N_{a2}$ and assuming $N_{a1} = N_{a2}$ (equal quantization bits at the output of the high-rate and low-rate digital filters), the overall noise power N_t at any output of the DEMUX is given by:

$$(8) \quad N_t = \frac{N_i}{N_c} + \frac{N_q}{N_c} + S_i \, \delta_2^2 + 2 N_a$$

Thus, the signal-to-noise ratio at each DEMUX output is [4]:

$$(9) \quad \frac{1}{SNR_t} = \frac{1}{S_t/N_t} = \frac{1}{S_i/N_i} + \frac{1}{S_i/N_q} + \frac{2}{S_i/N_a} + N_c \, \delta_2^2$$

where we have assumed $S_t = \frac{S_i}{N_c}$.

Thus, the finite arithmetic wordlengths at each point of the DEMUX structure must be determined in order to introduce an overall degradation with respect to the input signal-to-noise ratio smaller than a specified value.

4. DESIGN RESULTS

In this section, the design results of the demultiplexer according to the analytic signal methods are presented for a QPSK modulation technique with a data rate R = 2.048 Mb/s. The number of input channels N_c is assumed equal to 8 and the filtering bandwidth B equal to 572.3 [4]. The channel spacing W is chosen equal to 1.536 MHz in order to minimize the overall numbers of multiplications required per second and per channel, and to guarantee an integer number of samples per symbol (3 samples/symbol). Under these assumptions, according to the procedure described in [4], the filtering specifications are the following:

- maximum acceptable in band-ripple: $\delta_1 = 4.84 \cdot 10^{-3}$ (max. freq. amplitude 0.042 dB);
- minimum out-of-band attenuation: $\delta_2 = 6.31 \cdot 10^{-3}$ (-44 dB).

It must be pointed out that these specifications represent the overall filtering requirements. Thus, the high-rate and low-rate low-pass prototypes have been designed according to different specifications [4]: their cascade must satisfy the overall filtering specifications reported herein.

The design of these digital filters has been carried out by using the Equiripple method [5]. The filtering specifications are satisfied by employing FIR digital filters with a number of coefficients equal to 35 and 23 for the high-rate and low-rate low-pass prototypes, respectively. The overall number of multiplications required per channel and per second [4] is thus equal to 124.42 Mmults/s/ch. The finite arithmetic design is carried out according to that outlined in sect. 3. The number of bits used for the quantization of the filter coefficients is derived through a computer rounding in order to still verify the filter coefficients. The other finite arithmetic wordlengths are chosen, according to eq. (9), in order to introduce an overall degradation less than 0.2 dB with respect to the input signal-to-noise ratio which guarantees an ideal bit-error rate equal to 10^{-9}.

The finite arithmetic wordlengths that result for the demultiplexer design are reported in Tab. 1. The results obtained through the theoretical analysis explained in sect. 3 and the computer simulation of the demultiplexer are reported in Tab. 2. In this table the input signal-to-noise ratio SNR_i is derived taking into account the equivalent noise bandwidth B_n=1.454 MHz of the actually designed filters, and the signal-to-noise ratio SNR_t is evaluated at each output of the DEMUX. In Fig. 3 the degradations introduced by the finite arithmetic implementation are reported as a function of the input signal energy to noise density ratio. It can be noted that the results obtained through computer simulation (simulation results) show a satisfactory agreement with those obtained through the theoretical analysis (analytic results), that represents a worst-case analysis.

5. CONCLUSIONS

In this paper a complete digital DEMUX design suitable for on-board interfacing FDMA and TDMA links has been presented. We have focused only on the analytic signal approach as this design method leads to a per-channel and high modular structure; the analytic signal approach has the advantage of a high flexibility, i.e. the resulting structure could vary on demand the bandwidth assigned to each channel simply by switching to a suitable new set of system parameters, and of more relaxed finite arithmetic wordlengths with respect to the other DEMUX approaches (i.e. block methods). In conclusion, the DEMUX design described herein represents an appropriate solution for the on-board processing system interfacing FDMA and TDMA links; in particular, it has been carried out aiming at its possible implementation by means of custom VLSI digital circuits.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the cooperation of ITALSPAZIO, Rome, Italy, and of P.L. Emiliani of I.R.O.E. in carrying out the research activities of the ESTEC contract. They also wish to thank G. Pennoni and W. Greiner of ESTEC for helpful discussions.

REFERENCES

[1] Bellanger, M.G., and Daguet, J.L., TDM-FDM Transmultiplexer: Digital Polyphase and FFT, IEEE Trans. Comm., vol. COM-22, Sept. 1974.

[2] Del Re, E., and Emiliani, P.L., An Analytic Signal Approach for Transmultiplexers: Theory and Design, IEEE Trans. Comm., vol. COM-30, July 1982.

[3] Bellanger, M., Digital Processing of Signals. Theory and Practice, J. Wiley&Sons, London 1984.

[4] ESTEC Contract 6096/84/NL/GM(SC), Multicarrier Demodulator Design, Final Report, 1986.

[5] Programs for Digital Signal Processing, ASSP Digital Signal Processing Committee, Ed., IEEE Press, N.Y., 1979.

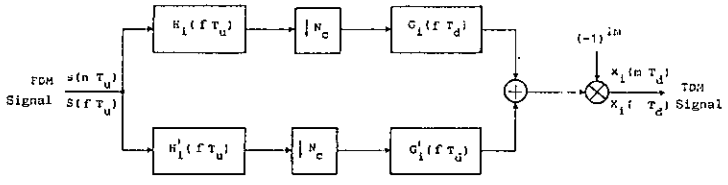


Fig. 1

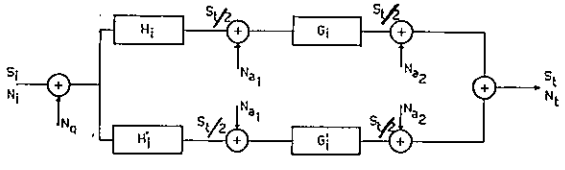


Fig. 2

INPUT SIGNAL QUANTIZATION	FILTERS $\bar{H}(fT_u)$			FILTERS $\bar{G}(fT_d)$		
b_q	b_c	b_m	b_a	b_c	b_m	b_a
8	12	11	8	11	11	8

TAB. 1 - 2.048 Mb/s Finite Arithmetic Design. Analytic Signal Approach.
b_q - Input signal wordlength.
b_c - Filter coefficient wordlength.
b_m - Filter arithmetic wordlength.
b_a - Output filter wordlength.

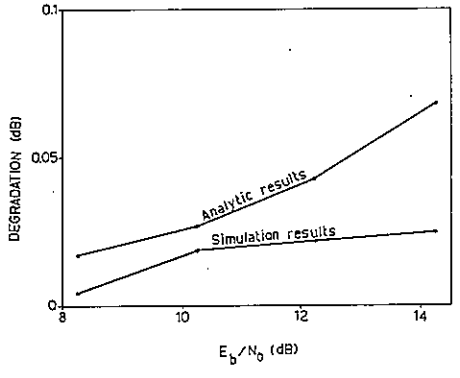


Fig. 3

E_b/N_0 [dB]	8.25		10.25		12.25		14.25	
	A	S	A	S	A	S	A	S
SNR_i [dB] (in the DEMUX noise filter bandwidth)	9.74	9.846	11.74	11.846	13.74	13.846	15.74	15.846
SNR_t [dB] (at the DEMUX output)	9.723	9.842	11.713	11.827	13.697	13.825	15.672	15.821
DEGRADATION [dB]	0.017	0.004	0.027	0.019	0.043	0.021	0.068	0.025

TAB. 2 - 2.048 Mb/s Finite Arithmetic Design.
A - Analytic results.
S - Simulation results.
B_n - 1.454 MHz (noise filter bandwidth).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

OPTIMUM SEQUENTIAL SIGNALLING AND DECISION TECHNIQUES FOR FEEDBACK COMMUNICATION SYSTEMS

Giuliano BENELLI

Dipartimento di Ingegneria Elettronica, Università diFirenze,
via S.Marta 3, Firenze, Italy.

In this paper the integration of the modulation operation and of the signaling and decision structure in oder to reduce the error probability in a communication system using a feedback channel is analyzed. Continous-phase-frequency modulations are in particular considered. The performance of the communication systems, determined by the Euclidean distance, are optimized as a function of the modulation index and of the mean transmission number of each symbol.

1. INTRODUCTION

In many communication systems it is possible to use an essentially noiseless feedback channel to improve the communications over a noisy feedback channel. The noiseless feedback channel permits generally a reduction in the amount of the signal energy and in the complexity of coding and decoding operations, required to achieve specified performance. Typical examples are the sequential decision schemes /1/,/2/, and the Automatic-Repeat-Request (ARQ) techniques /3/,/4/, using error detecting codes. In these communication systems, when the forward channel introduces an uncorrectable error patterns, the same information is retransmitted sometimes, until a correct reception of the information is detected. The error probability in all these communication systems depends on the Euclidean distance and/or the Hamming distance. The Euclidean distance is introduced during the modulation operation, while the Hamming distance depends on the channel coding scheme.

In this paper it is shown that the integration of the modulation operation in the signaling and decision structure of the feedback communication systems permits to reduce significantly the error probability. In classical feedback communication systems the signals associated to successive transmissions of the same symbol are all the same. In the scheme descibed in this paper the modulated waveforms corresponding to successive transmissions of the same symbol are different and are chosen in such a way to increase the Euclidean distance between the signals. The proposed scheme can be applied both to sequential decision feedback schemes and ARQ schemes. Continous-Phase-Frequency-Shift-Keying (CPFSK) molations, which are very attractive for their low bandwidth occupancy with respect to classical digital modulations, are in particular analyzed.

2. CPKSK MODULATIONS AND EUCLIDEAN DISTANCE FOR SEQUENTIAL SIGNALING

In a classical sequential signaling scheme each block of k informative symbols, coming out from the source, is encoded in a codeword, n_1 symbols long, of a code C of type (n_1,k). The code is assumed able to correct t errors and to detect s errors. In the sequential communication system described in this paper, each block of k information symbols is encodded in a codeword of n n_1 symbols, as shown in Fig.1. The first $(n-n_1)$ symbols, called the phase continuity symbols, are redundant symbols and are introduced in order to achieve a phase zero at the beginning of the informative part of the codeword. The successive k symbols are the informative symbols, while the last (n_1-k) symbols are the redundant symbols of the code C.

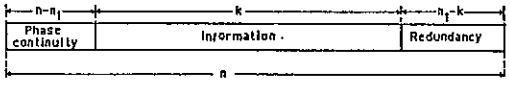


Fig.1 - Structure of the transmitted codewords.

Let us consider the transmission of a codeword $\underline{c}' = c_i$ of the code C, where c_i denotes the i-th component $\underline{c}'$. When this codeword is transmitted the first time, $\underline{c}'$ is encode in a codeword $\underline{c}_1 = c_{1,i}$, having length n, according to the format in Fig.1, and such that $c_{1,i+n+n_1} = c_i$ for $1 \le i \le n_1$. The j-th transmission of the codeword $\underline{c}'$ is decoted with $\underline{c}_j = c_{j,i}$ for $1 \le i \le n$. The index j is introduced because $\underline{c}_j$ can be different from $\underline{c}_k$ for j = k. Before its transmission, each symbol $c_{j,i}$ is sent to a CPFSK modulator which associates to this symbol a waveform $s_{j,i}(t)$ given by /5/ :

$$(1) \quad s_{j,i}(t) = \sqrt{\frac{2E}{T}} \cos\left[\omega_o t + \frac{ht}{T} c_{j,i} + x_i + \psi_j\right]$$

where T is the time-signaling interval, E the signal energy, $f_o = \omega_o/2$ the carrier frequency, h the modulation index, ψ_j the a phase term, constant during the transmission of a codeword, which will be defined later, and x_i is a phase-term introduced in order to maintain the phase continuity at the end and at the beginning of the time-signaling intervals, which is defined by :

$$(2) \quad x_i = x_{i-1} + (c_{j,i-1} - c_{j,i}) \ (\pi/2)$$

The phase paths in the CPFSK modulation can be represented through the phase trellis. The phase at the beginning of a codeword can assume anyone of the possible 2/h values, depending on the particular codeword transmitted. The phase ψ_j, constant during the j-th transmission of $\underline{c}_j$, is defined by :

$$(3) \quad \psi_j = (j-1)\, h$$

The first $(n-n_1)$ symbols of $\underline{c}_j$ are introduced in order to set the phase at the beginning of the first informative interval equal to ψ_j. The number of phase continuity symbols is $n-n_1 \le i \le 1 + (1/h)$. Infact, starting from any phase or state in the the phase zero can be achieved in 1/h intervals. The introduction of the phase ψ_j has as a consequence that the phases of successive transmissions of a symbol of $\underline{c}'$ differ for ψ_j, i.e. if $\vartheta_{j,i}(t)$ denotes the phase during the j-th transmission of the i-th symbol, then it results:

$$(4) \quad \vartheta_{j,i}(t) = \vartheta_{j-1,i}(t) + \psi_j$$

The transmission of the last n_1 symbols of the codeword, denoted as informative and redundant symbols, is considered. When a codeword is transmitted for the first time, these symbols are equal to $\underline{c}'$. For the successive transmissions, $c_{j,i}$ can be different from $c_{k,i}$ for k=j.

The sequences of the binary values assumed by $c_{j,i}$ for $n-n_1+1 \le i \le n$, in successive transmissions is denoted with $\underline{a}$:

$$(5) \quad \underline{a} = (c_{1,i}, c_{2,i}, c_{3,i}, \ldots\ldots)$$

If c_i' is equal to -1, then $\underline{a}$ is assumed equal to a prefixed sequence $\underline{a}_{-1}$, while if c_i' is equal to 1 then $\underline{a} = \underline{a}_1$. As an example, if the sequences $\underline{a}_{-1}$ and $\underline{a}_1$ are given by :

$$(6) \quad \begin{cases} \underline{a}_{-1} = (-1,-1,-1,\ldots\ldots) \\ \underline{a}_1 = (1,1,1,\ldots\ldots) \end{cases}$$

the phase trellis corresponding to the phases of successive transmissions of $c_{j,i}$ is shown in Fig.2 for a CPFSK modulation with h=0.5. On the other hand, if $\underline{a}_{-1}$ and $\underline{a}_1$ are shown as :

$$(7) \quad \begin{cases} \underline{a}_{-1} = (-1,-1,1,-1,1,-1,\ldots\ldots) \\ \underline{a}_1 = (1,-1,1,-1,1,-1,\ldots\ldots) \end{cases}$$

the phase trellis for h=0.5 is shown in Fig.3. As it can be seen from these figures the Euclidean distance depends significantli on the two sequences $\underline{a}_{-1}$ and $\underline{a}_1$ and increases significantly with j. In this way the error probability in the demodulation of the i-th symbol is lower with respect to the classical system, in which the mean error probability is the same in all the transmission of a codeword. Therefore it is quite important the choice of the two sequences $\underline{a}_{-1}$ and $\underline{a}_1$.

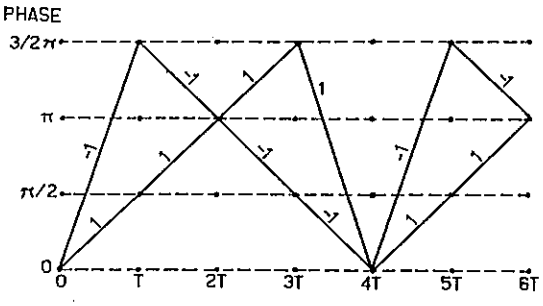


Fig.2 - Phase trellis of the sequences (6) for a CPFSK modulation with h=0.5

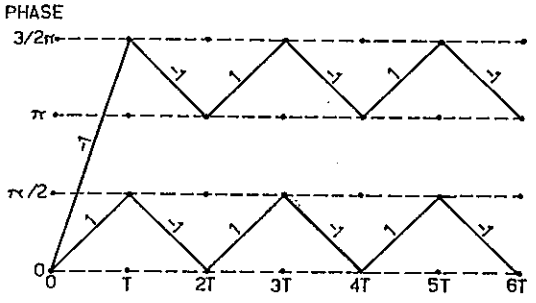


Fig.3 - Phase trellis of the sequences (7) for a CPFSK modulation with h=0.5.

3.OPTIMATION OF THE EUCLIDEAN DISTANCE AND RESULTS

The Euclidean distance and,therefore , the error probability of the communication systems described in this paper depend on the sequences $\underline{a}_{-1}$ and $\underline{a}_{1}$.In many cases,as in classical feedback communication systems,the two sequences $\underline{a}_{-1}$ and $\underline{a}_{1}$ are equal to the sequences given by (6).The Euclidean distance between these two sequences is shown in Fig.4 as a function of the modulation index.Moreover,for some modulation index value ,it can be found other sequences having higher Euclidean distance than sequences (6).As an example,Fig.5 shows the Euclidean distance between sequences (7) as a function of the modulation

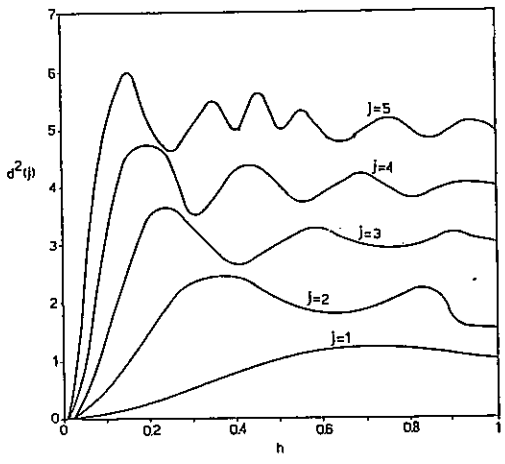


Fig.4.Euclidean distance between sequences (6).

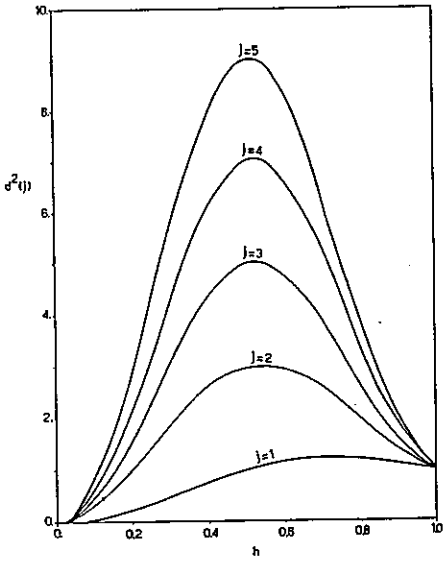


Fig.5 - Euclidean distance between sequences (7).

index. It can be seen that for some values of h these sequences present higher Euclidean distance with respect to sequences (6).

Fig.6 shows the highest value of the Euclidean distance, which can be achieved for each modulation index. These values are obtained by considering all the sequences having length j and choosing the two sequences for which the Euclidean distance $d^2(j)$ is maximum. Table 1 shows, for some values of h, the two sequences $\underline{a}_{-1}$ and $\underline{a}_1$ which give the higher $d^2(j)$. These sequences are represented by integer numbers, whose binary representation is equal to the desired sequence.

MODULATION INDEX h	NUMBER OF TRANSMISSIONS J				
	2	3	4	5	6
0.1	0-3	0-7	0-15	0-31	0-31
0.2	0-3	0-7	7-8	8-23	23-40
0.3	0-3	3-4	4-11	11-20	20-43
0.4	0-1	4-5	0-1	4-5	4-5
0.5	0-1	4-5	4-5	4-5	4-5
0.6	0-1	4-5	4-5	4-5	4-5
0.7	0-1	4-5	0-3	0-3	0-3
0.8	1-2	0-3	0-3	0-3	0-3
0;9	1-2	1-7	0-7	0-15	0-15

Table 1 - Sequences which give the higher Euclidean distance.

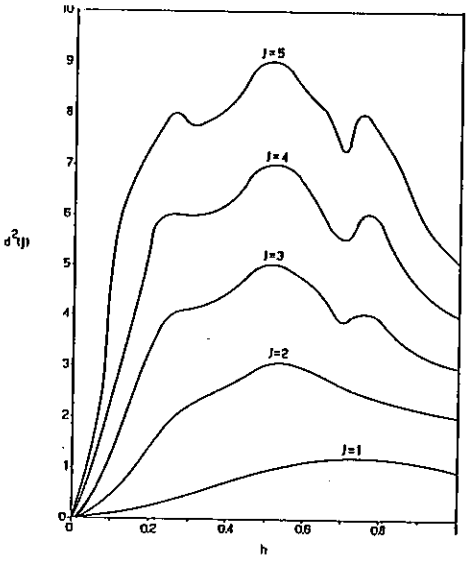


Fig.6 - Maximum Euclidean distance for different j.

REFERENCES

/1/ A.J.Viterbi, "The Effect of Sequential Decision Feedback on Communication over the Gaussian Channel", IEEE Trans. on Inf.Theory, p.81-92, 1965.

/2/ R.W.Lucky, "A Survey of the Communication Theory Literature", IEEE Tran. on Inform.Theory, p.725-739, 1973.

/3/ H.O.Burton, D.D.Sullivan, "Errors and Error Control", Proc. IEEE, vol.60, p.1293-1303, 1972.

/4/ G.Benelli, "An ARQ Scheme with Memory and Soft-Error Detectors", IEEE Trans.on Communications, p.285-288, 1985.

/5/ T.Aulin, C.E.Sundberg, "Continous-Phase Modulation - Part I: Full Response Signaling", IEEE Trans. on Communications, p.196-209, 1981.

Tutorial on DIGITAL PROCESSING FOR COMMUNICATIONS.

H. Meyr
RWTH Aachen
Aachen
West Germany

PAPER NOT AVAILABLE.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DIGITAL SIGNAL PROCESSING IN A COMMERCIAL SHORT WAVE RECEIVER

- A Preliminary Study -

Friedrich JONDRAL
AEG Aktiengesellschaft, Fachbereich Empfänger und Peiler
Sedanstraße 10, D-7900 Ulm (Donau)
Federal Republic of Germany

The present paper describes a preliminary study towards a commercial short wave receiver, which performs the main selection and demodulation processes by digital signal processing.

1. INTRODUCTION

After the appearance of single chip processors - like NEC 7720 or TMS 320 - on the market, the development of a short wave receiver (3 MHz,...,30 MHz), which performs the main selection and the demodulation of the received signals by the help of such processors, became feasible.

In the present paper a preliminary study towards such a receiver is described from a signal processing point of view. As important constraints of our development, the analog components of the receiver are to be considered, since they determine the operating conditions for the signal processor.

The high frequency (HF) unit, especially comprising the local oscillator and a crystal filter, is taken from a commercial HF receiver. In the HF unit the signal is mixed to an intermediate frequency (IF) and bandpass filtered by the crystal filter, which is in some sense taken as the anti-aliasing filter for the digital signal processing. In front of the analog-to-digital converter (ADC), the signal is amplified in such a manner, that its effective value is two bits below the operating limit of the ADC.

The analog-to-digital conversion is performed at the IF, taking into account the sampling theorem applied to bandpass signals (bandpass subsampling). It is a desired effect of the bandpass subsampling, that the signal is converted to a lower IF by this procedure.

First of all the signal processor performs a complex mixing of the digital signal, converting the intermediate frequency to zero now, i.e. the receiver uses the quadrature principle. All main selection filters (six different bandwidths are available) are realized as a FIR filter followed by an IIR filter. The FIR filter is mainly used to reduce the sampling frequency, while the IIR filter, consisting of second order sections in coupled structure, is delivering the steepness required by the desired selectivity. The demodulation processes for the various possible types of modulated signals are also realized as programs on a single chip processor.

The processor used for filtering and demodulation essentially consists of three single chip processors mounted on a standard circuit board.

After demodulation the signal is either directly transfered to an information destination or it is transformed to an audio signal by digital-to-analog conversion.

Figure 1 shows a block diagram of a short wave receiver with digital signal processing.

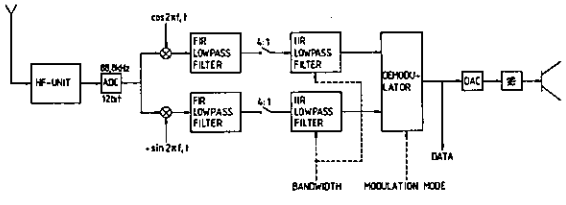


Figure 1

The starting point of the development discussed in this paper is the AEG receiver E1800 /1/, from which also the HF unit is taken.

The tolerance schemes of the desired main selection filters are given in table 1.

filter	passband kHz	stopband kHz
1	0.01	0.3
2	0.15	0.5
3	0.3	0.9
4	0.75	1.75
5	1.5	2.0
6	3.0	4.5

Table 1 (passband ripple $\leq$ 1 dB)

THEBYCHEFF filters of type I are used, since their amplitude characteristics decrease monotonicly in the stopband (desired stopband attenuation: 60 dB). The demodulator has to process signals of the following types:

A1A, A1B	CW telegraphy,
A2A, A2B	MCW telegraphy,
A3E	telephony,
R3E, H3E, J3E	SSB telephony,
F1B, F1C	two frequency shift keying (FSK2)

2. ANALOG-TO-DIGITAL CONVERSION AND TRANSFORMATION TO THE BASEBAND

Every radio signal with sinusoidal carrier may be written in the following form:

$$s(t) = 4a(t)\cos\{\omega_c t+\omega(t)t+\theta(t)\} \qquad (1)$$

The parameters of the signal s(t) are the amplitude a(t), the instantaneous information circular frequency $\omega(t) = 2\pi f(t)$, the zero phase $\theta(t)$, and the carrier circular frequency $\omega_c = 2\pi f_c$.

Arriving from the antenna the signal reaches the HF unit, which first of all performs a (real) mixing with the mixing signal $\cos\{\omega_c t-\tilde{\omega}_i t+\theta_i\}$, where θ_i may be taken to be zero without loss of generality. The result of this procedure is bandpass filtered by a crystal filter. Therefore we get as IF signal:

$$\tilde{s}_i(t) = 2a(t)\cos\{\tilde{\omega}_i t+\omega(t)t+\theta(t)\} \qquad (2)$$

The preselection crystal filter mentioned above is taken as the anti - aliasing filter for the succeeding digital signal processing. The amplitude and phase characteristics of this filter are drawn in figure 2.

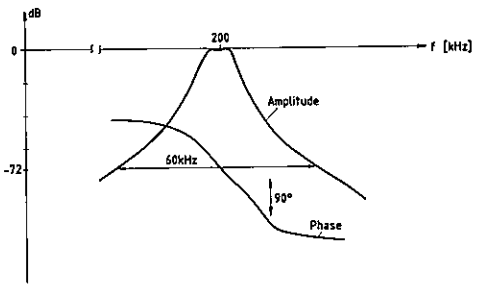


Figure 2

Since an ADC with 12 bit resolution is used, it is enough to determine the sampling frequency in such a manner, that aliasing effects only occur in frequency regions, which are attenuated by the crystal filter by more than 72 dB. The 72 dB attenuation frequencies are located at 170 kHz respectively at 230 kHz; i.e. the attenuation bandwidth is Δf_a = 60 kHz.

At $\tilde{f}_i$ a bandpass subsampling of the signal is performed. Since the signal information is contained in a relatively small frequency region around the IF $\tilde{f}_i$, we want to carry out the signal processing symmetrically relative to $\tilde{f}_i$. This supposition requires, that (after sampling) $\frac{1}{2}(n+\frac{1}{2})$ periods of the discrete spectrum must be located between the frequencies 0 and $\tilde{f}_i$. Therefore the possible sampling frequencies are:

$$f_s(n) = \frac{2\tilde{f}_i}{n + \frac{1}{2}} \qquad (3)$$

The frequency region of width Δf_u around the centre frequency $\tilde{f}_i$, which is not aliased after sampling, is determined from

$$\Delta f_u = \min\{\Delta f_a;\ f_s(n) - \Delta f_a\} \qquad (4)$$

Of course, $f_s(n) > \Delta f_a$ is required.

We choose n=4 and together with $\tilde{f}_i$ = 200 kHz, Δf_a=60 kHz, we get the sampling frequency f_s=88.8 kHz and an aliasing free region of width Δf_u = 28.9 kHz around the centre frequency. Since n=4 is an even number, the period of the discrete spectrum, that is located between -44.4 kHz and 44.4 kHz is in regular position.

The effect of bandpass subsampling is sketched in figure 3. The signal spec-

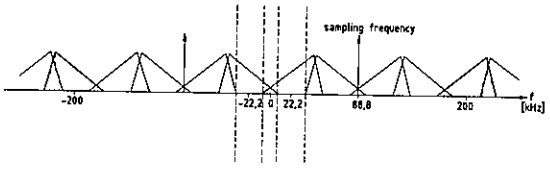


Figure 3

trum is continued periodically with period f_s. It is important, that the unambiguity interval contains an aliasing free region (-36.7 kHz, -7.8 kHz) ∪ (7.8 kHz, 36.7 kHz), which comprises the complete spectrum of the signal s(t). The image of the intermediate frequency in the unambiguity interval is f_i = 22.2 kHz.

The subsampling procedure has transformed the signal into a 12 bit data sequence $s_i(n\Delta t)$ of $88.8 \cdot 10^3$ sampling values per second:

$$s_i(n\Delta t) = 2a(n\Delta t) \cdot \cos\{\omega_i n\Delta t + \omega(n\Delta t) n\Delta t + \theta(n\Delta t)\}$$
$$t = f_s^{-1}, \quad n \in Z \qquad (5)$$

During the next step the complex discrete signal is computed from the real data sequence. This is performed by complex mixing of the real discrete signal (5) with the mixing frequency f_M = -22.2 kHz = $-f_i$. Afterwards the signal is passed through a linear phase FIR filter. This gives

$$\underline{s}(n\Delta t) = a(n\Delta t) \cdot e^{j\{\omega(n\Delta t) n\Delta t + \theta(n\Delta t)\}}, \quad j = \sqrt{-1} \qquad (6)$$

The FIR lowpass is designed in such a manner, that aliasing effects cannot occur within the passband of the succeeding IIR filters. The data rate at the output of the FIR filter may be reduced by a factor of 4; i.e. the data rate at the FIR filter output is $22.2 \cdot 10^3$ complex 16 bit values per second.

3. PERFORMANCE OF THE MAIN SELECTION

The digital signal (6), where Δt is now $4.5 \cdot 10^{-5}$ s, has to be lowpass filtered by a sharp recursive filter. The IIR filters, that were to be realized, had to meet the requirements of table 1. From this table it may be seen, that filter 5 requires the highest expense for its realization among all the six filters.

A first rough design starts from the point, that the filters should be realized as a cascade of second order

sections. The second order sections are represented in a canonic form. Because of the limitations of the coefficient and data wordlengths to 13 respectively 16 bits, a realization in canonic form seemed to be unprofitable. Therefore the second order sections are represented in the so called coupled structure (c.f. /2/, p. 190). Moreover the zeros of the IIR filter transfer functions are not taken into account for reasons of computing expense.

The block diagram of the recursive filter no. 5 as well as a rough sketch of the amplitude characteristic of the whole filter 5 are shown in figure 4.

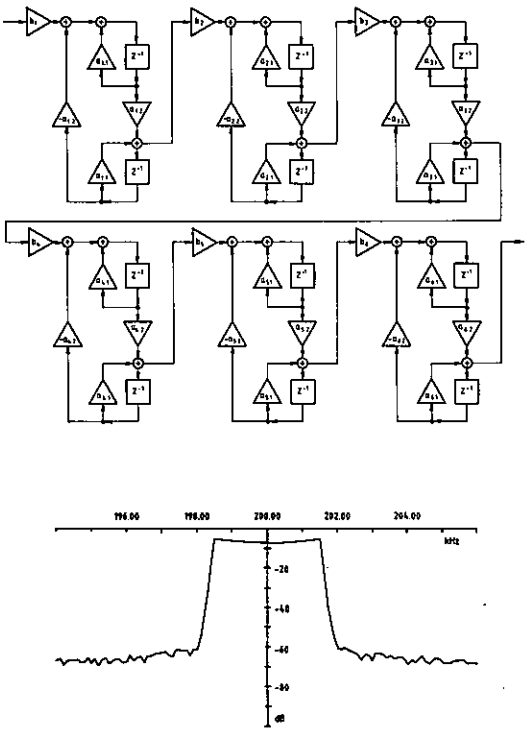


Figure 4

The filter design was performed with the help of standard design programs /3/, which were used to determine the second order sections in canonic form. The canonic representations were then transformed into the coupled structure.

4. DEMODULATION

The demodulation of amplitude modulated signals (A1A, A1B, A2A, A2B, A3E) is done by computing the magnitude of (6).

In the case of SSB signals, the local oscillator has to be mistuned by half the bandwidth B, since only one SSB filter is available. After filtering, this mistuning has to be undone. Therefore we have to apply the following transformation to an USB signal (6):

$$s(n\Delta t) = \mathrm{Re}\{\underline{s}(n\Delta t)\} \cdot \cos(\pi B n\Delta t) - \mathrm{Im}\{\underline{s}(n\Delta t)\} \cdot \sin(\pi B n\Delta t) \qquad (7)$$

For a LSB signal the frequency has to be translated by -B/2.

F1B and F1C signals may be considered as special cases of SSB signals as far as audio demodulation is concerned. For obtaining the data content of a F1B or a F1C signal a special FSK demodulation procedure is necessary.

5. REFERENCES

/1/ AEG: Allwellenempfänger E1800, Datenblatt

/2/ AZIZI, S.A.: Entwurf und Realisierung digitaler Filter, München 1981: R. Oldenbourg Verlag

/3/ IEEE ASSP: Programs for Digital Signal Processing, New York 1979: IEEE Press

/4/ RABINER, L.R.; GOLD, B.: Theory and Application of Digital Signal Processing, Englewood Cliffs (New Jersey) 1975: Prentice Hall Inc.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

THE USE OF SIGNAL PROCESSORS FOR SIMULATING DATA CIRCUITS

K. Herberger

Wandel & Goltermann GmbH & Co
Mühleweg 5, Postfach 45
D-7412 Eningen u. A., Federal Republic of Germany

A device for simulating a data circuit is described which, using digital signal processing algorithms, allows compliance with all the items required in CCITT Recommendation V.56. These cover both linear and non-linear distortions of any analog telephone line chosen at random. Three TMS32010 signal processors are used to implement the digital algorithms.

1. INTRODUCTION

Data circuit measuring instruments provide quality information and facilitate localisation of fault sources for reliable data transmission via telephone lines. Digital signal processing is particularly suitable for checking the correct function of these data circuit measuring instruments or for providing standards when developing new measuring instruments.

CCITT Recommendation V.56 (COMPARATIVE TESTS OF MODEMS FOR USE OVER TELEPHONE-TYPE CIRCUITS) describes both the linear and the non-linear characteristics of any analog telephone line chosen at random, a two-wire circuit being split up into a four-wire circuit via hybrid circuits.

A simulator can be looped-in in both directions to simulate the faults, distortions and fluctuations of the telephone network in a defined, controllable manner.

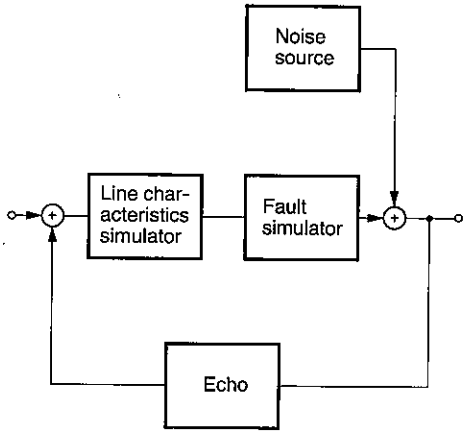


Fig. 1: Data circuit simulator

The requirements of CCITT Recommendation V.56 can be fulfilled completely with the aid of signal processors.

Simulation of the linear circuit characteristics is explained in the next section. This is followed by a description of the non-linear characteristics.

2. LINEAR CIRCUIT CHARACTERISTICS

2.1. CCITT requirements

The definition of linear circuit characteristics in the CCITT Recommendation covers both attenuation and group delay distortions, these in turn being subdivided into "symmetric" and "asymmetric" distortions. The distortion is obtained from specified attentuation and group delay values at given frequencies.

Symmetric attenuation distortion in dB

f/Hz	Mode 1	Mode 2
300	6	12
500	3	8
800	1	2
1600	0	0
2500	unspecified	8
2800	3	unspecified
3000	6	12

Symmetric group delay distortion in ms

f/Hz	Mode 1	Mode 2
500	3	4.5
600	1.5	3
1000	0.5	1.5
1800	0	0
2600	0.5	1.5
2800	3	3
2900	unspecified	4

Table 1: Typical line distortions

All distortions are weighted with factors between 0.1 and 1.6 in increments of 0.1, this producing a very fine graduation.

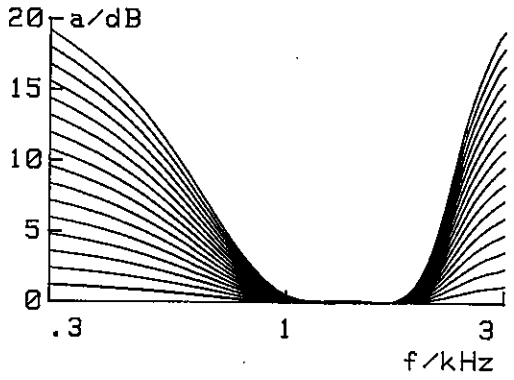


Fig. 2: Attenuation distortions weighted with k = 0.1 to 1.6

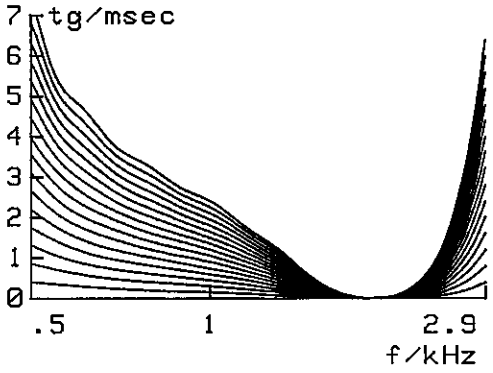


Fig. 3: Group delay distortions weighted with k = 0.1 to 1.6

2.2. Filter design

It is desirable to be able to adjust the attenuation and group delay distortions independently.

It is possible to simulate group delay distortions by using all-pass filters with a maximum order of 20; in this way, changing the group delay does not alter the value of attenuation distortion which has been set.

The problem lies in producing the attenuation distortions. It is important that the group delay distortions of the peripherals (i.e. input- and output transformers, anti-aliasing- and output low-pass filters) should be equalised at the same time.

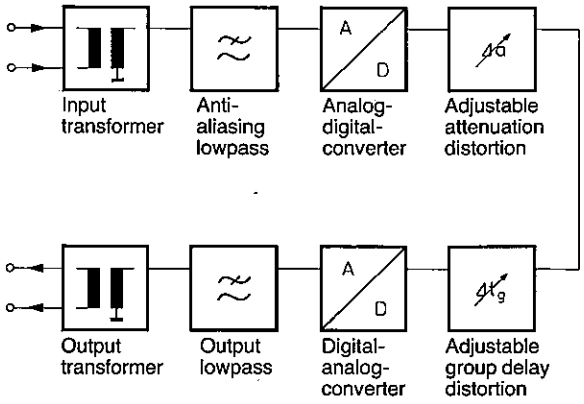


Fig. 4: Block diagram of a circuit producing linear line distortions

To this end, the same set of multiple poles of order two are chosen for all attenuation responses, these poles equalising the group delay within the frequency range of interest, but having no effect on the attenuation response. The zeroes of the transfer function then serve to set the desired attenuation characteristic /1/. The zeroes are all mirror images of eachother in the unit circle, and therefore do not affect the group delay response. If no attenuation distortion is required, the zeroes inside the unit circle are shifted so that they each cancel out one of the multiple poles, which results in a pure all-pass filter.

All the filters used for producing the attenuation distortion are 20th. order. The filter slope requirements do not demand such complexity, but this is necessary to ensure that the group delay error of the complete simulator remains less than 1 µs.
The distortions are thus implemented by utilising the symmetrical properties of poles and zeroes.
The filters producing the attenuation characteristics are formed by a cascade circuit of recursive digital filters of order 4, these being a good choice in view of the smaller memory space required.

The filters used for simulating the group delay characteristics consist of a network of cascaded 2nd. order filters /2/.

3. NON-LINEAR CIRCUIT CHARACTERISTICS

3.1. Frequency shift, phase hits, phase jitter

The signal to be distorted is mixed with the shift frequency in a quadrature modulator, the signal appearing at the output then being shifted by precisely this frequency. Phase jitter and phase hits can be generated in the same way.

The maximum frequency shift should be ± 10 Hz. Phase jitter with a jitter angle of between 0.2° and 30° is produced sinusoidaly with a frequency varying between 50 Hz and 300 Hz. Phase hits with a magnitude between 0° and ± 165° can be triggered at regular intervals of between 10 ms and 4 s.

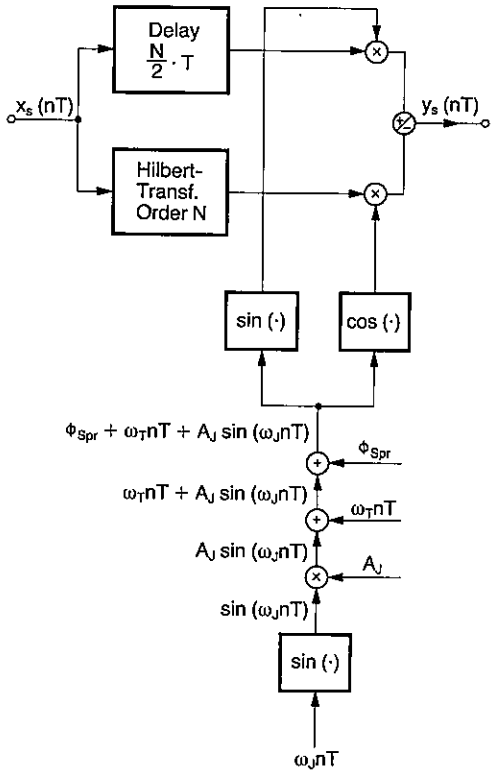


Fig. 5: Block diagram of a circuit for generating frequency shift, phase jitter and phase hits

The output signal y(nT) is calculated as follows for a frequency shift towards lower frequencies:

$$y(nT) = A_S \sin(w_S nT) \cdot \sin(w_T nT + A_J \sin(w_J nT) + \phi_{Spr}) + A_S \cos(w_S nT) \cdot \cos(w_T nT + A_J \sin(w_J nT) + \phi_{Spr})$$

$$= A_S \cos(w_S nT - (w_T nT + A_J \sin(w_J nT) + \phi_{Spr}))$$

Where

$x_S(nT) = A_S \sin(w_S nT)$ Input signal

$x_T(nT) = \sin(w_T nt)$ Frequency shift

$\phi_J(nT) = A_J \sin(w_J nT)$ Phase jitter

(w_J = Jitter frequency, A_J = Jitter amplitude)

ϕ_{Spr} Phase hit

A frequency shift towards higher frequencies is calculated by subtracting the two terms in the equation for y(nT).

3.2. Sudden changes of level, interruptions

Quasi-stochastic events, such as sudden changes of level and interruptions, are also important. Sudden changes of level between + 6 dB and - 6 dB in 0.1 dB increments and interruptions can be implemented easly by multiplying the output signal with a constant between 2 and 0.5 or 0, the duration and period of these events being controlled by a timer.

3.3. White noise

Band-limited white noise i.e. noise with a constant power density in the range from 300 Hz to 3400 Hz, is obtained by the software simulation of a feed-back shift register (length of the random sequence = $2^{15} - 1$), whose output values are passed to a digital telephone channel filter. This filter weights the spectrum in the desired manner, at the same time producing the standard distribution from the original uniform distribution.

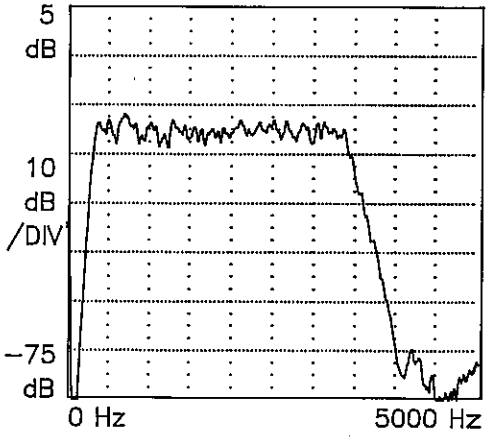


Fig. 6: Band-limited white noise

3.4. Impulsive noise

Impulsive noise and bursts are further important quasi-stochastic events. Impulses can be generated by adding a positive or negative DC component of variable value for a selectable duration.

In relation to bursts, studies /3/ on various data circuits have shown that the duration of the bursts and their intervals are statistically independent of each other and exhibit an approximately exponential distribution. The exponential distribution with the mean value T_m has the density function

$$p(t) = 1/T_m \cdot \exp(-t/T_m).$$

For the purposes of practical generation, the exponential distribution is best approximated by means of a discrete distribution, the geometric distribution. In this case, the density function is given by

$$w(nT) = P\cdot(1-P)^{n-1},$$

where T is the sampling interval and P a parameter of the distribution. The mean is calculated as T/P.
The bursts themselves are simulated by noise whose amplitude also changes in accordance with an exponential distribution /3/.

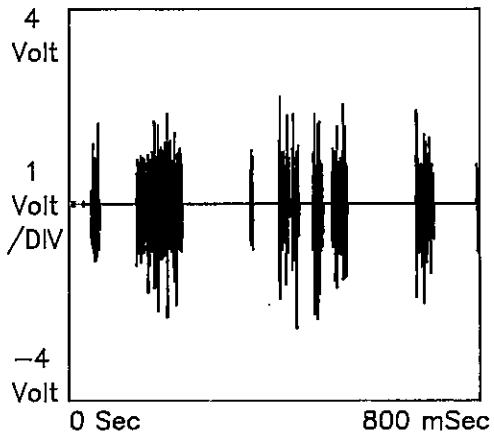


Fig. 7: Bursts

3.5. Single tone interference

Single tone interference, i.e. a sine-wave signal of variable frequency and amplitude added to the output signal, is generated digitally.

3.6. Harmonic distortion

Harmonic distortions for a specified input level can be formed by adding input values raised to the corresponding power /4/. 2nd, 3rd, and 4th order distortion factors are required.

Input signal:

$$x_S(nT) = A_S\cos(w_S nT)$$

Output Signal:

$$\begin{aligned} y_S(nT) &= c_0 + c_1 x_S(nT) + c_2 x_S^2(nT) \\ &\quad + c_3 x_S^3(nT) + c_4 x_S^4(nT) \\ &= \left(c_0 + 1/2\cdot c_2 A_S^2 + 3/8\cdot c_4 A_S^4\right) \\ &\quad + \left(c_1 A_S + 3/4\cdot c_3 A_S^3\right)\cdot\cos(w_S nT) \\ &\quad + \left(1/2\cdot c_2 A_S^2 + 1/2\cdot c_4 A_S^4\right)\cdot\cos(2w_S nT) \\ &\quad + 1/4\cdot c_3 A_S^3\cdot\cos(3w_S nT) \\ &\quad + 1/8\cdot c_4 A_S^4\cdot\cos(4w_S nT) \end{aligned}$$

However, these non-linear processes create frequency components which are greater than half the sampling frequency.

sampling frequency	f_s	= 10.0 kHz
Maximum signal frequency	f_{max}	= 3.4 kHz
4th harmonic	$4\cdot f_{max}$	= 13.6 kHz

The minimum sampling frequency required for this purpose is

$$f_{Smin} = 27.2 \text{ kHz}$$

For this reason, the sampling frequency is increased by a factor of three by interpolation before producing the non-linearity. Once the signal has been distorted, the sampling frequency is returned to the original value by decimation.

The interpolation and decimation filters are designed as FIR filters so as to avoid any additional group delay distortions /2/.

3.7. Listener echo

A listener echo can be generated by attenuating the output signal and delaying it in a shift register for the desired echo transmission time. The delayed signal is subsequently superimposed on the input signal.

4. CONCLUSIONS

The objective was to develop a device capable of generating all the linear and non-linear distortions described in CCITT Recommendation V.56. All the circuit characteristics described therin can be simultaneously simulated in real-time, using three TMS32010 signal processors with a sampling frequency of 10 kHz. The use of these efficient signal processors allows the circuit parameters to be varied within wide limits by means of software, and the possibility of adapting to new requirements or future modifications.

REFERENCES

/1/ Feistel, K.H., Ein Simulator für die Dämpfungs- und Gruppenlaufzeit-Verzerrungen von Fernsprechverbindungen, Frequenz No. 6 (1984), p. 126 - 130

/2/ Rabiner, Gold, Theory and Application of Digital Signal Processing (Englewood Cliffs N.J., Prentice Hall, 1976)

/3/ Pangratz, H., Ein Generator zur Nachbildung der auf Datenleitungen auftretenden Büschelstörungen, NTZ No. 5 (1972), p. 253 - 258

/4/ Schuon, Wolf, Nachrichten-Meßtechnik (Springer Verlag, Berlin, 1981)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DESIGN OF A HIGHLY FLEXIBLE DIGITAL SIMULATOR FOR NARROWBAND FADING CHANNELS

Helmut Brehm, Walter Stammler*, Martin Werner

Lehrstuhl für Nachrichtentechnik der Universität Erlangen - Nürnberg
D-8520 Erlangen, W-Germany

* AEG Aktiengesellschaft, Geschäftsbereich Hochfrequenztechnik
D-7900 Ulm, W-Germany

To simulate the fading effect, encounted in narrowband mobile radio communication, a digital generator is presented. It produces a multiplicative random disturbance, consisting of a product of a lognormal and a complex-valued Gaussian process. The generation algorithm is designed to realize a wide variety of channel parameters with little effort. The generator is implemented on a 16-bit signal processor TMS32010 and can be employed for realtime applications. Measurements indicate extremely good coincidence between experimental and theoretical probability densities and correlation functions.

1. INTRODUCTION

In mobile radio communication the quality of transmission is suffering from the so called 'fades' of the electrical field. Hence hard- or software generators, simulating the fading channels are required for the test of new encoding or modulation concepts as well as for the evaluation of new hardware equipment. Narrowband fading channels may be modelled by multiplying the complex input signal $\underline{u}(k)$ with a complex random disturbance $\underline{\tilde{z}}(k)$ /1-3/:

$$(1) \quad \underline{v}(k) = \underline{u}(k)\,\underline{\tilde{z}}(k).$$

Here $\underline{u}(k)$ and $\underline{v}(k)$ are equivalent lowpass signals, obtained from the RF-signals by demodulation with $\exp(-j2\pi f_o kT)$. For $\underline{\tilde{z}}(k)$ a generalized statistical model is chosen consisting of a lognormally disributed random sequence $s(k)$, multiplied by a complex-valued Gaussian random signal

$$(2) \quad \underline{z}(k) = x(k) + j\,y(k)$$

with statistically independent real- and imaginary-part /1-3/:

$$(3) \quad \underline{\tilde{z}}(k) = s(k)\,\underline{z}(k).$$

The phase of $\underline{z}(k)$ results to be uniformly distributed, the envelope exhibits a RAYLEIGH- or RICE-distribution, depending on whether the mean values $\langle x(k)\rangle$ and $\langle y(k)\rangle$ are zero or not. The influence of the factors $s(k)$ and $\underline{z}(k)$ may be characterized as follows: The factor $s(k)$ accounts for slow variations of the signal energy, which are caused by shading effects, whereas $\underline{z}(k)$ models the interferences due to multipath propagation of the signal on its way from the transmitter to the receiver's antenna in the moving vehicle.

Before a generation procedure for the random sequence $\underline{\tilde{z}}(k)$ can be discussed, the characteristics of the signals involved need to be explained in more detail. Then a digital algorithm ('fading generator') for on- as well as off-line generation of $\underline{\tilde{z}}(k)$ will be presented together with experimental results. To cover a wide variety of channel parameters, the bandwidths of the signals generated have to be varied over a range of three decades. The random sequences obtained, may be employed either for digital simulations in the baseband (see (1)) or for analog simulations after D/A-conversion and modulation.

2. CHARACTERISTICS OF THE SIGNALS

From theoretical calculations the following model for the power density spectrum of the Gaussian process $\underline{z}(k)$

$$(4) \quad \Phi_{\underline{z}\underline{z}}(f) = \begin{cases} 1/(2\pi f_D\sqrt{1-(f/f_D)^2}\,) & \text{for } |f| < f_D \\ 0 & \text{else} \end{cases}$$

can be derived under the assumption of uniform angular distribution of the received electrical waves /1/. The Doppler frequency

$$(5) \quad f_D = f_o\, v / c$$

is given in terms of the carrier frequency f_o, the vehicle velocity v, and the velocity of ligth c. From (4) the normalized autocorrelation

sequence results as

$$(6) \quad \varphi_{\underline{zz}}(k) = J_0 (\pi k f_D / f_S) ,$$

where J_0 represents the Besselfunction of order zero and f_S stands for the sampling frequency.

The factorprocess s(k) is characterized by a one-sided lognormal probability density function (pdf)

$$(7) \quad p(S) = (1/\sqrt{2\pi}\xi S) \exp[-(\ln S-\lambda)^2/(2\xi^2)]$$

$$\text{with} \quad \xi = (s_L \ \ln 10) / 20$$
$$\lambda = (F_{os} \ \ln 10) / 20 - \xi^2$$

depending on the squared ensemble average F_{os} [dB] of the magnitude of the electrical field at the receiver and its variance s_L [dB] . For the power density spectrum of s(k) the shape of a Gaussian curve with an additional delta-pulse at f=0, corresponding to the nonzero mean of the process (7), is quite common /3/:

$$(8) \quad \Phi_{ss}(f) = d \exp[-0.5 (f/f_L)^2] + c \ \delta_0(f).$$

The 'corner-frequency'

$$(9) \quad f_L = v / x_L$$

depends on the area-constant x_L and on the vehicle velocity v. Due to equ. (8), the correlation function has the sphape of a Gaussian curve as well.

Now the relations between the channel parameters and the corner frequencies are known and we may look at some numerical values. Choosing v = 3...300km/h , f_0 = 80...1000MHz , x_L = 1...150m, F_{os} < 20dB , and s_L < 15dB ; then the corner frequencies range from 0.2 to 280Hz for the Gaussian processes and from 0.05 to 1.9Hz for the lognormal distributed signals.

3. CONCEPT OF THE DIGITAL FADING GENERATOR

The fading generator is designed for high speed operation on a 16-bit signal processor TMS32010 in order to provide a tool for simulations as well as for realtime applications. The sampling rate is fixed to 2.5kHz in order to avoid the necessity of tuning clocks and tuning analog filters (after D/A-conversion) whenever channel parameters are changed. Moreover this solution facilitates digital simulations, where fixed sampling rates are given anyhow. For a first test the algorithm is implemented on a minicomputer PDP 11/60, where all the filter design is performed as well.

The Gaussian process:

The generation of the Gaussian random signals is illustrated in fig. 1. Starting off with uniformly distributed white noise from a PN-generator, we obtain a Gaussian distribution by nonlinear mapping of the amplitudes. The nonlinearity is realized by lookup-tables and includes linear interpolation between tabulated values. The desired power density spectrum (see (4)) is adjusted by a linear filter H(z) and subsequent interpolation of variable rate R_G. The use of an interpolator appears adventageous for several reasons:

* The distortions of the pdf caused by extreme narrowband linear filtering of PN-sequences can be avoided by this solution.
* By introducing an interpolator instead of using a filter with smaller bandwidth we reduce distortions of the filter response caused by finite coefficient wordlength on the signal processor (16 bit).
* Provided that an interpolator of rather simple structure can be employed, then a change of the interpolation rate R_G is performed on the signal processor with little effort. Hence the bandwidth of H(z) may be reduced by a factor R_G easily.

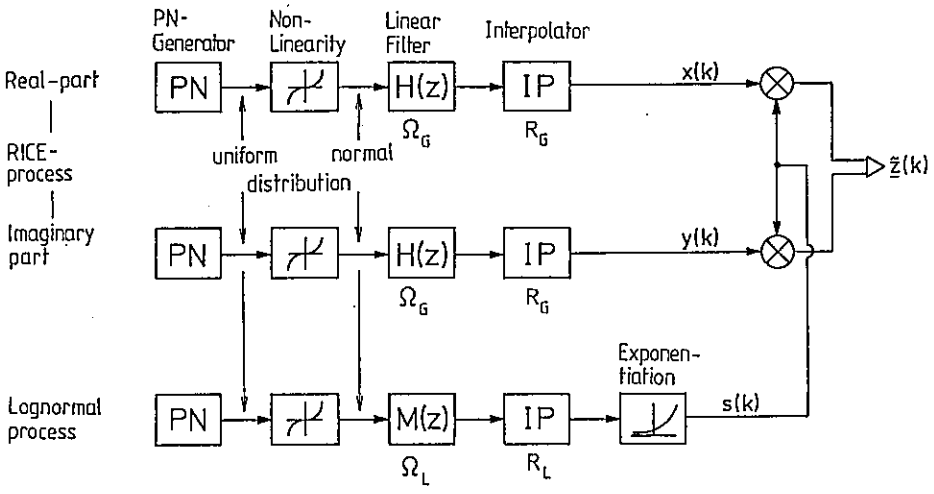


Fig. 1: Blockdiagram of the fading simulator.

To consider this last aspect, we choose a linear interpolator. Moreover we demand, that the interpolator is used only, if f_D drops below $f_S/440 = 5.68$ Hz. Thus the linear interpolator definitely is optimal in the mean square error sense (mse).
In general, interpolation will affect the distribution of a random signal. Since we have ensured, however, that interpolation is applied only in connection with sufficiently low corner frequencies of the linear filter, the interpolator will not change the univariate pdf. This may be proven by considering interpolation at L points as a switching between L FIR-filter outputs with normally distributed input signal. The filtered signals are Gaussian as well. Moreover they have the same power, if the autocorrelation of the input signal is unity for lags i=1,...N , where the interpolator-filter is of length 2N. In the case N=1 (linear interpolator) the correlation coefficient lies well above 0.999 .
The linear filters are designed such that a change of corner frequency can be done with little effort. We divide the frequency-range of interest (5.68<f<280Hz) into three intervals, each of which is represented by a standard filter. Within each interval, allpass transformation is applied to get the desired filter coefficients. The corner frequencies of the standard filters and the interval width are optimized with the intention, to keep non-linear frequency distortions due to allpass transformation as small as possible /4,5/. The standard filters are eighth-order IIR-filters and they are designed by reduced local search with initial guesses, obtained from elliptic lowpass fiters. The frequency response of the standard filter with lowest corner frequency ($f_S/110.5$) is illustrated in fig. 2 . The stopband attenuation is about 60dB.
The structure, selected for a minimum of coefficient quantization errors, consists of cascaded blocks of second order. The blocks are realized in normal-form with maximum and identical power of the state variables /4/.

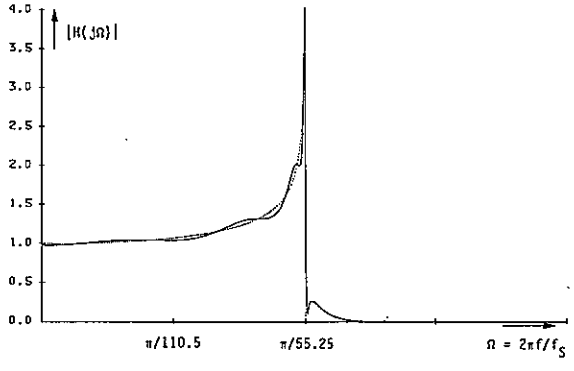


Fig. 2: Approximation of the desired frequency response $|H(j\Omega)|$ of the Gaussian signal (dotted line) by eighth-order IIR-filter (dragged curve).

The lognormal process:

Lognormally distributed amplitudes are obtained by exponentiation of normally distributed amplitudes (fig.1). This non-linear transformation is responsible for significant distortion of the autocorrelation function (fig.3).

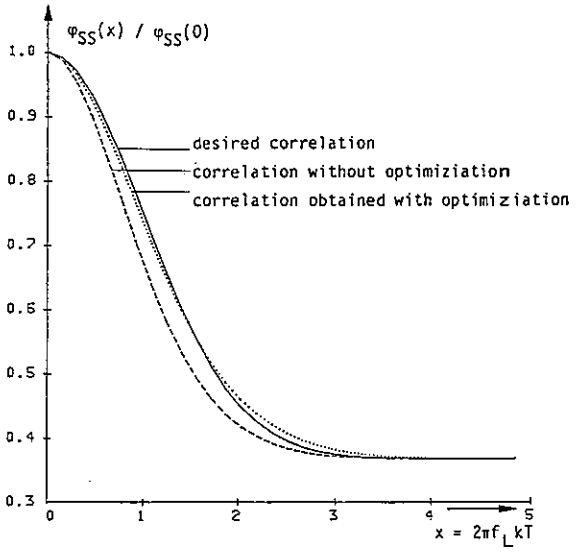


Fig. 3: Correlation of the lognormal p.ocess for $s_L = 8.686$dB after exponentiation.

Since an autocorrelation with Gaussian shape is desired at the output (see (8)), the corresponding input correlation should be equivalent to

$$(10)\quad g(k) = (1/\xi^2) \cdot \ln[1 + \exp(\xi^2-1)\cdot\exp(-0.5(2\pi f_L\, kT)^2)] \; .$$

Thus, any change of the parameters s_L, v or x_L would require a complete redesign of the digital filter M(z). To avoid this, we assume the input spectrum to be Gaussian as well. Now the corner frequency of M(z) is determined such, that it produces an approximation of the desired output correlation, which is optimal in the mse-sense. In practice, we first obtain the optimal corner frequency from a stored look-up table and then we perform an allpass transformation on a standard digital filter M(z). The generation of the Gaussian signal can be based on the principles discussed in the last section. The main differences concern the limit for interpolation (0.74Hz) and the fact, that only one standard filter is used for the range 0.74<f<1.9Hz. It should be kept in mind as well, that due to the Gaussian shape of the spectrum, the attenuation reaches a value of 60dB not before $f = 5.3\, f_L$.

4. RESULTS FROM A REALIZATION ON THE INTEGRATED SIGNAL PROCESSOR TMS32010

The algorithm described above, was implemented on the TMS 32010 from TEXAS INSTRUMENTS. It requires 790 command cycles of 200nsec length to produce a complex value $\tilde{z}(k)$. Both tables of typically 1K-words for the non-linearities may be integrated in the program-RAM as well.
Measurement results from this generator are illustrated in the following figures. Figure 4 showes the univariate pdf and the correlation of the Gaussian signal x(k) with a corner frequency of 10Hz. Experimental values (squares) and theoretical curves (dragged lines) are coinciding extremely well, even though theoretical correlation functions are evaluated for unquantized filter coefficients.
In the last figure results are given for the lognormal process with f_L = 0.66Hz and s_L = 7dB. Slight deviations in the correlation curves are due to the error by the approximation documented in fig. 3.

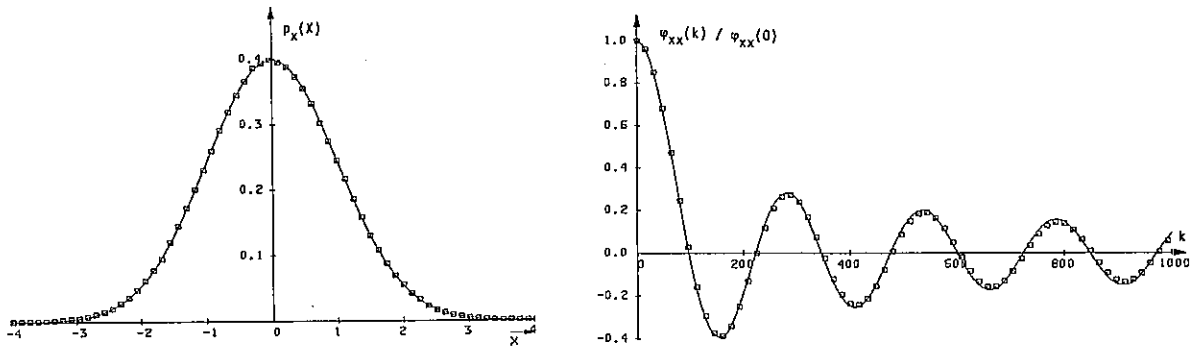


Fig. 4: Univariate pdf (left) and correlation function (right) of the Gaussian process x(k); normalized corner frequency f_D/f_S = 1/250; data base: 10^7 samples; □ experimental, —— theoretical results.

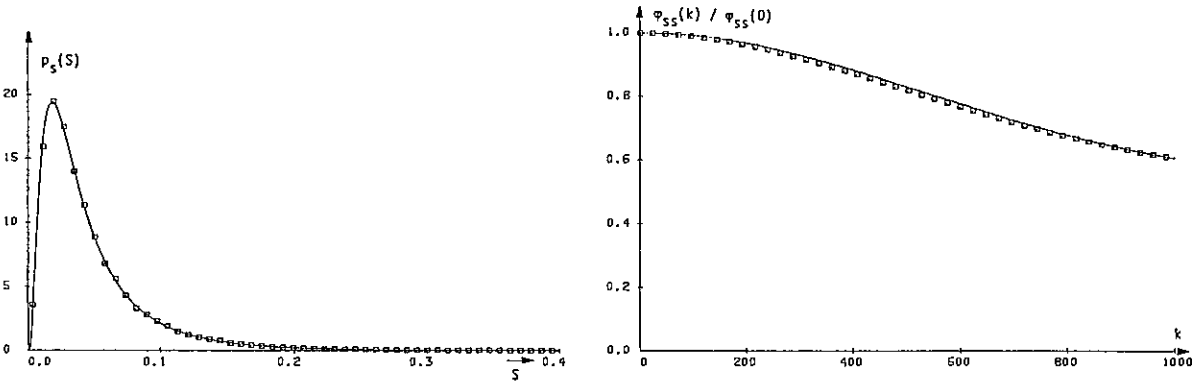


Fig. 5: Univariate pdf (left) and correlation function (right) of the lognormal process s(k); normalized corner frequency f_L/f_S = 1/134, s_L = 7dB; data base: 10^7 samples; □ experimental, —— theoretical results.

REFERENCES:

/1/ Jakes, W.C.: Microwave mobile communication, Wiley, New York, 1974.

/2/ Lorenz, R.W.: Zeit und Frequenzabhängigkeit der Übertragungsfunktion eines Funkkanals bei Mehrwegeausbreitung. Der Fernmeldeingenieur, vol.39, no.4, april 1985.

/3/ Aldinger, M.: Die Simulation des Mobilfunkkanals auf einem Digitalrechner, Frequenz, vol.36, 1982, pp. 145-152.

/4/ Werner, M.: Realisierung eines Digitalen Rauschgenerators zur Simulation von Fadingerscheinungen, Diplomarbeit am Institut für Nachrichtentechnik der Universität Erlangen - Nürnberg, Erlangen 1985.

/5/ Dittrich, L.: Echtzeitsimulation von Mobilfunkkanälen, Studienarbeit am Institut für Nachrichtentechnik der Universität Erlangen - Nürnberg, Erlangen 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMPLEMENTATION OF A HIGH SPEED VITERBI DECODER

J. Stahl, H. Meyr, M. Oerder

Aachen Technical University (RWTH)
Templergraben 55, D-5100 Aachen, Germany

Innovative satellite communication systems require that the given bandwidth be used efficiently. For recently proposed combined coding/modulation schemes the Viterbi algorithm proves to be the optimal estimator to decode the data. In this paper the main problem in implementing such a decoder at high speed is discussed. It is shown how the structure of a prototype decoder was influenced by the properties of the algorithm. A brief outline of a VLSI-implementation is given.

1. INTRODUCTION

The increasing demand for digital communication via satellites leads to the development of advanced coding/modulation schemes, which exploit the power and bandwidth on the satellite channels efficiently. The corresponding receivers tend to be much more complex than existing ones. Not only decoding but also synchronisation becomes a severe problem. In this paper the decoding part of an all digital receiver [1] is presented.

First the coding/modulation scheme being used will be presented. Then the Viterbi algorithm, as the optimal decoding procedure for this purpose, will be shortly reviewed with an emphasis on the realisation aspects. The main section deals with the structure of a prototype decoder, which was actually built up of TTL-components. The last section outlines some aspects concerning the future VLSIntegration of the decoder, which is presently under work.

2. CODING AND MODULATION

The Viterbi algorithm [2] (VA) is suitable in all cases, where coding schemes with a trellis structure are used. Important representatives of this category are the convolutional codes [3] and specially combined coding/modulation schemes, from which the so called Ungerboeck-Codes are well known [4]. Fig. 1 shows how such a code works:

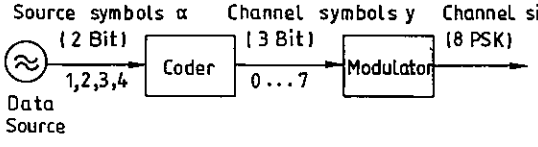


Fig. 1: Combined coding/modulation

The data source emits the data, to be transmitted, in groups of 2 bit. These are called the source symbols α. The coder uses this sequence to generate 3 bit channel symbols y which are fed into the modulator. Each of the 8 channel symbols correspond to one of the eight phase states of the output PSK-(phase-shift-keying)-signal.

For the digital receiver [1] we chose a code which is in most properties similar to codes suggested by Ungerboeck [5]. In Fig. 2 a trellis diagram of the code is given.

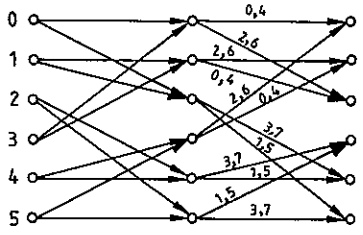


Fig 2: Trellis diagram

The dots represent the states of the coder and the arrows, labeled with the corresponding channel symbols, mark the transition of the coder state depending on the channel symbol y. To simplify the diagram the differential encoding between the channel symbols and the source symbols is not shown. The function of the coder can now be summarized as follows: given a coder state and a source symbol (one of four possible ones) the coder selects a path in the trellis diagram to the next state. The channel symbol belonging to this path is the output of the coder.

The main difference between the code and the Ungerboeck-code, which has the same coding gain of 3 dB is that we obtain a substantially better acquisition performance for the signal phase. This is accomplished through the rotationally

invariant code structure, which requires 2 extra coder states (6 instead of 4).

The properties of the high speed realisation of a Viterbi decoder, which are influenced by the coding/modulation scheme, are:

a) complex modulation (8PSK) which results in the requirement, that the VD has internal variables with higher accuracy than the VDs used with state of the art 4PSK and 2PSK modulation
b) 6 state code, which has the greatest influence on the hardware costs, because each state results in one arithmetical unit in a parallel VD
c) parallel transitions between states (there are only four different groups of channel symbols 0+4, 1+5, 2+6, 3+7, which correspond to the state transitions), which together with property d) simplifies the basic operations of the VA
d) each state has only two possible predecessors

3. VITERBI ALGORITHM

The Viterbi algorithm is well known as a recursive estimation of the most probable path through a trellis diagram. It works as follows:

For each recursive step :

1.) Compute a transition λ_y metric for each channel symbol y (the metric measures the probability of a path in the trellis diagram)

2.) For each state:

- Compute all sums of $\Lambda_{old} + \lambda_i$, where Λ_{old} is the accumulated path metric of the previous state and λ_i the metric for the transition leading into the new state with the channel symbol i.
- Compare the sums
- Select the largest sum and store its value as the new accumulated metric for this state.
- Extend the symbol sequence, which was stored with the corresponding previous state, with the symbol of the selected transition and store it with the present state.

Thus for each state the algorithm stores a sequence of symbols (survivor path), which most probably leads to this state. The memory of the survivor paths is finite and the oldest symbol of the most probable state is chosen as the estimated source symbol of this step.

The high speed implementation of the VA is strictly bounded by its recursiveness, because the results of one step must all be ready to perform the next one. Other algorithms (sequential decoding [3 pp. 349-378,6]) avoid this problem by searching only through a part of the trellis, but they do not have the following favorable properties of the VA:

- optimality
  the algorithm determines the most probably path through the trellis concerning the maximum-likelihood-principle

- constant workload
  in each step, there is a fixed count of operations to be carried out. This distinguishes it from sequential decoding algorithms, where the decoding speed depends on the input data (e.g. bad channel ==> low decoding speed).

- no data buffering
  each input data (digitized sample of the input signal) is used only once and need not be stored for later computations

- automatic acquisition
  no special initialisation, after the start of decoding or a temporary loss of signal, is necessary. After a few steps the algorithm tends to find the right path.

4. DECODER STRUCTURE

The Viterbi decoder can be subdivided into 3 major parts (fig. 3): the metric computation, the add-compare-select (ACS-) unit and the survivor memory.

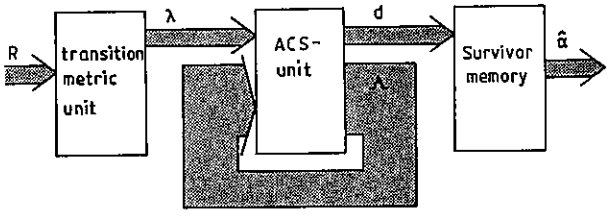


Fig. 3: Basic decoder structure

Each unit works at the same time on different data (pipelining) and they are clocked with the symbol rate, so that for each clock period one decoded symbol (2bit) is popped out at the end of the survivor memory. In order to get a high clock rate each functional unit has to be parallelized as much as possible. This holds particularly for the ACS-unit, where internal pipelining with the symbol rate cannot be used for speed up, since the recursive step must be done in one symbol time. In the design of a parallel VD the code structure can be exploited to reduce the hardware costs, which will now be shown.

- transition metric unit
the computation can be done in 8 parallel working units. The results could be fed directly into the following ACS-unit, but exploiting property c) (see 3.) of the code leads to a much simpler structure. If we look more closely at the ACS-operations in the case of state "0" there are four sums, which should be carried out and compared:

$$s_1 = \Lambda_0 + \lambda_0 \qquad s_3 = \Lambda_2 + \lambda_2$$
$$s_2 = \Lambda_0 + \lambda_4 \qquad s_4 = \Lambda_2 + \lambda_6$$

It follows that:

$$\max_{i=1\,4} s_i = \max(\max(s_1,s_2), \max(s_3,s_4))$$

$$= \max(\Lambda_0+\max(\lambda_0,\lambda_4),\Lambda_2+\max(\lambda_2,\lambda_6))$$

Therefore in the transition metric unit it is already be possible to decide whether λ_0 or λ_4 can belong to the survivor path (Thus the recursion cycle time in the ACS- unit can be reduced). This decision can be done in parallel for all pairs of transition metrics in the trellis (0+4, 1+5, 2+6, 3+7), from which only four different ones exist.

- ACS unit
From the statements above it follows that the ACS-Operation for one state is reduced to 2 additions and one comparison. If we compare this to an implementation without using the specific code properties we get the following table

number of additions	
direct implementation:	6 • 4 = 24
using code properties:	6 • 2 = 12
number of comparisons	
direct implementation:	$6 \cdot \binom{4}{2} = 36$
using code properties:	4 + 6= 10

Table 1: Hardware reduction in the ACS-unit

These reflections assume that all operations are done in parallel (e.g. if we compare 4 variables, we compare all possible pairs (6) and then decode the largest directly out of the 6 information bits from the comparators).

All operations are done with integer variables, so that a suitable overflow protection is needed. The well known method [3, pp. 258-261] is to compute the maximum of the path metrics and subtract this value from all state metrics. This operation has to be done in one symbol cycle in order to restrict the possible range - and thus the number of bits for the state metrics - as much as possible. The maximum selection and the subtraction of this varying value result in high hardware costs and normally slow down the operation speed of the VD. We avoid this by carrying out parallel monitoring all state metrics by only checking their highest bits. If they exceed a fixed value a fixed correction value is subtracted from all of them. This correction can be done efficiently in combination with the addition of transition metrics and requires therefore no extra hardware.

survivor memory
the VA requires the storing of survivor sequences during a certain length N. For high speed implementation it is not appropriate to store these in a RAM, because the tracing back to the oldest symbol of a state in each decoding step would result in N memory cycles. Even for low values of N this slows down the VA. We chose a specially organized memory which consists of cells like Fig 4.

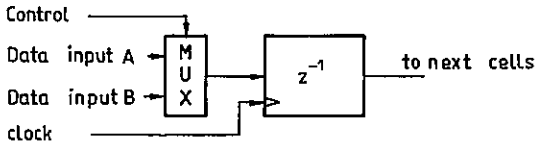


Fig. 4: Survivor cell

The memory cell can be fed with the information of two previous memory cells selected by the multiplexer control signal. This cell reflects the code structure (property d), see 3.).

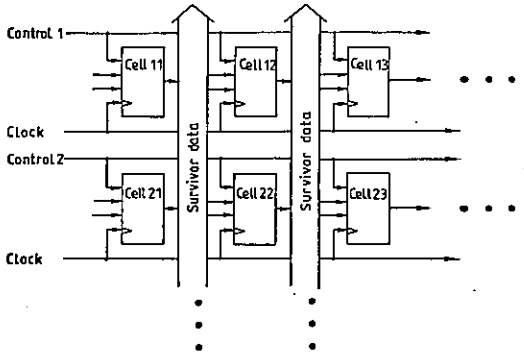


Fig. 5: Part of the survivor memory

By connecting the cells to form 6 registers (one for each state of the code) with length N the search of the oldest symbol in a sequence can be accomplished by parallel updating of the register contents of a state with the information sequence leading to this state (register exchange method). The control signals (fig. 5) are derived from the ACS-decisions and manipulate the data flow in the memory so that the oldest bits of each sequence can be accessed directly at the end

of the registers.
The length N depends on the code structure, and can be estimated regarding the length of the most probable error events during the decoding process. Simulation showed in our case that a decision depth of 20 symbols is sufficient (Fig. 5, SNR measured at the receiver input), thus making this solution feasible.

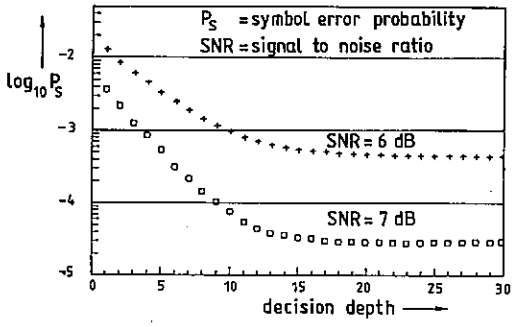


Fig. 6: Necessary decision depth (simulation)

We did the actual implementation of a prototype VD with standard and programmable TTL-ICs. Programmable logic (PAL) helped a lot to reduce the hardware costs, so that the parallel VD could be built up on 2.5 160x233mm boards using roughly 150 ICs. In this implementation we did not try to achieve high speed through using very fast technology. We wanted to show how fast - given a specific gate delay - a VD can work, when all the possibilities of the algorithm are used to speed up the operation. Thus at present our prototyp Viterbi decoder achieves a data rate of 12 Mbit/s, which is equivalent to 420 Million 8bit operations/s.

5. ASPECTS OF VLSI

Through the implementation of a prototype we could show how parallelism in the Viterbi algorithm can be exploited to achieve a high functional-throughput-rate. The decoder is intended for application on satellite channels, where much higher data rates (≈ 100 Mbit/s) are used. It should be possible to fill out the gap between the prototype and the desired performance through the use of VLSI-technology, since the gate delay, which at the board level depends mainly on the delay of the output drivers of the circuits, is considerably lower on the chip, depending on the technology, which can be used (Bipolar, CMOS).

The basic implementation aspects, shown in section 4. for the prototype, will hold also for the on-chip integration.

New problems are encountered due to the wiring delays of signals, which should be low to achieve a low recursion cycle time. Parallel functional units need to exchange their data on independent connections (avoiding multiplexing via a bus). Only the efficient realisation of this wiring can result in an acceptable wiring delay.
New investigations have to be made on how to design the individual functional units, e.g. a single ACS-cell, in order to achieve a suitable floorplan of the chip and thus low interconnection wiring delays.
A functional unit which is suitable for VLSI-implementation without changing the prototype design is the survivor memory. The structure of the memory as given in Fig. 6 can be used as a floorplan for the actual circuit layout. It is very regular (eg. Clock and Control signal run horizontally, data signals run vertically) and therefore easy to implement.

6. CONCLUSIONS

It could be shown, that decoding of complex coding/modulation schemes using the Viterbi algorithm is feasible even at high data rates. By exploiting the properties of the code substantial reductions in hardware costs can be achieved. A prototype decoder based on standard TTL-components was presented. The bottleneck of the computations is the recursive part of the algorithm, which can only be carried out quickly by using all the parallelisms the algorithm allows. Future investigations will concentrate on the VLSIntegration of a VD and should make an implementation of such a VD in satellite data link possible.

7. LITERATURE

[1] M. Oerder, G. Ascheid, R. Häb, H. Meyr: An all digital implementation of a receiver for bandwidth efficient communication, also published in this proceedings

[2] J.D. Forney: The Viterbi algorithm, Proc. of the IEEE, Vol. 62, Mar 73, p. 268-278

[3] A.J. Viterbi, J.K. Omura: Principles of digital communication and coding, McGraw-Hill, New York 1979

[4] G. Ungerboeck: Channel coding with multilevel/phase signals, IEEE Trans. on Inf. Th., Vol-IT-28, No. 1, Jan. 82, p. 55-67

[5] M. Oerder: Rotationally invariant trellis codes for mPSK-modulation, Int. Conf. on Communications, Chicago 85, Conf. Rec. p 552-556

[6] J.B. Anderson, S. Mohan: Sequential decoding algorithms : A survey and cost analysis, IEEE Trans. on. Comm., vol. COM-32, 1984, no. 2, pp 169-176

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

TMS-320 IMPLEMENTATION OF A 2400 bps V.26 MODEM

Dr. Joseph M. Perl, Alfred Bar and Jacques Cohen.

Tadiran Ltd., Communication Division, Digital Equipment Plant
P.O.B. 267 HOLON, Israel. Tel. (972-3) 5574730, Telex 35413.

An efficient software implementation of a standard, CCITT Recommendation V.26 modem on a TMS32010 microprocessor is presented. The modem contains all the necessary synchronization and detection subroutines for both A and B signalling alternatives, as well as a linear adaptive equalizer. The paper summarizes the software and the hardware implementations. It concludes with the modem performance on both compensated and uncompensated telephone channels.

1. INTRODUCTION

Since digital pulses cannot efficiently be carried over analog signal oriented networks, modems are being employed. Line modems range from voice grade of about 300 bit-per-second (bps), to wideband modems, up to or above 1 Mbps. These modems are designed to work on the existing telephone system and range in cost up to about 1$/bps . The high values of the cost range are reached when high-efficiency spectral modulation is required for bandwidth compression, or adaptive equalization is utilized to compensate for nonlinear, time varying channels. The newly introduced VLSI technology tends to lower these costs so as, for example, a widely used 212A Bell type modem, costs as low as 0.1 $/bps. Radio modems, on the other hand, are usualy more complex and expensive, especially if intended for fading, noisy channels (e.g. HF or troposcatter [1],[2]). Such radio modems operate in harsh environments especially in military aplications. As a result they are about ten times more expensive then line modems.

Following the general trend in electronics, the existing bulky modems are being replaced by compact units based on dedicated chips. Alternatively, modem functions are realized by software routines on digital signal processors [3]. Such an implementation is especially cost effective, if the modem function is only an add-on to other software on existing hardware. Such an implementation provides flexibility and future growth potential at no additional cost. If newly designed modems are to speak to the already existing ones and to each other, the manufacturers have to comply to existing standards. There are now, two broadly accepted families of modems:

a) The Bell models. A large proportion of modems utilized in North America are of Bell construction. For the 2400 bps rate of interest here, the model 201 is a DQPSK modem, suited either for leased lines or for the "general switched telephone network". Accordingly, for reliable operations, it requires C2 type conditioning of the line.

b) Modems following the "Consulting Comitee for International Telephone and Telegraph Administrations (CCITT) Recommendations". These recommendations are generaly adopted in Europe and on international connections between the U.S. and countries which follow the CCITT recommendations. The V.26 CCITT recommendation for 2400 bps rate, is equivalent with the Bell model 201C. According to this recommendation, the V.26 modems are limited to leased circuits which conform to M.1020 CCITT requirements (similar to Bell C2 conditioning).

Although the popularity of 2400 bps and higher rate modems is increasing , CCITT V.26 modems seem to be limited either to conditioned, leased circuits or to the switched circuits of those countries where the public lines are capable of accepting high bit rates. This situation has been changed, by adding channel equalizing functions, to the basic modems.

The present paper describes such a V.26 type modem, software implemented on a TMS 32010 digital signal processor. The MD601 line modem incorporates an adaptive equalizer, whose task is to improve the modem performance on unconditioned telephone circuits.

Chapter 2 starts with a review of the requirements of the CCITT,V.26 reco-

mmendations followed by the MD601 modem description. In chapter 3, the actual implementation is discussed and both hardware and software aspects are considered. In chapter 4, the modem performance is presented, followed by conclusions.

2. MODEM DESCRIPTION

2.1 Features

The main characteristics of modems conforming to the CCITT V.26 recommendation are:

*synchronuous 2400 bps data rate
*full duplex,four wire operation
*carrier frequency of 1800 Hz
*four phase DQPSK modulation with two alternative coding arrangements (Table 1)

Dibit	Phase change	
	Alternative A	Alternative B
00	0 degrees	+ 45 degrees
01	+ 90 degrees	+ 135 degrees
11	+ 180 degrees	+ 225 degrees
10	+ 270 degrees	+ 315 degrees

Table 1 - V.26 Coding Alternatives

2.2 Functional Block Diagram

In Fig.1 the functional block diagram of the MD601 modem is given.
The transmitter (Fig. 1a) generates a DQPSK modulated tone of 1800 Hz, from a sine table. The values of the signal were pre-calculated using a Fortran simulation of the modem and then stored in the memory. Similarly, the various digital filter coefficients were calculated off-line and stored for real time use. The same Fortran program has been used for the simulation of the finite word effects of both computing and coefficients. The modulation tables, one for each alternative , implement both the modulation and shaping functions and are a trade-off between the real-time and memory needs of the modem. The computed discrete-time signal,

$$V(n) = \sum_k [X(k).g(nT - kT).\cos(WonT) - Y(k).g(nT - kT).\sin(WonT)] \quad (1)$$

for Wo = 2.Pi.1800
X(k) , Y(k) are the bipolar bits and g(n) = shaping filter

is deposited in the DMA device memory. It is subsequently extracted in analog form,via a D/A converter.

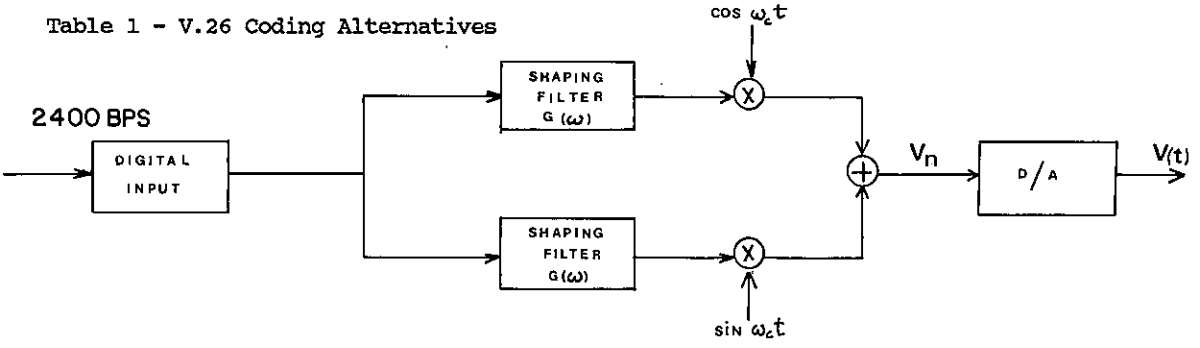


Fig.1-a .Transmitter Block Diagram

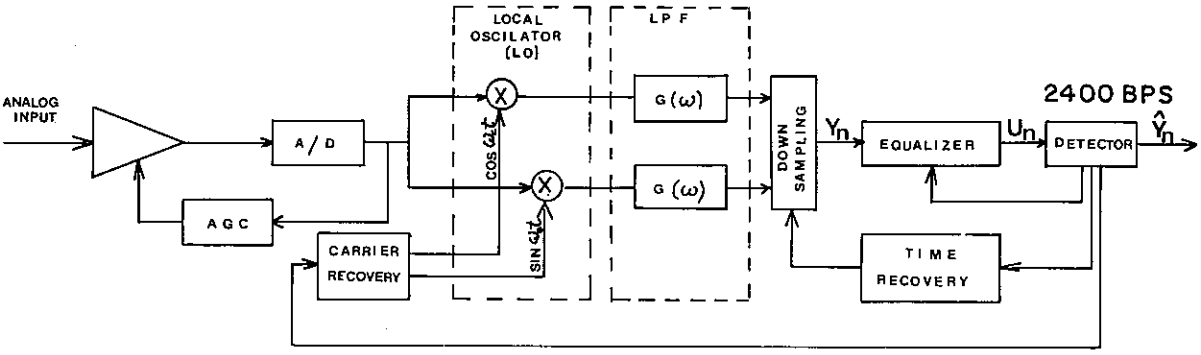


Fig.1-b. Receiver Block Diagram

$$V(t) = Re\ (SUM_{k}\ [(X(k) + jY(k)).\ g(t - kT).exp(-jWot)\ ]\ )\quad (2)$$

The receiver (Fig.1b) gets its input from the DMA memory, where it is deposited as analog samples by the A/D converter. The anti-aliasing filter serves in the AGC loop which is software controlled.

Pre-filtering

After the local oscillator (LO), the low pass shaping filters (2) are used to filter out the high frequencies. Following the 1:6 downsampling, imposed by the 7200 sampling rate, the symbol samples enter the equalizer.

Equalizer

The output of the shaping filters Y(n) passes through 3 equalizing loops in order to obtain the final phasor U(n). The equalizer contains, in fact, a linear adaptive equalizer ,a "fine" AGC and the phase jitter corrections. The "fine" AGC and the phase jitter correction loops maintain the amplitude and phase tracking in the equalizer loop. The phasor on which the differential bitdetection is performed is obtained by taking the minimum distance E(n) between U(n) and each of the 4 possible phasors (Table 1).

Linear adaptive equalizer

The linear adaptive equalizer compensates for the distortions of the channel. The adaptive equalizer in MD601 can be replaced by a fixed equalizer for those applications where the convergence time of the adaptive equalizer is critical (short messages of a few seconds duration). The adaptive equalizer output U(n) at time nT can be written as:

$$U(n) = \overline{Y}^{T}(n)\ .\ \overline{W} = \sum_{i=0}^{N-1} Y(n-i).W(i)\quad (3)$$

The complex vector $\overline{Y}(n)$ is the output of the predection shaping filters and $\overline{W}$ is the complex coefficients vector of the equalizer taps. Due to real-time considerations, N=7.

The coefficients vector $\overline{W}$ is corrected every symbol:

$$\overline{W}(n)=\overline{W}(n-1)-(2*alphal*E(n-1)*\overline{Y}^{*}(n-1))$$

where

$$E(n-1) = Y(n) - \hat{Y}(n)\quad (4)$$

is the error between the detected phasor and the transmitted phasor,

$\overline{Y}^{*}(n-1)$ is a complex conjugate vector containing the previous values of Y(n), and alphal is a constant factor chosen as a compromise between the convergence speed and the jitter on the taps of the equalizer.

Carrier Recovery

The phase error obtained in the phase jitter correction loop is used (after integration) to make small corrections to the demodulator carrier phase argument : this is done in order to keep the phase jitter correction as small as possible. Thus the jitter correction by the equalizer corrects small and short-term errors around the phase maintained by the carrier recovery loop.

Time Recovery

A zero crossing type routine is being used. Digital FIR filters are used to filter the 1200 Hz component out of the recovered data. The zero crossing logic commands a programmable counter, which in turn adjusts the sampling frequency. In order to speed the time recovery process the transmitter sends an optimal "10101..." sequence at the start of each transmission, according to the CCITT recommendation.

3.MODEM IMPLEMENTATION

The MD601 line modem is implemented on a (6*8) inch PC card on which the TMS32010 with its memories and supporting logic is located. The card consumes up to 8 watts in 5V and +/- 15V, but in addition to the MD601 line modem, it contains a full MD600 HF modem [4].

3.1 Hardware Configuration

The TMS32010 microprocessor of Texas Instruments, contains 144 words of internal RAM. The 16 K words instruction memory is contained in ROM. The TMS has access to only 4 K simultaneously. To provide access to the whole ROM a paging technique is being used. The first 2 K of ROM (0 to 7FF) are always available, while the last 2 K (800 to FFF) are software selectable.
There is provision for 4K RAM memory. The access to the RAM is via input and output ports. Part of this RAM is used for DMA: 256 words for the analog input and 256 words for the analog output.
The DMA counter can be read at any moment and is used as the real-time clock of the software. The digital input information from the terminal and the transmit clock are read through the least significant bits of the DMA input words.

The digital output information from the modem (receiver) to the terminal is outputed during an interrupt routine, whose rate is programmable and is a function of the bitrate. Thus the digital I/O is implemented via software USART's.

3.2 Software Architecture.

The task scheduler of MD601 executes the tasks on a FIFO basis. The wait mode is responsible for the timing of the various receiver and transmitter tasks, based on the DMA counter.

Table 2 - Real-Time Load Distribution

Transmitter	16%
Receiver	54%
System Overhead	23%
Total	93%

This real-time allocation is to be seen as a result of a number of design iterations and implies some trade-offs between time and memory requirements. Because of the overhead (save and restore of 128 words of internal RAM) caused by invoking a task, a compromise was reached at the receiver tasks between the number of symbols to be treated simultaneously and the jitter at the time recovery loop caused by the processing delay when too many symbols are treated simultaneously.

4.MODEM PERFORMANCE

4.1 SNR Performance

Figure 2 shows the theoretical bit error rate for coherent DQPSK modem and the bit error rate obtained by the MD601 implementation. The modem performance differs by less than 1 dB from the theoretical performance.

Table 3. Channel simulator results

SNR (dB)	Group delay 1	Group delay 2	Atte-nuat.	BER Alt. A	BER Alt. B
6	0	0	0	5 10^{-3}	6 10^{-3}
10	0	0	0	1.4 10^{-5}	1.8 10^{-5}
10	0	0	6	1.2 10^{-4}	1.8 10^{-4}
10	0	6	0	7.2 10^{-4}	1.2 10^{-3}
10	6	0	6	6.7 10^{-4}	5.3 10^{-4}
10	6	6	0	1.2 10^{-4}	2 10^{-5}
10	6	6	6	1.4 10^{-4}	1.2 10^{-4}

4.2 Channel Simulations

The results in Table 3 were obtained by using the W & G channel simulator [5]. The numbers in the group delay columns represent the position of the simulator front panel controls. These BER values were measured after the lock-in time of the modem, which comprises of the initial time recovery, the carrier recovery and the linear equalizer convergence time. Under the most difficult W & G channel conditions this lock-in time was less than 500 ms, being typically of the order or 150 ms.

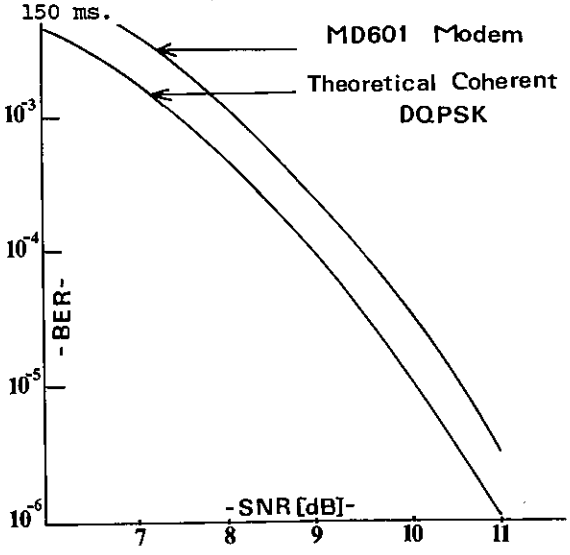


Fig.2 - MD601 BER vs. SNR Performance

5. CONCLUSIONS

A software realization of a standard V26 modem was presented. The modem was implemented on the TMS 32010 signal processor and is packaged in a single, compact PC board. An adaptive equalizer was added, that improves its performance on difficult channels. The performance was measured to be within 1 dB of theory.

REFERENCES

[1] "MD-23 High Speed HF Data Modem", Tadiran Ltd. Communications Divivsion, Holon, Israel, 1984.
[2] "RF-3466T High Speed Tactical HF Data Modem", Harris Corp. RF Communication Group, Rochester, N.Y. 1984.
[3] J.A. Iapicco and S.P. Verma, "Voiceband Modem Implementation Techniques for Programmable Digital Signal Processors", PP. 71.2.1 - 72.2.6, IEEE Conf. 1981.
[4] J.M. Perl and E. Trachtman, "Viterbi Decoding on HF Channels", submitted to MILCOM-86 Conference.
[5] Telephone Channel Simulator TLN-1 from Wandel and Golterman.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DESIGN AND ANALYSIS OF SERIAL AND PARALLEL DATA TRANSMISSION SYSTEMS HIGHLIGHTED BY TIME-FREQUENCY DUALITY PRINCIPLE

Slobodan Nedić
Institute "Mihajlo Pupin"
Volgina 15, 11000 Belgrade, Yugoslavia

Outgoing from the well known synchronous serial data transmission system of Nyquist or partial response type, an actual dual system is introduced in this paper according to the principles established by Bello. Polyphase networks with DFT processing are used to extend time-frequency duality relationship existing between serial and standard parallel data transmission systems. Benefits gained from viewing a standard parallel system as dual of serial one are discussed in a rather qualitative manner. Some quantitative results are provided with respect to combating the effects of multiplicative noise, based on automatic adaptive equalization in transformed domain.

1. INTRODUCTION

In the course of data transmission over voice band channels, (standard telephone and HF radio ones), synchronous parallel data transmission systems have been introduced long ago. Originally used for high speed HF data transmission, these systems have become more and more attractive for several reasons:gradual speed adaptibility, inherent channel measurement capability, line signal amplitude distribution compatible with FDM system requirements, inherent optimality with respect to the presence of white Gaussian noise, as well as reduced vulnerability to impulsive noise disturbances, the last property being confirmed by theoretical analysis and practical measurements, besides being deduced intuitively.

For both systems, serial and parallel one, considerable work has been done in the past with respect to design approaches and performance analysis. The aim of this paper is to point out the possibilities of using known results relating to serial data transmission in design and analysis of a parallel one, and vice versa.

Factors, such as the usual use of time and frequency conceptualization, as well as limited DFT - lengths exploited in practice, contribute to the fact that this dual system has much in common with a standard synchronous parallel data transmission system, where a number of standard low-rate synchronous data signals are orthogonaly frequency division multiplexed [4,5]. Although time-frequency dual of a serial and standard parallel system can be designed to behave identicaly, the former system suits better some system design approaches such as the use of differential, trellis or frequency domain partial response coding. Also, the time-frequency duality principle enables an efficient analysis of certain dual channel impairments and better understanding of its results.

After citing some time-frequency duality definitions, the actual dual system is introduced relying on TDM-FDM conversion approach based on polyphase networks with DFT processing, [3]. The possibilities of using results known for one system to get corresponding results in dual system with dual impairment are treated in a rather qualitative manner. Particular attention is devoted to the method of combating effects of multiplicative noise based on the use of automatic adaptive equalization in transformed domain. Some quantitative results are given in case of frequency offset and phase jitter for linear and decision feedback equalizer structures.

2. TIME-FREQUENCY DUALITY DEFINITIONS

For the purpose of introducing the dual of serial data transmission system we are going to deal with, it appears useful to recall some of relevant time--frequency duality definitions given in [1].

Time function and corresponding spectrum which describe the input (or output) of an element in a communications signal processing network are *dual network variables.*

The function $z_2(x)$ is the *direct dual* of $z_1(x)$ if it is the (direct) Fourier transform and the *reflection dual* if it is the inverse Fourier transform of $z_1(x)$.

Let O_t^E and O_f^E denote those input-output operators of signal processing element E which relate input time functions to output time functions and input spectra to output spectra, respectively. Then element E_2 is *direct dual* of E_1 if $O_f^{E_1} = O_t^{E_2}$ and *reflection dual* if $O_t^{E_1} = O_f^{E_2}$.

Two communication signal processing networks are *dual networks* if they have the same graph and if each element in one graph is the dual of corresponding element in other graph.

The statistical duality definitions which relate functional relationship between corresponding statistics are not given explicitly here.

3. ACTUAL DUAL OF SERIAL DATA TRANSMISSION SYSTEM

The term actual is used here to denote the system which behaves in frequency domain in exactly the same way as the serial system does in time. Regarding data transmission system as a tandem of three signal processing networks representing transmitter, communication channel and receiver, respectively, it turns out that the transmitter and the receiver can be viewed as indirect and direct dual networks of the corresponding networks in the serial system, respectively. For example, if transmitter output signal in the serial system is given by

$$S(t) = \sum_k D_k G(t-kT), \qquad (1)$$

the corresponding signal in the dual system is representable as

$$S(f) = \sum_k D_k G(f-kF). \qquad (2)$$

Complex symbols, envelopes and spectra are assumed troughout the paper.

As far as practical design is concerned, the indirect dual transmitter can be imagined as TDM/FDM transmultiplexer based on cascading IDFT processor with polyphase network as shown in Fig. 1, as a generalization of classical approach to realize standard parallel data transmission system given previously in [4].

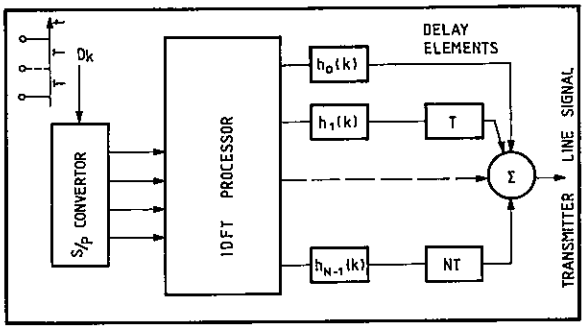


Fig. 1. Transmitter signal syntesis based on cascading IDFT processor and polyphase network

Besides being implititly contained in [5], this approach to realize the generalized parallel data transmission system seems not to be widely exploited.

To realize dual (frequency) behaviour, the reference prototype impulse response shape has to be given by dual function, i.e. network variable, of original (time domain) system impulse response. Choosing the reference filter impulse response to be of the FIR type, desired phase linearity is automatically ensured and undersampled impulse responses $h_1(k)$ are obtainable by direct decomposition [3].

Dual receiver takes the form of network transpose to that one corresponding to the transmitter, i.e. it is given by the connection of polyphase network and DFT processor.

4. DESIGN ASPECTS

As far as "symbol rate", i.e. "symbol spacing" is concerned, it is dictated by practical reasons related to the familiar conceptualization and use of time and frequency notions. Being given by 1/NT, where T represents symbol spacing in original time domain system and N stands for the number of frequency multiplexed symbols, i.e. subchannels in standard parallel system, it reflects these practical constraints. However, the number of frequency domain symbols as high as 512 can be easily supported by current signal

processors. This fact considerably justifies the treatment of standard parallel system as dual of the serial one. This concept can be taken to the advantage in explaining existing and introducing new strategies in standard parallel systems. For example, frequency differential PSK, applied successfully in some modems [9] takes the form of ordinary DPSK. Frequency-domain partial-response signaling scheme, introduced long ago [2] becomes the familiar PRS mehtod of data transmission. Some approaches made to reduce interchannel interference such as introduction of guard-time intervals [4] as well as constraining subchannels spectrum overlapping [5,10], can be understood in their effects by studying original serial system from which the corresponding dual system is derived (Nyquist's criterion, for example).

Here and in what folows, some kind of generalized sampling is assumed. Sampling with periodic δ functions in time domain has its counterpart in DFT coeficients calculation.

5. PERFORMANCE ANALYSIS

As already mentioned, time-frequency duality can be used to predict the performance of dual, and thus of standard parallel system by exploiting some results known for the original system. The concept of network duality, accompanied by the corresponding concept of random process duality [1] can be advantageously applied. Here some examples are given.

As is shown in [1], Gaussian noise is a self dual random process. Using this fact, inherent optimality of linearly independently (and thus orthogonally, too) multiplexed systems with respect to white Gaussian noise becomes evident. In that context, dual channel capacity theorem, deduced in [1], once again poins out the equivalence of these two systems when designed correctly.

By characterizing random channel of the WSS US type by corresponding time-frequency correlation function, known results, relating to the error probability estimation in time domain, [7,8] could be almost directly applied to the dual system with dual channel behaviour, and vice versa.

Having in mind the impact of impulse noise probability density tails on error probability in serial data transmission systems, reduced sensibility of parallel system to this kind of impairment might be explainable by altered type of "amplitude" distribution in frequency domain, i.e. at the output of receive end filter bank. Relying on the central limit theorem, this distribution could be expected to approach Gaussian one, especially when the number of impulsive disturbances in time unit is high. Also, when this specific number is low, the advantage of FDPSK modulation based systems in comparison to TDPSK based one [6], should also be explainable by using some similar duality relationship.

6. COMBATING EFFECTS OF MULTIPLICATIVE NOISE BASED ON USING AUTOMATIC ADAPTIVE EQUALIZATION

Time-frequency selective fading, as well as frequency offset and phase jitter may have disastrous effects on parallel data transmission system performance when the subchannels are closely spaced. And it is this tight subchannel spacing (of the order of 10Hz) which enables majority of parallel system attractive features to be achieved.

Having in mind time frequency duality of the network type, existing between time and frequency variant transmission channels, (modulation and filtering functions, respectively), the effects of multiplicative noise on dual (parallel) system line signal are expected to be the same as are the effects of linear distortion on original serial system one.

Receiver in dual (parallel) data transmission system acts in such a way that the output signal "samples" are ecpressable as

$$\hat{D}(k) = D(k)Z(0) + \sum_{n \neq k} D(n)Z(k-n) \quad (3)$$

Here $D(\cdot)$ denotes an ideal (complex) signal "sample" and $Z(\cdot)$ stands for the complex spectrum of multiplicative process, possibly modified by receiver filter bank.

From (3) it follows that the resulting "intersymbol" interference can be represented, and thus suppressed, in the same way as intersymbol interference, resulting from time spreading (linear distortion) effects, is coped with in standard serial data transmission system.

In this sense we give some computer simulation results relating to the application of linear and decision feedback equalization when multiplicative noise has the form of frequency offset and phase jitter, as thay appear in voice band telephone and HF (Doppler effect) radio channels.

A parallel data transmission system (modulator and demodulator, i.e. transmitter

and receiver) consisting of 266 + 1 subschannels, (1 stands for one reference subschannel), with four-level (FDPSK) modulation has been simulated using 512 - point FFT algorithm. Simulation results, obtained from a limited number of runs, are given in Figs. 2, 3 and Table I. Some comments to them are in order.

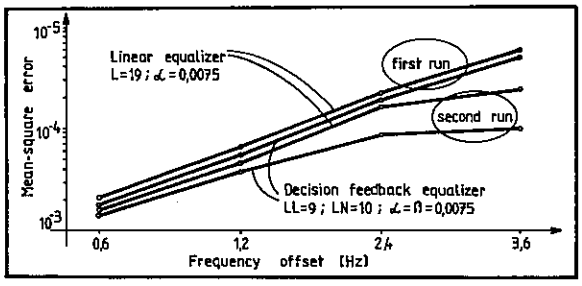


Fig. 2. Mean-square error versus frequency offset

Table I Mean-square error for several combined impairments

S/N (dB)	fofss (Hz)	ajitt (rad)	fjitt (Hz)	FIRST RUN		SECOND RUN	
				linear	non lin.	linear	non lin.
0	0	0,25	60	$5,8 \times 10^{-4}$	$4,9 \times 10^{-4}$	$1,2 \times 10^{-4}$	$1,2 \times 10^{-4}$
0	0	0,25	120	$1,2 \times 10^{-3}$	$1,2 \times 10^{-3}$	$9,2 \times 10^{-4}$	$9,1 \times 10^{-4}$
0	0	0,5	60	$2,1 \times 10^{-3}$	$2,3 \times 10^{-3}$	$4,9 \times 10^{-4}$	$4,7 \times 10^{-4}$
0	0	0,5	120	$4,7 \times 10^{-3}$	$5,2 \times 10^{-3}$	$3,7 \times 10^{-3}$	$3,5 \times 10^{-3}$
0	1,2	0,25	60	$1,0 \times 10^{-3}$	$1,0 \times 10^{-3}$	$1,9 \times 10^{-4}$	$1,5 \times 10^{-4}$
20	1,2	0,25	60	$1,9 \times 10^{-2}$	$2,0 \times 10^{-3}$	$1,2 \times 10^{-2}$	$1,0 \times 10^{-2}$

Because of the limited length of signal sample blocks, equalization has to be performed repeatedly on the same block. For moderate frequency offsets, satisfactory results are obtainable already after the second run, Fig. 2. For higher frequency offsets, equalizer parameters, lengths L for linear, LL and LN (linear and nonlinear part) of decision feedback equalizer, respectively, as well as corresponding adaptation constants α and β, have to be choosen carefully, which imply higher convergence times.

The advantage of nonlinear (decision feedback equalizer, observable in Fig. 2, disappears when phase jitter and additive noise are included, Table I. The known advantage of nonlinear equalizer with respect to input noise enhancement has not been observed during the performed simulation runs, probably because of limited number of used multiplicative noise samples. All other fenomena, encountered in standard equalizer adaptation process, such as tape rotation property, for example, have been recognized here, too.

7. CONCLUSION

In this paper an attempt has been made to point out the usefulness of time-frequency duality concept when applied to design and analyse serial and parallel data transmission systems. The observed possibility of combating the effects of frequency offset and phase jitter using an automatic adaptive equalization procedure has been outlined.

Elaboration of other duality aspects, mentioned here in a rather qualitative manner, should be the object of further research.

REFERENCES

[1] Philip Bello, "Time-Frequency Duality", IEEE Transactions on Information Theory, pp.18-33, Jan. 1964

[2] Pierre Schmid et al., "Frequency Domain Partial-Response Signals for Parallel Data Transmission", IEEE Trans. on Comm.Technology, pp.536-543, Oct. 1969

[3] M.Bellanger and J.Dagnet, "TDM-FDM Transmultiplexer: Digital Polyphase and FFT", IEE Trans. on Comm., pp. 1199-1205, Sept. 1974

[4] S.B.Weinstein and P.Ebert, "Data Transmission by Frequency Division Multiplexing Using the Discrete Fourier Transform", IEEE Trans. on Comm.Technology, pp. 628-634, Oct. 1971

[5] Botaro Hirosaki, "An Orthogonally Multiplexed QAM System Using the Discrete Fourier Transform", IEEE Trans.on Comm. pp. 982-989, July 1981

[6] S.Nedić, "Effects of Impulsive Noise on Data Transmission Using FDPSK", Electronics Letters, 5th July 1984

[7] P.Bello and B.Nellin, "Influence of Fading Spectrum on Binary Error Probabilities of Incoherent and Differentially Coherent Matched Filter Receivers", IRE Trans. on Comm,Systems, pp. 160-168, June 1962

[8] P.Bello and B.Nellin, "Effect of Frequency Selective Fading on the Binary Error Probabilities of Incoherent and Differentially Coherent Matched Filter Receivers", IEEE Trans. on Comm. Systems, pp. 180-186, June 1963

[9] G.Porter, "Error Distribution and Diversity Performance of a Frequency Differential PSK HF Modem", IEEE Trans. on Comm. Technology, Vol. COM-16, No. 4, August 1968

[10] R.W.Chang, "Synthesis of Band-Limited Orthogonal Signals for Multichannel Data Transmission", Bell System Technical Journal, pp. 1775-1796, Dec. 1966

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

NONLINEAR ECHO CANCELLATION AND MULTI-INPUT DISCRETE VOLTERRA SERIES

A. BORYS, W. RUPPRECHT, U. TRICK

Universität Kaiserslautern, Fachbereich Elektrotechnik, Postfach 3049
6750 Kaiserslautern, West Germany

Abstract - This paper deals with the discrete Volterra series for multi-input systems. It is shown here how to obtain such a series for an echo canceller which is used in a two-wire data transmission link, and which, as the practice shows, is a nonlinear system. Previously, this system has been treated in a somewhat artificial way, i.e. by considering the nonlinear echo path and the linear transmission path separately. In other words, no interaction was permitted. In contrast, this paper presents a rigorous mathematical approach to the problem. Also, it is proved here under what conditions the previous method leads to correct results. To make the analysis easier, the "associated" single-input Volterra series is introduced.

1. INTRODUCTION

In the literature [1-3], a problem of nonlinear echo cancellation has been considered which is of interest when data signals are transmitted in both directions over a two-wire link. It has been shown [1], [2] that the response of a nonlinear echo path can be treated as a function of a finite number of the near-end transmitted signal bits. Moreover, the function determining the response of the echo path can be expanded in a special binary series with a finite number of terms [1]. On the other hand, however, the above series is nothing else but a single-input discrete Volterra series representation written down for binary signals [3]. There is some inconsequence in the approaches [1],[3] due to the fact that they assume no interaction between the nonlinear echo path and the linear transmission path. Although it is admitted in [2] that the noise also consists out of products of nonlinear interactions among the far-end signal bits and the intermodulation products, this does not explain sufficiently the assumptions on which the above two methods are based. Here we show that, in reality, the nonlinear echo cancellation system is a nonlinear system with two inputs, see Fig. 1, so it should be analysed using the two-input discrete Volterra series. Note, that now the echo signal also includes, in a natural way, the intermodulation products between the transmitted near-end and far-end signal bits.

2. MULTI-INPUT DISCRETE-TIME VOLTERRA SERIES

In general, a multi-input discrete-time Volterra series can be written as

$$y(k) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_N=0}^{\infty} y_{n_1 \ldots n_N}(k) \qquad (1)$$

where

$$y_{n_1 \ldots n_N}(k) = \begin{cases} h^{(0\ldots0)} & \text{for } n_i=0, i=1,\ldots,N \\ \sum_{i_1=0}^{\infty} \cdots \sum_{i_L=0}^{\infty} h_{i_1 \ldots i_L}^{(n_1 \ldots n_N)} \cdot \prod_{i=1}^{N} \prod^{n_i} x_i(k-i_s) & \text{otherwise.} \end{cases} \qquad (2)$$

In (1) and (2), $y(k)$ and $x(k)$ represent output and input signal, respectively; $L=n_1+\ldots+n_N$ and $\prod^{n_i} x_i(k-i_s)$ means n_i times $x_i(k-i_s)$ where i_s goes through all the indices $i_1, i_2, \ldots, i_L$.

Let us now assume that a system described by eqs.(1) and (2) possesses a finite memory and two inputs (N=2). Then we can write

$$\begin{aligned} y(k) = {} & h^{(00)} + \sum_{i_1=0}^{M_1-1} h_{i_1}^{(10)} x_1(k-i_1) + \\ & + \sum_{i_1=0}^{M_2-1} h_{i_1}^{(01)} x_2(k-i_1) + \\ & + \sum_{i_1=0}^{M_1-1} \sum_{i_2=0}^{M_1-1} h_{i_1 i_2}^{(20)} x_1(k-i_1) x_1(k-i_2) + \\ & + \sum_{i_1=0}^{M_2-1} \sum_{i_2=0}^{M_2-1} h_{i_1 i_2}^{(02)} x_2(k-i_1) x_2(k-i_2) + \\ & + \sum_{i_1=0}^{M_1-1} \sum_{i_2=0}^{M_2-1} h_{i_1 i_2}^{(11)} x_1(k-i_1) x_2(k-i_2) + \ldots \end{aligned} \qquad (3)$$

Eq.(3) means that to determine the output of the system considered, only the last M_1 samples of the input signal x_1, i.e. the samples $x_1(k)$, $x_1(k-1)$, ... , $x_1(k-M_1+1)$, and the last M_2 samples of the input signal x_2, i.e. the samples $x_2(k)$,$x_2(k-1)$,...., $x_2(k-M_2+1)$, are needed. Moreover, note that in general, $M_1 \neq M_2$ in eq.(3).

3. ASSOCIATED SINGLE-INPUT VOLTERRA SERIES

The objective of this section is to find a single-input discrete Volterra series which will be related to the series given by eq. (3). Proceeding intuitively, we could define such a series as

$$y(k) = h^{(0)} + \sum_{j_1=0}^{M-1} h^{(1)}_{j_1}\, x(k-j_1) + \tag{4}$$

$$+ \sum_{j_1=0}^{M-1}\sum_{j_2=0}^{M-1} h^{(2)}_{j_1 j_2} x(k-j_1)x(k-j_2) + \cdots$$

where

$$M = M_1 + M_2; \tag{5a}$$

$$h^{(0)} = h^{(00)}; \tag{5b}$$

and when

a) each of the indices j_1 , ... , j_L is less than or equal to M_1-1

$$h^{(L=n_1)}_{j_1\cdots j_L} = h^{(n_1 0)}_{i_1\cdots i_L}\ ; \tag{5c}$$

note that for this

$$j_p = i_p,\ p = 1\ ,\ \ldots\ ,\ L \text{ holds}; \tag{5d}$$

b) each of the indices j_1 , ... , j_L is greater than M_1-1

$$h^{(L=n_2)}_{j_1\cdots j_L} = h^{(0 n_2)}_{i_1\cdots i_L}\ ; \tag{5e}$$

note that for this

$$j_p = i_p + M_1,\ p = 1\ ,\ \ldots\ ,\ L \text{ holds}; \tag{5f}$$

c) some of the indices j_1 , ... , j_L are less than or equal to M_1-1 and some of them are greater than M_1-1

$$h^{(L=n_1+n_2)}_{j_1\cdots j_L} = h^{(n_1 n_2)}_{i_1\cdots i_L}\ ; \tag{5g}$$

note that now, for those indices $j_1,\ldots,j_L$ which are less than or equal to M_1-1, the relation (5d) holds, and for those which are greater than M_1-1, the relation (5f) should be applied;

and

$$x(k-j_s)=x_1(k-i_s) \text{ for } j_s=i_s=0,1,\ldots,M_1-1 \tag{5h}$$

$$x(k-j_s)=x_2(k-j_s+M_1) \text{ for } j_s=M_1,M_1+1,..,M-1. \tag{5i}$$

In both (5h) and (5i), s=1,2,....,L.

We observe that the series (4) possesses more components than the series (3) for a given order of the nonlinearity $L=n_1+\ldots+n_N \geq 2$. However, all these excessive components are identical with some other components already encountered.

Property 1

The $(j_1,j_2,\ldots,j_n)$-sample of the n-th order impulse response of a single-input system,

$$h^{(n)}_{j_1 j_2 \cdots j_n}\ ,$$

is the same for every permutation of a given set of indices $j_1,j_2,\ldots,j_n$.
Note that the Property 1 generalizes for the multi-input case. Then we can write:

Property 2

The $(i_1,i_2,\ldots,i_{n_1};i_{n_1+1},\ldots,i_{n_1+n_2};\cdots;$ $i_{L-n_N+1},\ldots,i_L)$-sample of the $(n_1,n_2,\ldots,n_N)$ component of the L-th order impulse response of a N-input-system,

$$h^{(n_1 n_2 \cdots n_N)}_{\underbrace{i_1 \cdots i_{n_1}}_{n_1} \cdots \underbrace{i_{L-n_N+1} \cdots i_L}_{n_N}}\ ,$$

is the same for every permutation in the following sets of indices:

$$\{i_1,\ldots,i_{n_1}\},\ldots.$$

$$\ldots\ldots,\ \{i_{L-n_N+1},\ldots,i_L\}\ .$$

Applying Property 2 in (3) and Property 1 in (4), where needed, we finally arrive at the conclusion that both (3) and (4) contain the same set of the "essential terms". (We denote by "essential terms" those terms which are not identical with some others.) However, the output y(k) determined by eq.(4) is not equal to that obtained from eq.(3) in general; we can make these two functions identical by updating the weighting factors $h^{(L)}_{j_1\ldots j_L}$, $L \geq 2$ in eqs.(5).

In order to see how to do this, we consider an example with $M_1=M_2=2$. For this case we must choose

$h_{00}^{(2)} = h_{00}^{(2,0)}$, $h_{01}^{(2)} = h_{10}^{(2)} = h_{01}^{(2,0)} = h_{10}^{(2,0)}$,

$h_{11}^{(2)} = h_{11}^{(2,0)}$, $h_{22}^{(2)} = h_{00}^{(0,2)}$,

$h_{23}^{(2)} = h_{32}^{(2)} = h_{01}^{(0,2)} = h_{10}^{(0,2)}$, $h_{33}^{(2)} = h_{11}^{(0,2)}$,

for which no updating is needed, and

$h_{02}^{(2)} = h_{20}^{(2)} = \frac{1}{2} h_{00}^{(1,1)}$, $h_{12}^{(2)} = h_{21}^{(2)} = \frac{1}{2} h_{10}^{(1,1)}$,

$h_{03}^{(2)} = h_{30}^{(2)} = \frac{1}{2} h_{01}^{(1,1)}$, $h_{13}^{(2)} = h_{31}^{(2)} = \frac{1}{2} h_{11}^{(1,1)}$,

for which updating is performed,

and so on.

We denote the series (4) with the updated coefficients $h_{j_1\ldots j_L}^{L}$ an "associated (with (3)) single-input Volterra series". Note that this associated series can be used in the analysis instead of the series given by (3) because it represents the same output function as the latter.

The associated single-input Volterra series can be, after eliminating all the identical terms, represented in the following form [3]:

$$y(k) = d^{(0)} + \sum_{j_1=0}^{M-1} d_{j_1}^{(1)} \times (k-j_1) +$$

$$+\ldots+ d_{012\ldots(M-1)}^{(M)} x(k)\, x(k-1)\ldots x(k-M+1) \qquad (6)$$

for binary signals. The coefficients $d^{(0)}$ and $d_{j_1 j_2 \ldots j_n}^{(n)}$, n=1,2,...,M, in (6) represent the new weighting factors.

4. APPLICATION OF ASSOCIATED VOLTERRA SERIES TO NONLINEAR ECHO CANCELLATION

In a realistic model of the nonlinear echo cancellation at transmission of data signals, one has to take into account the fact that both paths: echo path and transmission path "interact" with each other. This "interaction", which is responsible for the intermodulation products between the far-end and near-end transmitted signals, can be illustrated as shown in Fig.1.

The nonlinear two-input system in Fig.1 can be described by the discrete two-input Volterra series (3), or equivalently by the associated single-input Volterra series which is used in what follows. Additionally, with a(k) and b(k) representing the binary signals, the associated Volterra series assumes the form (6)

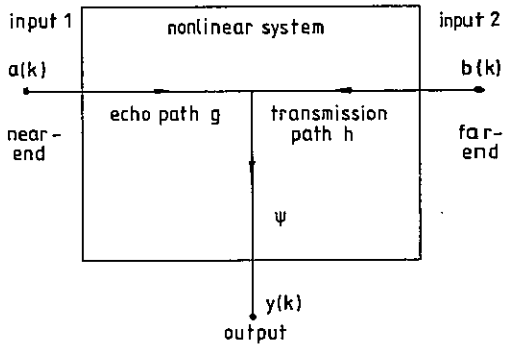


Fig. 1.

Let us now describe the signals and quantities pertaining to the scheme of Fig.1 in a vector form. Thus we have:

1. the transmitted near-end data vector

$$\underline{a}_k = [a(k),\ldots,a(k-M_1+1)]^T ,$$

2. the transmitted far-end data vector

$$\underline{b}_k = [b(k),\ldots,b(k-M_2+1)]^T ,$$

3. the auxiliary transmitted data vector

$$\underline{x}_k = [a(k),\ldots,a(k-M_1+1),\ b(k),..,(k-M_2+1)]^T,$$

4. the "nonlinear" auxiliary transmitted data vector

$$\underline{u}_k = [1,a(k)a(k-1),\ldots,a(k)a(k-M_1+1),a(k)b(k),.$$
$$\ldots,a(k)a(k-1)\ldots a(k-M_1+1)b(k)b(k-1)\ldots$$
$$\ldots b(k-M_2+1)]^T ,$$

5. the echo path vector associated with its linear part

$$\underline{g} = [g(0),\ldots,g(M_1-1)]^T ,$$

6. the transmission path vector associated with its linear part

$$\underline{h} = [h(0),\ldots,h(M_2-1)]^T ,$$

7. the vector pertaining to the nonlinear parts of the echo and transmission paths

$$\underline{\Psi} = [g_{00}, g(0_a,1_a), \ldots, g(0_a,(M_1-1)_a),$$

$$g(0_a,0_b), \ldots, g(0_a,1_a,\ldots,(M_1-1)_a,0_b,1_b,\ldots,$$

$$(M_2-1)_b]^T ,$$

8. the vector of the digital adaptive transversal filter coefficients

$$\underline{c}_k = [c_0(k), \ldots, c_{M_1-1}(k)]^T .$$

Using the Volterra series given by (6) and the definitions stated above, we can express the echo signal of the linear canceller, e(k), in the form

$$e(k) = y(k)-v(k) = \underline{a}_k^T \cdot \underline{g} + \underline{u}_k^T \cdot \underline{\Psi} \qquad (7)$$

where

$$v(k) = \underline{b}_k^T \cdot \underline{h} + n(k) \qquad (8)$$

n(k) in (8) represents the noise signal.

Moreover, the residual signal is then given by

$$r(k) = y(k) - \hat{e}(k) =$$

$$= \underline{x}_k^T \cdot \begin{bmatrix} \underline{g} \\ \underline{h} \end{bmatrix} + \underline{u}_k^T \cdot \underline{\Psi} + n(k) - \hat{e}(k) \qquad (9a)$$

or

$$r(k) = \underline{a}_k^T(\underline{g}-\underline{c}_k) + \underline{u}_k^T \cdot \underline{\Psi} + v(k) \qquad (9b)$$

In (9a), $\hat{e}(k)$ is of the form $\hat{e}(k) = \underline{a}_k^T \cdot \underline{c}_k$

(9c)

Observations:

1. Note that the echo signal determined by eq. (7) consists of:

   a) the "linear" echo $\underline{a}_k^T \cdot g$,

   b) the dc component g_{00},

   c) the components resulting from nonlinear interactions among the near-end transmitted data bits, for example, $g(0_a,1_a)a(k)a(k-1)$,

   d) the components resulting from nonlinear interactions among the far-end transmitted data bits, for example, $g(0_b,1_b)b(k)b(k-1)$,

   e) the intermodulation-type products, for example, $g(0_a,0_b)a(k)b(k)$.

2. Note that the above formulation of the problem corresponds with that in [3]. The only difference is that we have here $\underline{g}$ instead of $\underline{g}_1$ and $\underline{\Psi}$ instead of $\underline{g}_n$. Moreover, the function R_N defined in [3] has now a new form, namely:

$$R_N = R_{DC} + R_{AA} + R_{BB} + R_{IM} \qquad (10)$$

where

$$R_{DC} = g_{00}^2 \qquad (11)$$

$$R_{AA} = g^2(0_a,1_a)+\ldots+g^2(0_a,1_a,\ldots,(M_1-1)_a) \qquad (12)$$

$$R_{BB} = g^2(0_b,1_b)+\ldots+g^2(0_b,1_b,\ldots,(M_2-1)_b) \qquad (13)$$

$$R_{IM} = g^2(0_a,0_b)+\ldots+g^2(0_a,1_a,\ldots,(M_1-1)_a, 0_b,1_b,\ldots,(M_2-1)_b) \qquad (14)$$

Note that the functions R_{DC}, R_{AA}, R_{BB} and R_{IM} pertain to the components b), c), d) and e), respectively, mentioned in Observation 1. Moreover, observe that the approaches [1] and [3] are correct when the following inequality

$$R_{DC} + R_{AA} \gg R_{BB} + R_{IM} \cong R_{IM} \qquad (15)$$

is satisfied.

REFERENCES

[1] Agazzi, O., Messerschmitt, D.G. and Hodges, D.A., IEEE Trans. Commun. (1982) 2421.

[2] Agazzi, O., Hodges, D.A. and Messerschmitt, D.G., IEEE Trans. Commun. (1982) 2095.

[3] Borys, A., Rupprecht, W. and Trick, U., IEEE Trans. Commun., submitted for publication.

Acknowledgement

The first author would like to thank the Alexander von Humboldt-Foundation, Bonn, West Germany for financial support.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

APPLICATION OF DIGITAL SIGNAL PROCESSING TO PREVENTION OF HOWLING IN HANDSET-FREE TELEPHONES

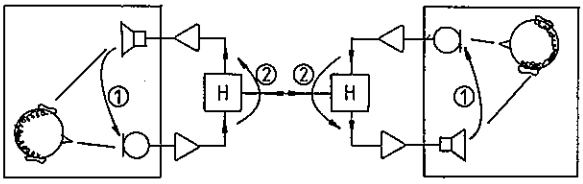
B. Hätty and J. Sitzmann

Institut für Netzwerk- und Signaltheorie
Fachgebiet Theorie der Signale
Technische Hochschule Darmstadt
D-6100 Darmstadt, West-Germany

This contribution deals with measures stabilizing the acoustical feedback loop in a handset-free telephone system. Adaptive compensation of the electrical near-end echo, frequency shifting of the loudspeaker input and, finally, insertion of speech controlled attenuation are applied. The general purpose signal processor MB 8764 by Fujitsu has been used for implementation.

1. INTRODUCTION

In modern communication systems use of "hands-free" telephones is desireable. Such devices place microphone and loudspeaker (see figure 1) in front of the subscriber instead of combining them into a handset.



1 = acoustical feedback
2 = electrical feedback
H = hybrid circuit

Fig. 1: "hands-free" telephone link

Electrical and acoustical feedback, however, (figure 1) leads to echoes that in turn may cause instabilities, noticeable as the well known howling. Strong electrical (near-end) echoes of several milliseconds duration [1] are caused by the hybrid circuit at the two-wire/four-wire junction due to the impedance mismatch in real systems. Furthermore, the acoustical coupling between microphone and loudspeaker at the location of the distant subscriber may lead to echoes 10...100 times longer than the electrical ones [2]. Therefore, a "hands-free" telephone must be equipped with measures to stabilize the feedback loop caused by the above mentioned echo paths.

On principal, loop gain can be reduced by using sound absorbing material together with microphones and loudspeakers exhibiting specially shaped response or radiation characteristics. Only in special cases like lecture halls and teleconference studios, however, such measures are economical and feasible

Existing "hands-free" telephones are furnished with so-called echo suppressors: Speech controlled loss devices attenuate one of the communication paths in order to reduce feedback and to guarantee loop stability. In fact, in currently available echo suppressors [3] the required high attenuation of these loss devices permits only "half-duplex" transmission. This means that simultaneous talking of both subscribers is not possible. In particular, it is not possible for the listener to interrupt the talker. Furthermore, in noisy environment the detection of the active subscriber by the echo suppressor may be unreliable. Therefore, parts of fluently spoken words may be clipped.

Instead of reducing loop gain by measures as mentioned above, one could stabilize the feedback loop and prevent howling by the compensation of all echoes. In contrast to the short electrical echoes caused by the hybrid circuit the compensation of the acoustical echoes is much more complex. The enclosures considered in this paper are situated in an office environment. The acoustical properties of these enclosures are time-varying and lead to long acoustical echoes. Their compensation requires adaptive FIR filters with at least 1000 coefficients [2]. The implementation of such adaptive compensators is not yet solved adequately.

In the following contribution an approach is proposed based on the idea of reducing the electrical and acoustical echoes such that only a low additional attenuation is required. Thus, transmission will no longer be limited to "half-duplex". The measures applied are described in section 2. A principle of a speech control which is rather insensitive to noisy environment is explained in section 3. Finally,

section 4 presents an implementation of the described concept on the commercially available digital signal processor MB 8764.

2. REDUCTION OF ECHOES

2.1. Frequency Shifting

According to a previous proposal [4], an improvement of acoustical feedback stability can be achieved by inserting a frequency shifter (typically 5 Hertz) into the feedback loop. For illustration, figure 2 shows the frequency response of an office room measured between microphone output and loudspeaker input.

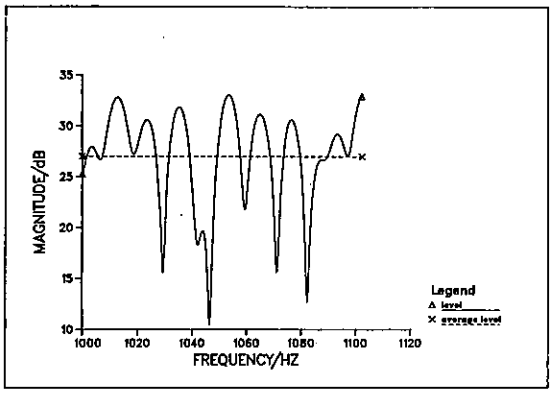
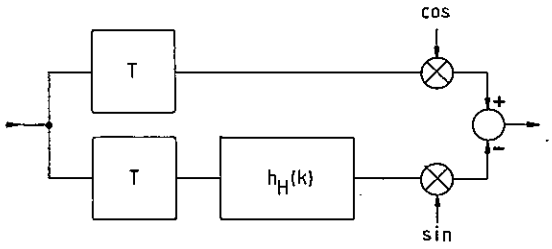


Fig. 2: frequency response of an office room

This function exhibits the typical "interference pattern" of minima and maxima of multipath propagation systems like the configuration (loudspeaker – enclosure – microphone – system) we are considering. In case of frequency shifting, the loop gain is determined by the average magnitude of the frequency response. The difference between maximum and average levels (max. 10 decibels) represents a gain reduction in the acoustical feedback loop. For more information see [4].

The required frequency shifter is implemented by a single sideband modulator based on a discrete Hilbert filter as shown in figure 3. The degree of the (nonrecursive) Hilbert filter $h_H(k)$ is chosen as to provide properties like sufficient crosstalk attenuation of the suppressed sideband. A design procedure and a table of filter coefficients are given in [5]. Overall, frequency shifting including the generation of the orthogonal carriers can be implemented digitally at low complexity. A shift magnitude of 5 Hertz proved to yield a maximal increase of 10 decibels in stability at negligible speech distortions.



T = delay

Fig. 3: frequency shifter

2.2 Compensation of the Electrical Echoes

The wide spread of telephone link parameters affect the quality of the hybrid balance, causing electrical echoes within a wide level range. Such, using "hands-free" telephones, instabilities can occur in the feedback loop at each subscriber set. In order to avoid those instabilities the electrical echoes have to be cancelled. Due to the wide range of line parameters, adptive compensators are required (figure 4). In contrast to the acoustical echoes mentioned in the introduction, the short electrical echoes can sufficiently be eliminated by a low-order adaptive FIR filter.

The estimated echo z[k] is computed by convolving the microphone signal x[k] with the impulse response of the adaptive FIR filter $c_i[k]$ according to

$$z[k] = \sum_{i=0}^{N-1} c_i[k]\, x[k-i] \qquad i = 0,\ldots,N-1$$

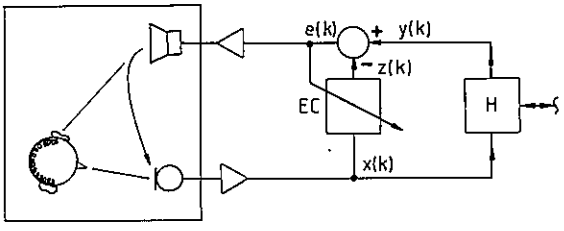
and it is subtracted from the hybrid echo y[k]. The filter coefficients $c_i[k]$ are adjusted by the well known LMS algorithm [6]:

$$c_i[k+1] = c_i[k] + \frac{a[k]\, e[k]}{\sum_{j=0}^{N-1} x^2[k-j]}\, x[k-i]$$

where a[k] is used to control the convergence speed.

Reduction of a[k] prevents misadjustment of the filter coefficients in case of signals generated by the distant subscriber. The speech detectors, described in the next section, may

perform this task without increase in complexity. However, further investigations have proved that after call setup, line parameters do not vary during the following telephone conversation. Therefore, adjustment is only necessary at the time the telephone link is established. A short training sequence – normally white noise – may be used and the coefficients $c_i(k)$ may be frozen afterwards.



EC = echo compensator
H = hybrid circuit

Fig. 4: echo compensator

Using an adaptive FIR filter with 32 coefficients, the echo compensator guarantees an attenuation of about 30 decibels.

3. ATTENUATION OF THE RESIDUAL ECHOES

In order to limit residual echoes, variable loss devices are inserted into the loudspeaker and microphone path (see figure 5). The increase in feedback stability achieved by the procedures discussed in the foregoing section allows a proportional decrease in total attenuation of these loss devices.

Depending on the activities of both parties, the total attenuation has to be split up between the loudspeaker and the microphone path. Neglecting transitions, two states of splitting are possible:

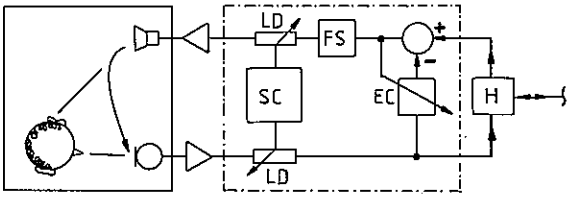
- only one subscriber is talking: his microphone path may not be attenuated so that the total attenuation has to be inserted into the loudspeaker path and vice versa.
- in the case of no activities or doubletalk the total attenuation has to be split up equally on the microphone and the loudspeaker circuits.

Signal levels within a wide range and noisy environment are not allowed to impair the proper detection of these states. From this, the following structure can be derived.

In order to reduce the dynamic range of the input signals an A-law compander is implemented. The following speech detector is based on the burst behaviour of human speech which differs from (quasi-) stationary noise signals. In order to detect bursts, three time- windows of different lengths (about 1 millisecond to 16 seconds) are implemented by low order recursive filters [7,8]. Comparing the filter outputs e.g. comparing signal powers averaged over three windows of different lenghts allows fast speech detection. Depending on the decision of both speech detectors, the variable loss devices in the microphone and the loudspeaker path are adjusted as explained above. The transitions between the different states are smoothed in order to avoid inconvenient fading.

4. IMPLEMENTATION OF A "HANDS-FREE" TELEPHONE ON THE DIGITAL SIGNAL PROCESSOR MB 8764

The general structure of the concept proposed is shown in figure 5. The components within the dashed line are implemented on the digital signal processor MB 8764. The signal processor is interfaced by multiplexed A/D and D/A converters to an inhouse telephone system and a "hands-free" telephone set consisting of a loudspeaker and a microphone. The input signals are sampled at a rate of 8 kHz and converted with a linear resolution of 12 bit. In order to avoid local instabilities of the subscriber set, the hybrid echoes are cancelled by the echo compensator EC corresponding to figure 5. The frequency shifter FS and speech controlled loss devices LD together guarantee stable operation under a wide range of external conditions.



SC = speech control EC = echo compensator
FS = frequency shifter H = hybrid circuit
LD = loss device

Fig. 5: "hands-free" telephone

In order to point out problems in context with a real-time implementation of the needed algorithms, some relevent features of the MB 8764 are presented. The MB 8764 is a pipelined two address machine of the "Harvard" type. This means that in every cycle (100 nanoseconds cycle time) the instruction code and two operands are fetched and simultaneously one

arithmetic operation (e.g. multiplication and accumulation) is evaluated. Due to this internal structure, an efficient evaluation of the scalar product required for non recursive filters is possible. Therefore, FIR filters have been implemented for most signal processing tasks except the power estimation where recursive structures have been implemented. Thus, wasting of processor storage by high-order FIR filters has been avoided. These recursive filters, however, require 32 bit fixed-point arithmetic in order to reduce roundoff errors. In comparison with the implementation of FIR filters with 16-bit fixed point arithmetic processor power is saved nevertheless. Additionally, storage demand has been reduced by the use of a specific algorithm [9] calculating the samples of the two carriers required for frequency shifting (section 2.1).

Finally, all algorithms can be implemented by one processor [10]. The hardware can be placed on a single board (160 mm * 100 mm).

5. CONCLUSION

The entire system has been tested and verified under various conditions (e.g. different line parameters) with the aid of an inhouse telephone system. Informal tests using tape-recorded telephone conversations show that doubletalk is possible in a satisfactory way. In particular, the talker will always recognize an interruption by the listener.

Further investigations showed that it is necessary to adapt the dynamic range of the input signals to the A/D converter by an automatic gain control (AGC) device. Finally, the concept proposed may be expanded to subbands split up by filter banks based on polyphase networks (PPN).

ACKNOWLEDGEMENTS

These investigations were supported by TELENORMA (TN). The authors thank their colleagues for inspiring discussions and critical remarks.

REFERENCES

[1] Sondhi, M.M., Berkley, D.A., Proc. of the IEEE (1980) 948.
[2] Becker, T., Hänsler, E., Schultheiss, U., Frequenz (1984) 142.
[3] MC 34018, Motorola Inc. (1985).
[4] Schroeder, M.R., The Journal of the Acoustical Society of America (1964) 1718.
[5] Herrmann, O., AEU (1969) 581.
[6] Widrow, B., Proc. of the IEEE (1975) 1692.
[7] Rabiner, L.R., Schafer, R.W., Digital Processing of Speech Signals (Prentice Hall, Eaglewood Cliffs, N.J., 1983).
[8] Barnwell, T.P., ICASSP (1977) 1.
[9] Schüßler H.W., Digital Signal Processing 2 (in German, Erlangen, 1983).
[10] Sitzmann, J.R., Diplom Thesis (Darmstadt, 1985).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

LEAST-SQUARE CANCELLATION DECISION FEEDBACK RECEIVER

Arie Reichman

Tadiran Ltd.
Communication Division
P.O. Box 267, 58102 Holon, Israel

1. INTRODUCTION

The popular adaptive equalizers in data communication are constrained in two ways. First, the configuration is determined by the complexity involved. The preferred equalizers are linear or linear feedback. Second, the most meaningful quality criterion, which is the minimum probability of error, is conventionally considered to be impractical for adaptive applications. Other, more convenient, criteria have found widespread use in optimizing the equalizer, i.e., the peak distortion criterion and the mean-square-error criterion [1].

In Forney's classic paper [2], he presented a nonadaptive receiver designed to attain minimum probability of error. In the present paper, an equivalent configuration is presented. The implementation of the optimal receiver in either form requires the knowledge of the channel impulse response and of the statistics of the noise. When the channel and the statistics of the noise are not known and may change slowly in time, the optimal receiver presented here becomes adaptive by the use of least-square (LS) filters.

Lattice filters are a suitable form of implementation of the LS filters for the adaptive receiver, because of their fast convergence and the moderate amount of computation, which is of the same order of magnitude as the popular LMS algorithm [3].

2. SIGNAL MODEL AND THE OPTIMAL RECEIVER

The algorithm to be employed here is time-discrete, and hence for simplicity is assumed that the signal observed by the receiver is also generated in time discrete fashion. A reasonable model for the bit rate sampling of the receiver's in-phase and quadrature channel signals is given by complex baseband sequence y(t), with t restricted to integer values.

$$y(t) = c(t) * d(t) + n(t)$$

where * denotes convolution c(t) is the combined impulse response of the transmitter, channel and receiver, n(t) is zero near Gaussian noise, and d(t) is the ±1 data sequence.

The minimum probability of error is obtained by the maximum likelihood receiver, which is conventionally implemented by choosing the smallest likelihood variable computed as shown in Figure 1.

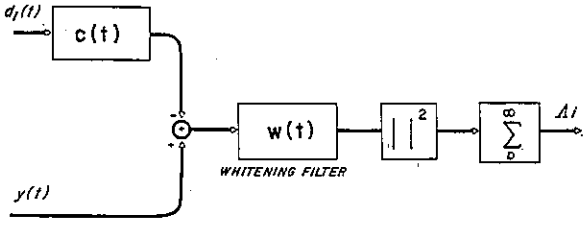


Figure 1. Calculation of the Maximum Likelihood Variable

The likelihood variable is obtained by:

a. Subtracting from the received signal a filtered version of an assumed data sequence, $d_i(t)$.
b. Whitening the difference.
c. Squaring.
d. Integrating.

The estimated data is the assumed data sequence corresponding to the smallest likelihood variable calculated for each possible data sequence. The number of processors would seem to be huge, but a practical version of this receiver can be implemented in a much simpler manner, using the Viterbi Algorithm (VA)[2].

3. THE ADAPTIVE RECEIVER

The implementation of the optimal receiver requires the knowledge of the channel impulse response c(t) and the characteristics of the whitening filter w(t), which are related to the statistics of the noise. When the channel and the statistics of the noise are not known and may vary slowly in time, we can use the LS algorithms to find these filters and adapt them to the changing conditions. The LS algorithms are discrete; hence, the input signal y(t) must be discrete or discretized.

Two types of LS filters are necessary for the implementation of the adaptive receiver [4]:

a. The LS predictor can be used for whitening a signal.
b. The LS joint process estimator (JPE) can be used as a channel identifier.

The architecture of the LS Lattice filter is shown in Figure 2.

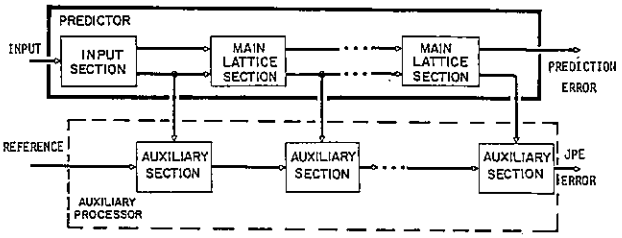


Figure 2. Architecture of LS Lattice Filter

The main lattice sections perform the prediction and can be used as a predictor. The auxiliary processors (AP) estimate a joint process, and together with the predictor, form the JPE.

An adaptive version of the optimal receiver described in the previous section is obtained using the processor shown in Figure 3 for calculation of the maximum likelihood variable. The processing consists of the following:

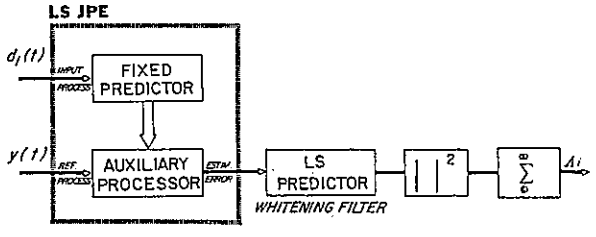


Figure 3. Adaptive Calculation of the Maximum Likelihood Variable

a. JPE, whose coefficients are computed according to the input and reference processes. The input process is the assumed data sequence, and the reference process is the received signal. In the common case when the data d(i) is a sequence of independent identi- cally distributed random variables it can be easily shown that the predictor of the JPE is simply a shift register containing the assumed present and last M symbols, where M is the memory of the channel. In this case, the JPE algorithm computes only the coefficients of the auxiliary processor.

b. LS predictor, which performs whitening of the noise. This filter is required only when the input noise is not white.

c. Squaring of the predictor output and summation over one symbol duration interval.

Again, a simplification of receiver can be obtained by a VA receiver with properly defined states and matrices. In this paper we choose to further simplify the receiver with some degradation in performance.

4. THE CANCELLATION DECISION FEEDBACK RECEIVER

The simplified receiver, shown in Figure 4, is obtained by taking the decision without delay, comparing the output of two processors which have the same initial conditions (the same coefficients and the same assumed previous data), but differ in the present bit assumed.

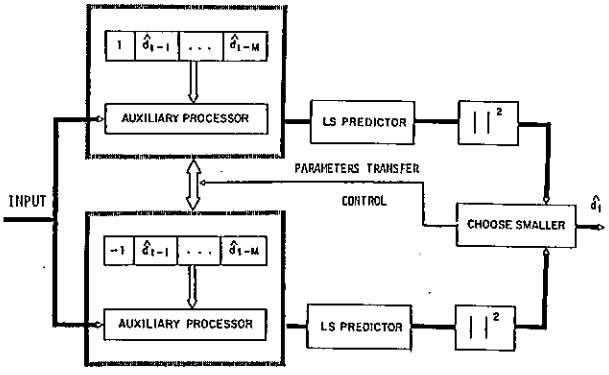


Figure 4. Adaptive CDF Receiver

After the computations related to a specific input y(t) are completed, the parameters of the AP in each processor become different because of the different assumed present bits. In accordance with the outputs of the processor, a decision is made concerning the value of the present data bit. Upon making the decision $\hat{d}$:

a. The new decision is loaded into the JPE state register and the previous decisions are shifted.

b. The processor for $d \neq \tilde{d}$ is reinitialized with the current coefficients of the AP of the d processor.

Hence, if no decision error is made, properly loaded processors are maintained up to the current bit being received, that is, the equalizer system is decision directed and the processors reduce the effect of previous bits. The receiver therefore, is called Cancellation Decision Feedback or CDF.

5. CONVERGENCE WITH AND WITHOUT A TRAINING SEQUENCE

Most equalization systems employ a training sequence (TS) such that the receiver "learns" the channel during the training; afterwards the unknown data is sent. The adaptive VA receiver operation, as has been described so far, does not require any TS.

A training mode (TM) of operation may be added before the regular mode. Only one adaptive processor is required during the TM, where the data sequence function d(t) is the TS known by the receiver as shown in Figure 5. During the TM, the auxiliary processor parameters adjust to the appropriate values. When the TM ends and the system enters the normal operating mode, the parameters of the filters are transferred to all the processors, and the regular adaptation proceeds.

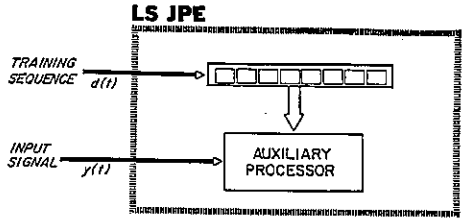


Figure 5. Training Mode of Operation

Based on simulations, it has been observed that for some channels the TS is not required and the algorithm converges in several tens of symbols to the appropriate coefficient. On the other hand, a TS is required for other channels, otherwise, the algorithm converges to the wrong set of coefficients. For example, without a TS for the channel $c_0 = 1$, $c_1 = 2$, and $c_2 = 1$ the coefficients of the AP may converge either to $c_0 = 2$, $c_1 = 1$, $c_2 = 0$ or to the correct set. Hence, in this case, c_0, the estimated first coefficient, may converge to the largest coefficient instead of the first coefficient.

From simulations, it is conjectured that there is no need for a TS and that the algorithm converges to the correct coefficients for a channel satisfying:

a. the open eye condition:

$$|c_0| > \sum_{i=0}^{\infty} |c_i|,$$

b. and causality: $c_i = 0 \ @ \ i < 0$.

For other channels, a TS is recommended.

6. SIMULATION RESULTS

We simulated a discrete channel which has a sampling rate equal to the data rate. The data is a binary sequence of i.i.d. r.v., the channel has an impulse response given by:

$$c(t) = a_0\delta(t) + a_1\delta(t-1) + a_2\delta(t-2)$$

and the additive noise is white.

Two sets of a_i coefficients were checked for the CDF receiver and their results compared with the non-adaptive receiver which has complete knowledge of the channel impulse response. For the first set of coefficients, $a_0=1$, $a_1=0.5$, $a_2=0.25$, the "open eye" condition holds and a TS is not required. For a second set, $a_0 = 1$, $a_1 = 2$, $a_2 = 1$, a training sequence is required. The results are given in Figure 6.

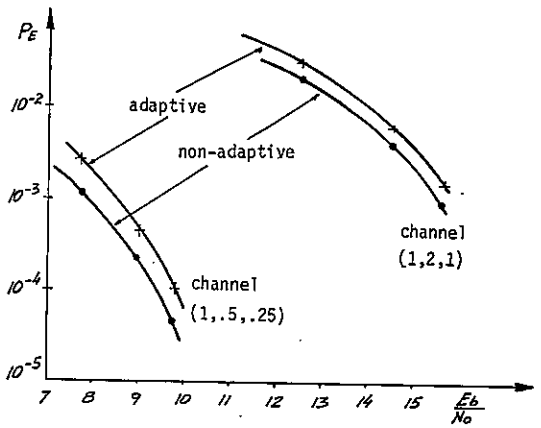


Figure 6. Simulation Results

Differential encoding was required for the adaptive receivers without a TS and was used in all simulations. The reason is that each equalizer has two stable states. In one we obtain the correct output; in the other the opposite, namely, a completely inverted output. It may happen that the equalizer switches states, that is, at some point the

output data sequence changes polarity from the correct to the "inverted" sequence. This effect is overcome by the use of a differential data encoder. Hence, one decision error becomes two data errors; if the equalizer changes state, one data error occurs. For high signal-to-noise ratios equalizer state changes are quite rare, thus the price we pay by using differential encoding is a multiplication of the probability of error by 2.

7. CONCLUSIONS

The adaptive receiver uses least-square lattice filters to perform equalization and noise cancellation. The adaptive receiver converges very quickly to an optimal receiver. For an "open eye" channel, the adaptive receiver requires neither the knowledge of the noise statistics and the impulse response of the channel, nor a training sequence of known data bits. For other channels, a short training sequence is required. The receiver also adapts rapidly to changes in the impulse response of the channel or in the statistics of the noise.

REFERENCES

[1] Proakis, J.G., Digital Communications, (McGraw-Hill, 1983).
[2] Forney, G.D., Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference, IEEE Trans. Inform. Theory (May 1972).
[3] Friedlander, B., Lattice Filters for Adaptive Processing, Proc. IEEE, (August 1982).
[4] Reichman, A., and Scholtz, R.A., Adaptive Spread-Spectrum Systems using Least-Square Lattice Algorithms, IEEE Journal on Selected Areas in Communication, (Sept. 1985).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

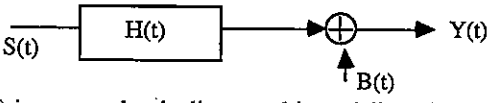
SOME IMPROVED ADAPTATIVE ALGORITHMS IN DIGITAL UNDERWATER ACOUSTIC CHANNEL EQUALIZATION.

Raphaël LUCAS (*) - Jacques MARTIN (**)

In this paper some adaptative algorithms in an equalization problem in underwater acoustics context are proposed. All the procedures take control of parameters evolution and, being data-dependent, nice trade offs between convergence rate and misadjustment error are obtained. Our contribution consists in making some modifications in the so-called L.M.S., D.S.D. and L.S.R. algorithms. The resulting procedures are full adaptive in essence and in their parameters and they will be reported as : N.L.M.S. (Normalyzed Least Mean Squares), A.D.S.D. (Adaptative Differential Steepest Descent), A.L.R.S. (Adaptative Linear Random Search). The algorithms have been tested with success.

INTRODUCTION

In many propagation problems (submarine acoustic communication, ionospheric or radio-urban communication) [1], [2] besides additive noise the channel often exhibits multipath propagation, which destroys the quality of the link. We can schematize the channel as follow :



H(t) is supposed to be linear and it modelizes the channel response without noise. If the channel fluctuations are shorter than the elementary digit period of the transmitted signal, the adopted model for describing the impulse response H(t) is random and we usually choose a Detection procedure [3]. When the fluctuations are not so rapid (a few elementary digits periods) we can use equalization techniques which have successful and wide experience in telephonic channel [4]. The aim of this work is an adaptive F.I.R. filter able to be piloted by different algorithms. Three brand-new algorithms are presented based on classical ones but with some modifications in order to enhance both performance and robustness. In the case of a two paths channel we can simplify the structure of the filter and by untroducing an estimate of the delay between the two paths we can inprove the convergence. Then the adaptive filter is tested in a real environment.

1. CHANNEL MODEL AND EQUALIZER STRUCTURE

1.1. Channel model

Let us assume that the received signal Y(t) is composed by the sum of several delayed transmitted signals

$$Y(t) = \sum_{i=0}^{P} a_i \; s(t - t_i) + B(t) \qquad (1)$$

Although the methods presented here are quite general, we use only a two paths model which is a very frequent practical situation. We will see later that in this case the procedure can run faster. So we consider the following digital model :

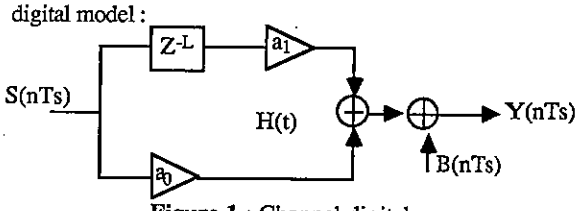


Figure 1 : Channel digital

T_s is the sampling period, LT_s is the delay between the two paths. The transmitted signal is generally modulated (with a carrier in the range of kHz for submarine acoustic medium) and the proposed method is assumed to work after demodulation so the parameters a_0, a_1 are complex.

1.2. Equalizer structure

If no noise is present, the optimum filter in order to recover S(n) from Y(n) is the inverse filter with the following transfer function :

$$h(z) = \frac{1}{a_0 + a_1 z^{-L}} = \frac{1}{a_0 (1 + R z^{-L})} \qquad (2)$$

A FIR filter $\hat{h}(z)$ can be derived from (2) by expanding h(z) and truncating its impulse response

$$\hat{h}(z) = 1/a_0 \, (1 - R z^{-L} + R^2 Z^{-2L} + \ldots + (-1)^Q R^Q Z^{-QL}) \qquad (3)$$

$$R = a_1 / a_0 \qquad |R| < 1 \qquad (4)$$

From (3) it is easy to see that the gains of the filter appear every L samples and equal zero for the other samples. When $|R| > 1$ or when noise is present, the structure does not change [5] and we need only (see fig. 2-b) a row of gains w_i distant from each other by the delay L.

The receiver block diagram and the Q coefficients adaptive equalizer that can be used are shown in figure 2 .

d(n) is a reference signal used at the beginning of the communication. Usually d(n) is a pseudo-random recurring sequence [6]. The system must work in two separate phases, the learning (L) and the commucation (c) phases :

a) **Learning phase** : during this phase, the transmitter sends the known sequence d(n). Three operations are achieved :

- the parameters of H(t), L, a_0, a_1 or the number of paths if

(*) D.T.T.I. - E T S E T - Jorge Girona Salgado - s/n - BARCELONA 34 - SPAIN
(**) CEPHAG - E N S I E G - B.P. 46 - 38402 St Martin d'Hères - FRANCE.

they are more than two as well as their strengths are estimated by a correlation system described later.
- the number of coefficients Q is selected with regards to the power ratio of paths (Q increases with | R |)
- starting from an initial set of weights, the optimum filter is approximated by the adaptive algorithm.

When the error is small the next phase is entered in.

b) Communication phase : the unknown message which must be composed with a finite alphabet is now sent. The received signal is equalized by the adaptive filter and the message is decoded. d(n) is then replaced by a decision on the estimated signal.

According to the error power, the logic can stop the adaptive filter and the learning phase can be reassumed.

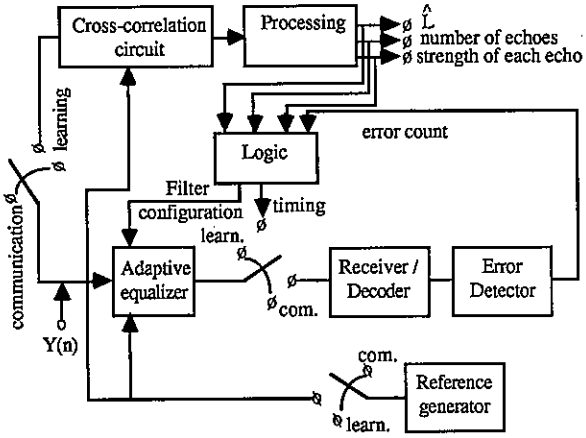


Figure 2-a : "Simplified receiver block diagram".

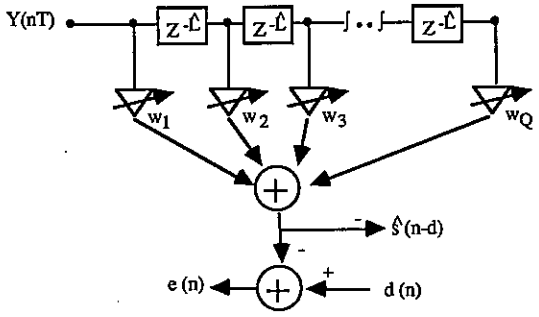


Figure 2-b : "Adaptive equalizer"

Note that a change of the number of paths causes the structure to be quite different. We then use a more flexible structure with an error margin allowed around $\hat{L}$. Figure 3 shows a structure with a margin of two cells :

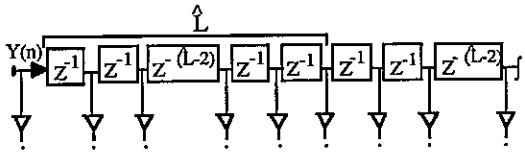


Figure 3 : Structure with error on $\hat{L}$

2. ADAPTIVE ALGORITHMS

Three algorithms are presented which are modified versions of classical ones [7]. Two of them are based on the steepest Descent method : the NLMS and the ADSD ; the third LRS one is a Random Search family algorithm.

The error signal (Fig. 2-b) is given by :

$$e(n, k) = d(n) - \underline{W}^T(k) \bullet \underline{Y}(n) \quad (5)$$

where n is the temporal index, k the current iteration index.

$$\underline{W}^T(k) = (w_1(k), w_2(k) \ldots w_Q(k)) \quad (6)$$

$$\underline{Y}^T(n) = ((y_1(n), y_2(n) \ldots y_Q(n)) \quad (7)$$

$$y_1(n) = y(n) \quad y_i(n) = y_i [n - (i-1)\hat{L}] \quad (8)$$

The problem is to find the weight vector $\underline{W}$ which minimizes the mean square error : $\xi(k) = E\{| e(n, k) |^2\}$ (9)

This leads to the so called WIENER solution [7]

$$\underline{W}opt = \underline{W}_{WIENER} = \underline{R}_{yy}^{-1} \bullet \underline{P} \quad (10)$$

$$\underline{P} = E \{ d^*(n) \bullet \underline{Y}(n) \} \quad (11)$$

$$\underline{R}_{yy} = E \{ Y^*(n) \bullet \underline{Y}^T(n) \} \quad (12)$$

The classical steepest descent method is formulated as follows :

$$\underline{W}(k+1) = \underline{W}(k) - \mu \nabla_{\underline{W}} \xi(k) \quad (13)$$

where the scalar factor μ takes control of the adaptation step size and the gradient is taken with respect to the complex conjugated weights [8]. Many algorithms can be derived from (13) since two terms have to be estimated

- the m.s.e. $\xi(\bullet)$
- the gradient of this m.s.e.

2.1. Normalized Least Mean Squares (NLMS)

In the LMS algorithm the m.s.e. estimate is the instantaneous error and the gradient used is the true one.

$$\hat{\xi}(k) = | e(n, k) |2 \quad (14)$$

therefore :

$$\underline{W}(k+1) = \underline{W}(k) + \mu e(n) \underline{Y}^*(n) \quad (15)$$

The final weight vector will defer from the optimum $\underline{W}_{opt}$ and we can define the misadjustment M as :

$$M = E\{ \text{excess m.s.e.} \} / (\text{m.s.e.})_{WIENER} \quad (16)$$

we can find the expression of M in [7]

$$M = \mu T_r(\underline{R}_{yy}) / 2 \quad (17)$$

$T_r(\underline{R}_{yy})$ denote the trace of $\underline{R}_{yy}$ and is proportional to the power of the received signal, then it is more convenient for the algorithm to have a non constant but varying μ which changes with the power of the received signal. So in order to improve the robustness of the LMS algorithm we normalize μ by an estimation of the received signal power, what we call the NLMS algorithm [9] :

$$\underline{W}(k+1) = \underline{W}(k) + 2M e(n) \underline{Y}^*(n) / (Q \hat{P}_y) \quad (18)$$

$$\hat{P}_y(k) = \beta \hat{P}_y(k-1) + (1-\beta) | \underline{Y}(n) |^2 \quad (19)$$

β is a parameter which controls the memory size of estimation (19).

The convergence time for L.M.S. algorithm given by [7] is:

$$T_k = 1 / (2 \mu \lambda_{av}) \text{ itérations} \quad (20)$$

where λ_{av} = average of the $\underline{R}_{yy}$ eigenvalues. As this algorithm takes one sample in each iteration, the convergence time is the number of samples. T_n is equal to T_k, and we can relate misadjustment and T_n by :

$$M = Q / (4 Tn) \quad (21)$$

2.2. Adaptive Differential Steepest Descent (A.D.S.D.)

In this case the m.s.e. estimation is calculated on a window of N error samples :

$$\hat{\xi}(n) = \frac{1}{N} \sum_{j=0}^{N-1} | e(n-j) |^2 \quad (22)$$

The gradient of ξ is measured by taking "symmetric differences" as shown in figure 4 :

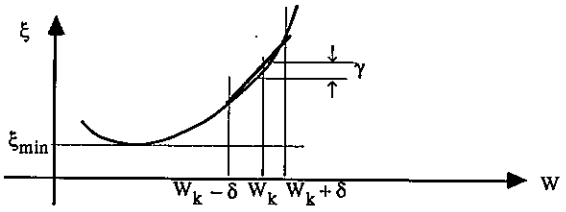


Figure 4 : Differential Steepest Descent $\nabla(\bullet)$

$$\mathrm{Re}(\hat{\nabla}_i) = (\hat{\xi}(W_i + \delta) - \hat{\xi}(W_i - \delta)) / (2\,\delta) \qquad (23\text{–}1)$$

$$\mathrm{Re}(\hat{\nabla}_i) = (\hat{\xi}(W_i + j\,\delta) - \hat{\xi}(W_i - j\,\delta)) / (2\,\delta) \qquad (24\text{–}1)$$

$i \in [1, Q]$

It should be noted that 4NQ samples are needed to complete each iteration. The result is an excess-error γ and we can define a parameter called perturbation as follows :

$$P = \gamma / \xi_{min} = \delta^2 \bullet \lambda_{av} / \xi_{min} \qquad (24)$$

Note that we are going to have the same perturbation whether the weights are near the optimum or far away. A first problem will be to adapt δ. Another extra-error results from the m.s.e. estimation. It is quantified by the misadjustment calculated in [9] :

$$M = \mu\, Q \bullet \xi_{min} / (2\, N\, \delta^2) \qquad (25)$$

and T_n is given by : $Tn = 2\, N\, Q / (\mu\, \lambda_{av})$ (26)

so : $M = Q^2 / P \bullet Tn$ (27)

The combination of (24),(25) and the minimization of the total misadjustment (M=P) leads to the following adaptive values of δ and μ : (for the i° component of $\underline{W}_k$) [9]

$$\delta_{i+1}^2 = 1/2\, [\, \hat{\xi}(W_i+\delta_i) + \hat{\xi}(W_i-\delta_i)]\; P/\hat{P}_{y1} \qquad (28)$$

$$\mu_k = 2\, N\, P^2 / \hat{P}_{y2} \qquad (29)$$

P is the perturbation desired. $\hat{P}_{y1}$ is a signal power estimation based on the 2N previous samples and $\hat{P}_{y2}$ is a signal power estimation based on the 4N previous samples (It differs generally from $\hat{P}_{y1}$). Both can be calculated using the recursive formula (19) with different values of β.This algorithm, the ADSD, has shown in experiments better robustness than DSD.

2.3. Adaptive linear random search (ALRS)

Random search algorithms seek to improve performance by changing randomly the system parameters. One classical method is the L.R.S. algorithm [7] :

$$\underline{W}(k+1) = \underline{W}(k) + \beta\, [\hat{\xi}(\underline{W}(k)) - \hat{\xi}(\underline{W}(k)+\underline{U}(k))]\; \underline{U}(k) \qquad (30)$$

where $\underline{U}(k)$ arises from a random vector generator designed to have a covariance of $\sigma^2 \underline{I}$ and a zero mean. $\hat{\xi}$ is evaluated as in (22). β and σ^2 take control of stability and convergence time. Perturbation and misadjustment are also present in this algorithm and one can find their expressions in [7] :

$$P = \sigma^2\, \mathrm{Tr}(\mathbb{R}_{yy}) / (2\, \xi_{min}) \qquad (31)$$

$$M = 2\, Q\, B\; \xi_{min} / N \qquad (32)$$

The convergence time in number of iterations is

$$Tr = 1 / (2\beta\, \sigma^2\, \lambda_{av}) \qquad (33)$$

As each iteration takes 2N samples :

$$Tn = N / (\beta\, \sigma^2\, \lambda_{av}) \qquad (34)$$

and $M = Q^2 / P\, Tn$ (35)

In the same manner as for the ADSD, we obtain the ALRS algorithm with adaptive values of σ and β :

$$\sigma^2(k) = \frac{2 \bullet P}{Q \bullet \hat{P}_y(k)}\; \hat{\xi}(k) \qquad (36)$$

$$\sigma^2(k) = \frac{N \bullet P}{2\, \hat{\xi}(k) \bullet Q} \qquad (37)$$

where $\hat{P}_y(k)$ is a signal power estimation based on the 2N previous samples computed by the recursive expression (19).

3. SOME THEORETICAL COMPARATIVE CONCLUSIONS AND SIMULATIONS

All the three algorithms described above perform in a similar way as the algorithms they are derived from but the robustness has been quite improved. The best performant one is the NLMS with a misadjustment proportionnal to the number of weights (square of that number for the other). However the NLMS algorithm is not so efficient as the ALRS or ADSD because it uses a very coarse m.s.e. estimation. Therefore it will need more iterations to reach a given s.s. error but as each iteration takes only one sample, the total number of samples is less than the equivalent one for the other algorithms. Figure 4 shows the enhanced robustness of the NLMS (4b) with respect to the LMS (4a) where the m.s.e. is plotted versus the number of iterations. At the iteration 500 the received signal power increases suddenly producing the LMS fail when the NLMS can manage the problem itself.

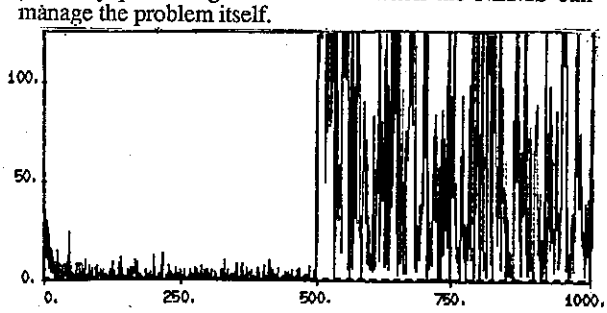


Figure 4-a : Square error versus number of iterations. LMS algorithm

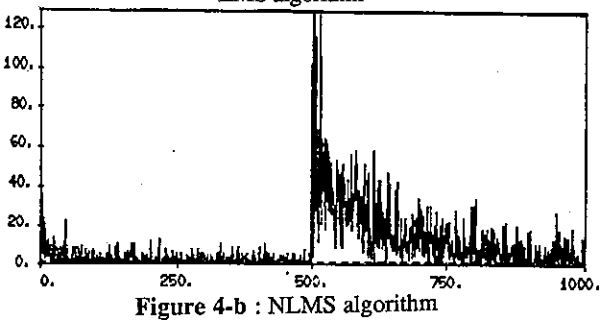


Figure 4-b : NLMS algorithm

4. APPLICATION IN REAL ENVIRONMENT

As we said before, the data under analysis here come from an underwater acoustic experiment which has been realised in the lake of CASTILLON (South-east of France). The transmitter and the receiver immersed five or ten meters deep were geometrically disposed so that only two paths could be considered (direct path + surface reflection). The transmitted signals were pseudo-random binary sequences

which modulate the amplitude of the carrier ν_0=2 kHz or 5 kHz (four periods for one elementary binary digit). The binary sequences are used both for the learning and the communication phases. The demodulation provides the two components $P_Y(t)$ and $Q_Y(t)$ which are then treated according to figure 5 in order to obtain the estimated delay $\hat{\tau}$. We used serial digital modular technology for all the treatment.

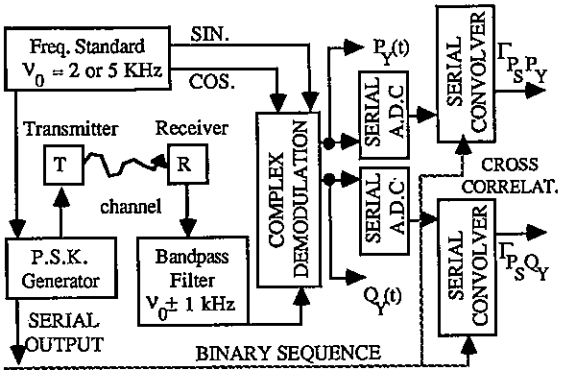


Figure 5 : Demodulation and cross correlation circuit

We briefly illustrate the good running of the adaptive filters with figure 6 which shows the two components of the received signal before and after the NLMS filter together with the transmitted sequence.

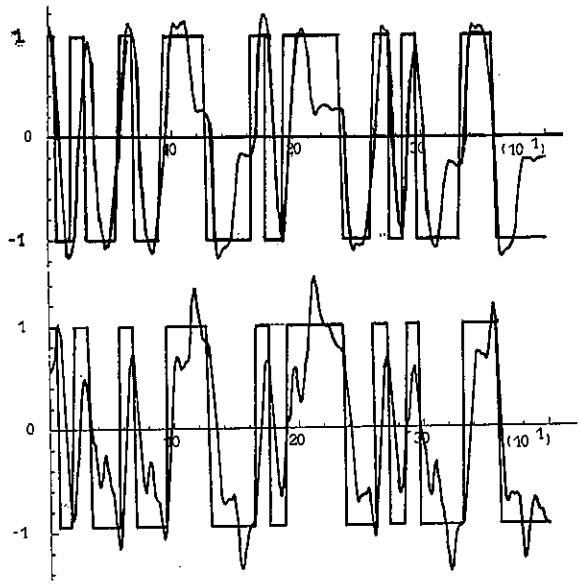
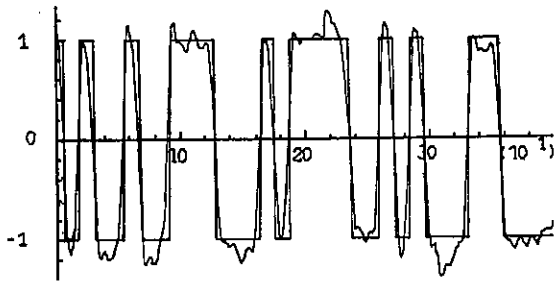


Figure 6-a : The two components of the received signal



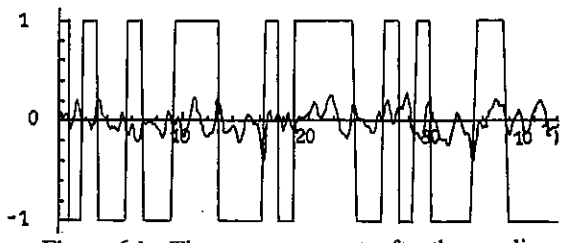


Figure 6-b : The two components after the equalizer (NLMS alg.)

The learning sequence has been stopped after twenty binary digits (twelve samples for one digit). We tested with success the two other algorithms.

CONCLUSION

The classical LMS, DSD, LRS adaptive algorithms have the drawback to work with non adaptive parameters. In many practical situations the choice of these parameters is often difficult and as they are not modified along the treatment the algorithms can fail. So we have proposed here a simple method to change these constant parameters into adaptive ones and now we dispose of three full adaptive algorithms (NLMS,ADSD, ALRS). They have been tested with success on real data coming from an underwater acoustic medium. The whole system has been described. The study goes on with the comparison of the three algorithms on more distorted data.

ACKNOWLEDGMENT :

The experiment here has been supported by the French Direction of Naval Constructions. This study was performed within the context of Franco-Spanish cooperation with the contribution of both french and spanish Foreign Offices.

BIBLIOGRAPHY :

[1] S.M. FLATTE : "Sound transmission through a fluctuating ocean" Cambridge University Press.

[2] G.L. TURING : "Introduction to Spread-Spectrum Antimultipath techniques to urban digital radio" Proc. of IEEE March 1980, pp 321-353

[3] G. TSIRITAS : "Transmission suivant deux trajets à évanouissement de Rayleigh : récepteurs optimaux" Ann. Télécom. T 36, n° 11-12 Nov. Déc. 1981, pp 585-594

[4] O. MACCHI : "Le filtrage adaptatif en télécommunications" Ann. Télécom. T 36, n° 11-12 Nov. Déc. 1981, pp 613-625

[5] J. MARTIN : "Etude de Récepteurs Optimaux après Transmission dans un Canal à Trajets Multiples. Application au cas d'un canal certain à deux trajets" Thèse DI INPG Nov. 1983.

[6] S.U.H. QURESHI : "Fast start-up Equalization with Periodic Training Sequences" IEEE Tr. Inf. Theory, Vol 5, Sept. 1977, pp 555-563

[7] B. WIDROW, J.M. MACCOUL : "A Comparison of Adaptive Algorithms based on the methods of Steepest Descent and Random Search" IEEE T.A.P. Vol 24, n° 5, Sept 1976, pp 615-638

[8] D.H. BRANDWOOD : "A Complex Gradient Operator and its Application in Adaptive Array Theory" IEEE Proc. Vol 130, Parts F-H n°1, pp.11-16

[9] R. LUCAS : "Evaluation de Algorithmas para sistemas de Arrays Adaptativos" DTTI-ETSET Barcelona, Sept. 1985

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

TIMING JITTER EFFECTS IN AN ECHO CANCELLER FOR FULL-DUPLEX DATA TRANSMISSION

S. MARCOS*, O. MACCHI* AND J.B. PINTAUX**

*Laboratoire des Signaux et Systémes, E.S.E., 91190 Gif sur Yvette, France
**Société Anonyme des Télécommunications, 41 rue Cantagrel, 75013 Paris France.

We analyse the effect of timing recovery circuit jitter of a sine form on the performance of an echo canceller that is slaved on it, in a full-duplex data transmission system. We observe how low-frequency jitter can be attenuated by the adaptation algorithm whereas high-frequency decreases the echo canceller performances for large echo to far-end signal ratio.

1. INTRODUCTION

In two-wire full duplex data transmission, the far-end received signal is disturbed by an echo due to impedance mismatches. An adaptive digital echo canceller (EC), as shown in Fig.1, will estimate the echo path impulse response and thus cancel the interfering echo signal. The EC may have to attenuate the echo entering the receiver by as much as 20 dB relative to the far-end received signal. This paper deals with an asynchronous EC. We specifically address the problem of timing jitter originating in the fact that the EC input as well as received signal are sampled in synchronism with the far-end recovered data (jittered clock) while the transmitter has local synchronous timing.

In [1], Falconer has already addressed a similar problem within the frame of digital subscriber loops. But in [1], it is the transmitter that is slaved on the central office terminal clock (recovered with jitter) whereas the EC is fed in at a local synchronous rate. Moreover the emphasis is on the analysis of jitter spectral characteristics. Here the problem is set up within the frame of modems in which the EC and the receiver would be slaved on the same clock. The problem is tackled by establishing a theoretical model which could be regarded as a time-varying identification problem, and to which we could apply some results of Falconer.

The relevant system is depicted in Fig.1. At each step, the timing recovery system of the receiver gives the sampling phase τ'_k with which, on the one hand, the EC is fed in, and on the other hand, the received signal is sampled. Due to additive noise and residual echo through the timing filters, the recovered phase τ'_k contains the constant phase that is optimal for the purpose of far-end signal recovery, plus some jitter τ_k. It is often necessary to cancel the echo on the whole baud interval T_b so that the receiver performs regardless of the time-varying sampling phase. For this purpose, the received waveform and echo replica are sampled at a multiple of the symbol rate m/T_b satisfying the Shannon theorem. The sampling instant is thus

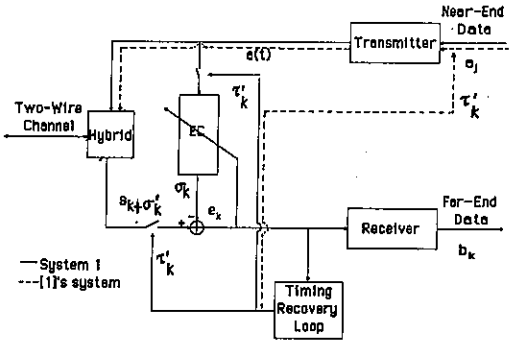


Figure 1 : Echo canceller in a full-duplex receiver.

$$t_k = kT_s + \tau_k, \qquad (1)$$

where $1/T_s$ is a Nyquist rate, and the sampled echo signal can be written

$$\sigma'(t_k) = \sum_i C(iT_s)a((k-i)T_s + \tau_k), \qquad (2)$$

C(t) being the echo analog response and a(t) the analog transmitted signal. The echo replica generated by the EC at the same instant is

$$\sigma_k = \sum_{i=-M}^{M} \hat{C}_i^k a((k-i)T_s + \tau_{k-i}), \qquad (3)$$

where $\hat{C}_i^k$ and 2M+1 are respectively the coefficients and the dimension of the EC. The residual echo $\sigma'(t_k) - \sigma_k$ thus contains not only a misadjustment due to adaptation and finite

*This work has been supported by the Teleinformatics Division of Société Anonyme des Télécommunications (Paris), grant N° 84471500.

dimension of the filter, but also a lag error due to timing jitter which will decrease the EC performances.

2.A THEORETICAL MODEL

System model

Assuming in equations (2) and (3) that τ_k is much less than T_s, one gets

$$\sigma'(t_k) = \sum_i C(iT_s)[a((k-i)T_s)+\tau_k \dot{a}((k-i)T_s)], \quad (4)$$

$$\sigma_k = \sum_i \hat{C}_i^k [a((k-i)T_s)+\tau_{k-i} \dot{a}((k-i)T_s)], \quad (5)$$

where $\dot{x}$ denote the derivative of x. Assuming the EC long enough and neglecting the second order error

$$\sum_i \tau_{k-i} \dot{a}((k-i)T_s)[C(iT_s)-\hat{C}_i^k],$$

the residual echo is

$$\sigma'(t_k) - \sigma_k = \sigma'_1(t_k) - \sigma_{1,k} \quad (6)$$

with

$$\sigma_{1,k} = \sum_i \hat{C}_i^k a((k-i)T_s), \quad (7)$$

$$\sigma'_1(t_k) = \sum_i C(iT_s)a((k-i)T_s+\tau_k-\tau_{k-i}). \quad (8)$$

It follows from Shannon theorem that

$$\sigma'_1(t_k) = \sum_i a((k-i)T_s)C(iT_s+\tau_k-\tau_i). \quad (9)$$

Equations (7), (8) represent the system in [1] (Clock with dotted line in Fig.1) : an invariant path C(t) is fed in with jittered data whereas the EC input is not jittered. Thus both configurations in Fig.1 are equivalent.

Now in the equivalent model (7), (9), the symbol rate at the input of the EC and echo path is synchronous. All the jitter effects are carried on by the echo line. Hence system 1 can be modelled by system 2 depicted in Fig.2 where vectors C_k, $\hat{C}^k$ and A_k are respectively

$$C_k{}^T=[C(-MT_s+\tau_k-\tau_{k+M}),\ldots,C(MT_s+\tau_k-\tau_{k-M})], \quad (10)$$

$$\hat{C}^{kT}=[\hat{C}_{-M}{}^k,\ldots\ldots\ldots,\hat{C}_{M}{}^k],$$

$$A_k{}^T=[a((k+M)T_s),\ldots,a((k-M)T_s)].$$

This system (7), (9) describes identification of a time-varying filter C_k by an adaptive filter C^k fed in with constant phase timing, as theoretically studied in [2].

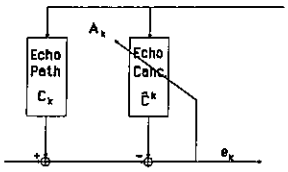


Figure 2 : System 2

Residual echo in the presence of jitter

The EC vector is updated by the classical LMS algorithm

$$\hat{C}^{k+1} = \hat{C}^k + \mu A_k^* e_k \quad (11)$$

where μ is the positive step-size, and e_k is the far-end signal after EC containing also the residual echo. Algorithm (11) is intended to recursively minimize the residual $E[|e_k|^2]$. Using the notations

$$d_k = C_{k+1} - C_k \quad ; \quad d^2 = E[|d_k|^2] \quad (12)$$

for the time increment of the variable echo path, it is shown in [2] (eq.(4.4)) under the assumptions

$\{d_k\}$ and $\{s_k\}$ are white, zero-mean and independent sequences, (H1)

$|A_k|^2 = KA = cte$, (H2)

that the average residual echo to far-end signal ratio is

$$R/S = \frac{\mu AK}{2-\mu AK} + \frac{d^2/\mu S}{2-\mu AK} = (R/S)f + (R/S)j, \quad (13)$$

where R, S, A denote the powers of residual echo, far-end signal and echo respectively (K=2M+1). The first contribution $(R/S)_f$ in (13) represents the usual EC fluctuations for a fixed echo path. It decreases as the step-size decreases. The jitter contribution $(R/S)_j$ increases as μ decreases, indicating the difficulty of the EC to track the echo path variations. Clearly the optimum step-size μ_{opt} that compromizes between fluctuations and jitter is

$$KA\mu_{opt} = 2y^2[(1+1/y^2)^{1/2} - 1] \; ; \; y^2 = KAd^2/4S. \quad (14)$$

The minimum (R/S) is 3 dB above the fluctuation component $(R/S)_f$. Thus it appears that the EC performances depend on μ and on the jitter d^2, as is natural, but also on the echo to signal ratio A/S. Moreover optimum performances do not depend separately on d^2 and A/S but on their product through the single parameter y^2 called non-stationarity degree in [2]. Also from (10) and (12) we compute

$$d^2 = \sum_i E[|\delta_k(i)|^2]|\dot{C}(iT_s)|^2, \qquad (15)$$

$$\delta_k(i) = \tau_{k+1} - \tau_{k+1-1} - \tau_k + \tau_{k-1} \cdot \qquad (15a)$$

3. SIMULATION PRESENTATION

System 1 has been simulated, with (baseband) echo path and transmitter spectral shaping having bandwidth 2800 Hz, both of the form

$$C(t) = \frac{\sin(\pi t/Tb)}{\pi t/Tb} \cdot \frac{\cos(\lambda o \pi t/Tb)}{(1-4\lambda o^2 t^2/Tb^2)} \qquad (16)$$

with $1/T_b$ = 2400 Hz, λ_o =1/6, m=3; C_k, $\hat{C}^k$, $\sigma'(t)$, σ_k, e_k and a(t) are real values. The complex symbols corresponding to a 16 QAM diagram are encoded into the real emitted signal a(t) through spectral shaping and modulation with carrier ν=1800 Hz. Then the analog signal a(t) enters both the echo path and the EC. The EC has 63 T_s-spaced real taps. To get the echo signal we have sampled the function (16) at the rate 1/Ts with a varying phase $\tau'_k = T_b/12 + \tau_k$ and performed the convolution with symbol sequence. The EC input is generated in the same way by sampling (16) at rate $1/T_s$ and at phase $\tau'_k = \tau_k$. The echo path has also been modulated. The far-end signal is obtained in the same way.

4. LOW FREQUENCY SINE JITTER

The simulated jitter is the sine wave

$$\tau_k/T_b = \propto \sin(2\pi f_o k T_s) \qquad (17)$$

with f_o= 0.5 Hz and $\propto = 3.10^{-2}$. Such a jitter is realistic [3] e.g. due to a cyclostationary signal like residual echo entering the timing recovery loop. Figure 3-a (resp. 3-b) illustrates the influence of μ (resp. A/S) on the jitter effects. Notice that the residual echo power is a squared sine-wave like $|d_k|^2$. The numerical value of d^2 is $d^2 = 2.5\ 10^{-10}$ in our case (16) and depends only on the product $\propto f_o$. By increasing μ which yields faster algorithm convergence, the jitter is better tracked. Actually this makes the R/S curves flatter by smoothing the R/S maxima. In Fig.4-a, the curves representing R/S_{max} versus μ and A/S exhibit a minimum and then join the curve of the jitter-free case (which is independent of A/S). In Fig.4-b the theoretical curves of R/S according to (13) and (15) are plotted versus A/S and μ together with the average simulated R/S. There is an excellent agreement. The existence of an optimum μ is checked which leads to a trade-off between the EC fluctuations and the jitter tracking. For any A/S realistic value, one can find a μ above which the R/S curve joins $(R/S)_f$. Notice that for very small μ-values, the algorithm performs better than predicted by theory.

As a conclusion, comparing the theoretical and the simulation curves in Fig.4, at least for realistic adaptation parameters and A/S values, we have checked that theoretical modelling of system 1 by system 2 is satisfactory, although the sine jitter has deterministic variations and not white random variations as stated in the model (H1).

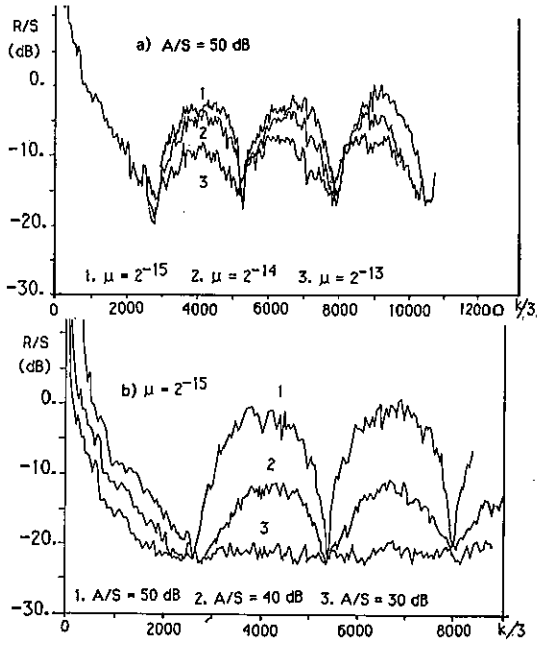


Figure 3 : Influence of a) μ ; b) A/S ,on the jitter effects.

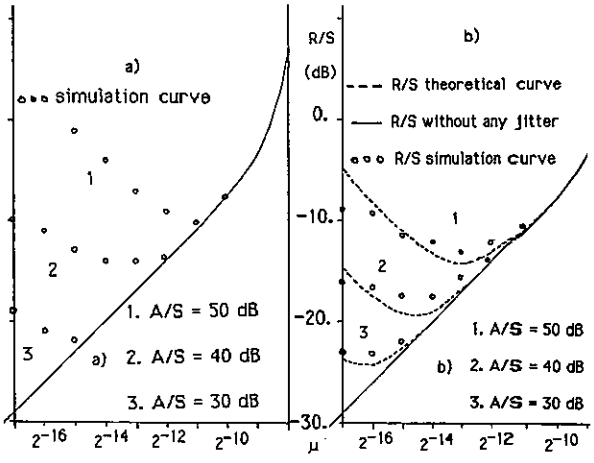


Figure 4: a) R/S_{max} ; b) R/S, versus A/S and μ for a 0.5 Hz sine jitter.

5. INFLUENCE OF THE JITTER FREQUENCY

It is well-known that the convergence speed of algorithm (11) is an increasing function of μ and that algorithm stability requires that μ is limited to the range]0,2/KA[. Thus, irrespectively of the echo to signal ratio, there is a limiting jitter frequency above which the algorithm cannot correct the jitter effects. For consistancy with the results of Section 4, we have kept the same value of $d^2=2.5\ 10^{-10}$. Consequently, the jitter amplitude parameter α will be a function of frequency f_0. In Fig.5 the theoretical curve (13) for A/S=40dB is plotted together with the simulations curves for $f_0 = 5$ Hz, $\alpha = 3.10^{-3}$ and $f_0 = 15$ Hz, $\alpha \approx 10^{-3}$. The curves a) of average simulated R/S agree with (13) although less accurately than in Section 4 and with a flatter minimum. Now the curves b) become flat : for high jitter frequency, R/S_{max} becomes insensitive to μ-variations. This is illustrated in Fig.6 which shows that increasing μ deteriorates R/S_{min} - according to the jitter free term $(R/S)_f$ - without decreasing R/S_{max}. Thus, in agreement with intuition, algorithm (11) cannot identify an echo path subject to high frequency variations, even with increased μ. Clearly the relatively poor agreement of the simulations with theory is due to inadequacy of the independent increment model (H1) for the echo path variations, when the jitter frequency increases. In other words for a deterministic sine jitter the theoretical definition of " slow variations " given in [2] through $y^2 \ll 1$ is not sufficient to ensure a good behaviour of the adaptive filter, because the jitter effects depend both on the jitter frequency and amplitude.

On the other hand, the analysis in [1] states that roughly

$$R/A = (\tau_k - \tau_{k-1})/T_b ,$$

(-44 dB in our case) and that the maximum acceptable jitter frequency is $f_0=\mu/T_b$. In the case A/S=40 dB, $\mu=2^{-15}$ it yields R/S=-4 dB and f_0=0.1Hz. But in Section 4, we have found then that R/S=-18 dB with f_0=0.5 Hz. Hence the analysis in [1] is a bit pessimistic. Actually it does not take into account the tracking capability of the algorithm and computes the lag error as if the EC was fixed at the constant (jitter free) value of the echo path. This is the presumable reason.

6. CONCLUSION

Systems employing EC to achieve full-duplex transmission can be very sensitive to timing jitter if the echo to far-end signal ratio is large. This problem is modelled as the identification of a time varying echo path. Examples for which the simulations are in a good agreement are shown. It is observed that the EC is most sensitive to the jitter frequency. Indeed, by an appropriate choice of the adaptation parameter the LMS algorithm can track low-frequency jitter. In other cases, the algorithm performances are quite deteriorated.

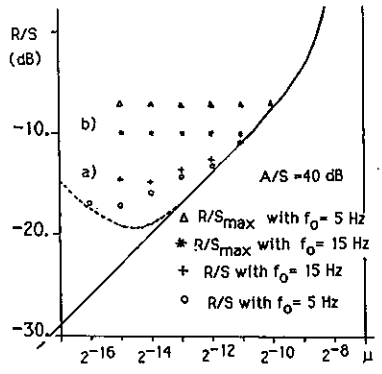


Figure 5: a) R/S ; b) R/S_{max} ,for a 5 or 15 Hz sine jitter.

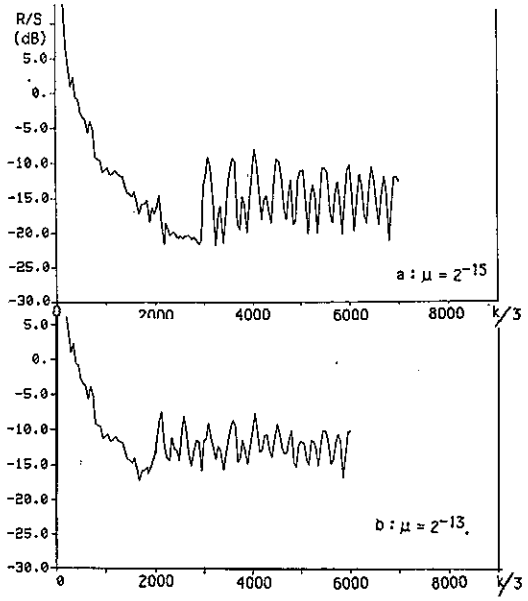


Figure 6 : Effect of a 5 Hz sine jitter on R/S
a : $\mu = 2^{-15}$, b : $\mu = 2^{-13}$.

REFERENCES :

[1] . FALCONER D.,"Timing Jitter Effects on Digital Subscriber Loop Echo Cancellers:Part1",IEEE Trans. Commun., vol.COM-33, NO.8, AUGUST 1985.

[2] . MACCHI O., "Optimization of Adaptive Identification for Time-Varying Filters",IEEE Trans. on Automat. Contr., vol. AC-31, NO.3, MARCH 1986.

[3] . FRANKS L.E., BUBROUSKI J.P.,"Statistical Properties of Timing Recovery Scheme", IEEE Trans. Commun., vol. COM-22, NO.7, JULY 1974.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ADAPTATIVE ECHO CANCELLER BASED ON DM ENCODING AND LMS FILTERING

C.C.EVCI, Z.PICEL and G.FERRIEU

T.R.T. 5, Avenue Réaumur - 92350 Le Plessis-Robinson - FRANCE

In this paper, we present an adaptive echo canceller employing both delta modulation (DM) and least-mean square (LMS) adaptive filter. This tandem connection can significantly reduce the hardware complexity. The new scheme appears to have considerable potential in telecommunications as the simulation results are comparable to those obtained from the classical echo canceller with PCM.

1. INTRODUCTION

The concept of echo cancellation was originated in Bell Laboratories as a method for eliminating the echo impairments in telephone network. Early systems were bulky and expensive for commercial purposes. Fortunately, recent developments in VLSI technology make possible to fabricate an echo canceller (EC) for limited echo path length in one chip [1, 2]. Hence, the cancellers which have not been used widely in telecommunication channels due to their complexity, once again become very important.

It is a well-known fact that echo is a disturbing factor in long-distance communications. In long-haul transmission, echo arises from impedance mismatching at the 4-wire to 2-wire bridge network, usually called hybrid. The degradation in transmission quality depends on the echo level, delay and spectral characteristics. As an example, for transmission delays round-the-loop greater than 50 ms, each subscriber can experience an echo of his own voice. Therefore, means of controlling echo is an important issue in telephone networks [3].

Another potential application of echo cancellation is in the field of electroacoustics. An echo in an acoustic environment arises due to acoustic coupling between microphones and loudspeakers in teleconference rooms. Echo cancellation can be employed in order to solve these reverberation problems. However, the design parameters for telephone EC and acoustic EC are different as the latter has much longer echo path i.e., more filter taps (~3000-5000 taps in practice) are necessary for the realization purposes [4].

An echo canceller systhesizes a replica of the echo and subtracts it from the returned signal. Classical echo cancellers with PCM use a well-known LMS algorithm in order to model the echo path. We present in this paper, a different approach from that used in a classical echo canceller with PCM, employing both DM encoding at 64 kbit/s and LMS filter with 256 taps [5].

2. SYSTEM DESCRIPTION

Figure 1 depicts a block diagram of the proposed echo canceller. The input sequence $\{f_n\}$ band limited to 3.4 kHz and sampled at 64 kHz is LDM (Linear DM) encoded. An adaptive filter operating on DM binary output sequence $\{\Delta F_n\}$ attempts to compute impulse response so that it gradually matches the actual echo path. It must be noticed that the required LPF to recover the speech at the far end extremity is assumed to be included in the echo path model.

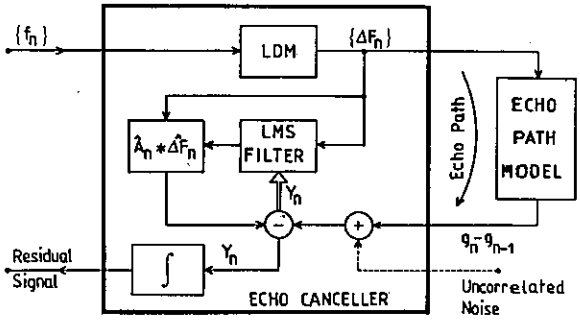


Figure 1

The advantages of DM coding in conjuction with LMS filtering in EC are two-fold. First, as the DM binary output sequence is ±1, LMS algorithm operates in a multiplication-free mode, only additions are required. Secondly, the coefficient adaptation parameter during each iteration is normalized by the same power level calculated over N (N-Tap filter) recent DM binary output samples. Consequently, these two factors can significantly reduce the hardware complexity.

The performance of echo canceller is ascertained by two parameters of importance, the speed of convergence and Echo Loss Enhancement mea-

sure (ELE in dB). In our simulations, we define ELE (dB) by equation (1), viz :

$$ELE(dB)=10.\log(\langle(g_n - g_{n-1})^2\rangle / \langle Y_n^2\rangle) \quad (1)$$

where $(g_n - g_{n-1})$ is an approximation of the derivative of echo and is obtained as follows : Let

$$g_n = e_1 . f_n + e_2 . f_{n-1} + \ldots + e_N . f_{n-(N-1)} \quad (2)$$

be the echo signal. $\hat{E} = [e_1, e_2, e_3, \ldots, e_N]$ are the coefficients of ideal echo filter.

Similarly, let g_{n-1} be

$$g_{n-1} = e_1 . f_{n-1} + e_2 . f_{n-2} + \ldots + e_N . f_{n-N} \quad (3)$$

Subtracting equation (3) from (2), we get

$$g_n - g_{n-1} = e_1 . (f_n - f_{n-1}) + \ldots + e_N . (f_{n-(N-1)} - f_{n-N}) \quad (4)$$

Also, in figure 1, the error signal to the 1-bit quantizer of LDM coder is an approximation to the derivative of the input signal under no overload condition, hence the components of equation (4) can be expressed by

$$Q[f_n - f_{n-1}] \approx \Delta F_n . \Delta \quad (5)$$

where $Q[.]$ is quantized version of $[.]$, ΔF_n is the binary output sequence and Δ is the step-size of LDM quantizer, usually optimized (Δ_{opt}) for the best performance of the encoder. The substitution of equation (5) with $\Delta = \Delta_{opt}$ in (4) yields

$$g_n - g_{n-1} \approx [e_1 . \Delta F_n + \ldots + e_N . \Delta F_{n-(N-1)}] \Delta_{opt} \quad (6)$$

This equation is important in the sense that with coefficient vector $\hat{E}$ and DM output sequence $\{\Delta F_n\}$, we get echo signal in difference domain.

LMS is the most common algorithm used in the implementation of echo cancellers. It adapts the a_{n+1} coefficient at (n+1)th instant in accordance with equation (7) so that it minimizes mean-square-error, $\langle e_n^2 \rangle$, viz :

$$a_{n+1,k} = a_{n,k} - \gamma . \delta \langle e_n^2 \rangle / \delta(a_{n,k}) \quad (7)$$

where γ controls the adaptation speed of the algorithm and k = 1,2,3, ... N. The second term in the RHS of equation (7) is called the gradient of the error. Further, equation (7) can be rewritten as

$$a_{n+1,k} = a_{n,k} + \gamma [(g_n - g_{n-1}) - \underbrace{\sum_{k=1}^{N} a_{n,k} \Delta F_{n-k}}_{Y_n}] \Delta F_{n-k} \quad (8)$$

In practical applications, there is a bound on convergence factor γ in order to avoid the divergence of the algorithm [6]. It is a function of the signal power of $\{\Delta F_n\}$ (= 1) and the order of filter, N. If we choose N=256, the value of γ amounts to 2^{-8}. Hence, equation (8) in vector form may be written as

$$\hat{A}_{n+1} = \hat{A}_n + 2^{-8} . Y_n . \Delta \hat{F}_n \quad (9)$$

In equation (9), it is clear that the elements of vector ΔF_n are ± 1, i.e., simply sign. Therefore the coefficient correction term depends only on Y_n and the sign of ΔF_n. LDM shown in figure 1 is assumed to be operating under no overload conditions. The optimum step-size of LDM is calculed from Abate's [7] empiric formula given by

$$\Delta_{opt} = \sqrt{\langle (f_n - f_{n-1})^2 \rangle} . \log_2 R \quad (10)$$

where $R = f_s/2f_c$, f_s is the sampling frequency and f_c is the signal bandwidth. In our study, instead of the sequence $\{f_n\}$, the binary output sequence $\{\Delta F_n\}$ resulting from Δ_{opt} is used.

3. SIMULATION RESULTS

3.1. Input signal

The EC performance analysis is usually done for a broadband white noise at the input as recommanded by CCITT [8]. This white noise sequence, $\{f_n\}$, low-pass filtered to 3.4 kHz, sampled 64 kHz (to compare to classical echo canceller in A or μ-law companded PCM environment) was quantized with 14 bits.

3.2. Echo path

In practice, the impulse response of the echo path, $\hat{E}$, is not known. However, in order to measure the performance of the echo canceller, we have simulated the target echo path by computing the impulse response of a second-order system with N = 256, f_s = 64 kHz and damping factor of 0.17. Figure 2 shows the resulting impulse response of the simulated echo path used as an ideal echo in our calculations.

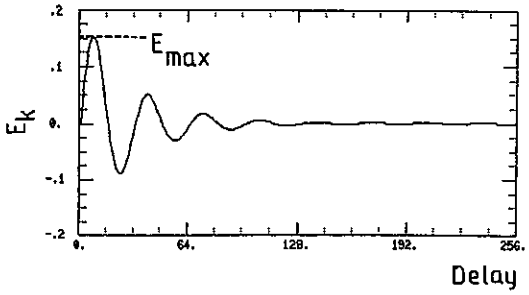


Figure 2

3.3. Performance of Echo Canceller

The sequence $\{f_n\}$ was fed into LDM as in figure 1. For the best signal-to-noise ratio (SNR) performance of the coder, step-size Δ was varied from 100.0 to 1600.0. As LDM coder maximizes SNR at one value of $\Delta = \Delta_{opt}$, it was found that the Δ_{opt} is 1000.0. The computed value of Δ_{opt} is in good agreement with equation (10). Furthermore, equation (6) was also verified by calculating the LHS and RHS of that relationship with Δ_{opt}=1000.0. This verification was necessary as the proposed scheme is based on the validity of this equation.

One way to measure the performance of the echo cancellers is to examine their convergence rates at certain time instants. The curves (a), (b),(c) in figure 3 show the ideal impulse response, $\hat{E}$, and the calculated response, $\hat{A}_n$, at the end of 12,28 and 44 ms of adaptation time, respectively. The adaptation parameter was taken as $\gamma = 2^{-8}$ which is exactly theinverse of the signal power of $\{\Delta F_n\}$ sequence over 256 samples.

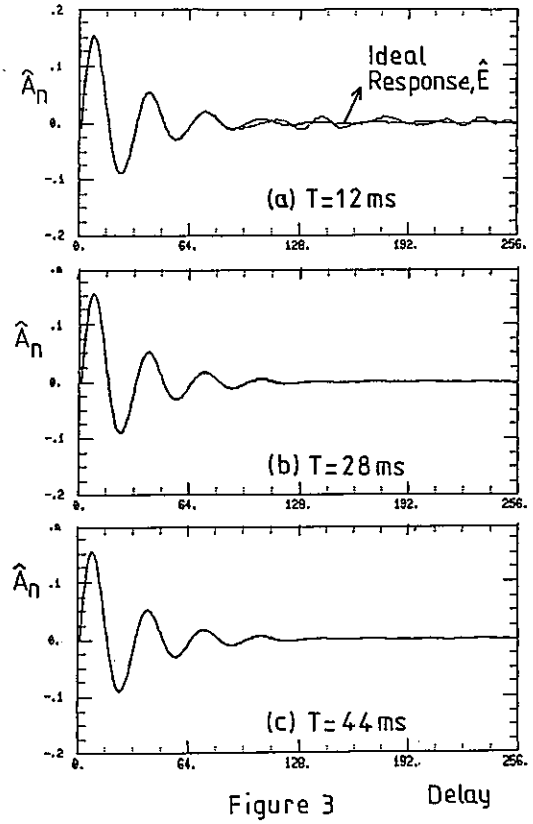


Figure 3

Next, we consider the ELE performance as defined by equation (1). In figure 4, the top curve shows the variation of ELE, for 32 bits coefficient precision. For instance, with this precision ELE of 50 dB was obtained in 80 ms. However, in practice much shorter coefficient word length must be used. It is therefore of interest to be able to determine apriori the minumum word lengths necessary to achieve specific ELE in a given time. Using equations given in section 2, it was determined, for example, that an ELE value of 30 dB can be obtained in 28 ms with 10 bits coefficient precision. Curves (a),(b),(c),(d),(e) in figure 4 correspond to 8,10,12,14 and 16 bits/coef. precision respectively, with $\gamma = 2^{-8}$.

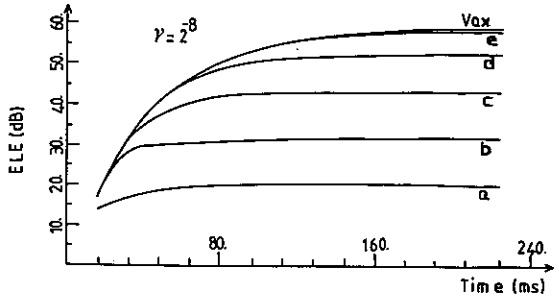


Figure 4

Another important performance measure is the convergence speed of echo cancellers in the presence of an uncorrelated noise having different power levels. For this, we use a block diagram shown in figure 1 with an additional uncorrelated noise source, see dotted lines. Figure 5 shows the results with precision of 10 bits/coef. It is clear that as the noise level is significantly below that of derivative of echo, we approach noise-free case. On the other hand, curves (d), (e) and (f) correspond to noise level of - 13 dB with $\gamma = 2^{-10}$, 2^{-9} and 2^{-8}, respectively. An inspection of curves (d) and (e) reveals that for the noise level of - B dB, we cancel the echo at least by B dB [9]. Hence, the results with the new method are consistent with those obtained from classical EC.

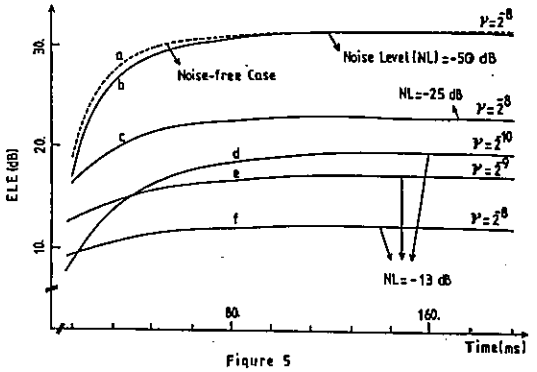


Figure 5

3.4. Improvements of the algorithm

In the previous sections we have shown some performance results of the proposed echo canceller. From figure 3 it can be noticed that the algorithm converges to its target value in

28 ms. However, the convergence speed can be improved by slight modification in the adaptation process as described next.

3.4.1. Repetition method

In order to improve the convergence speed, a repetition method [10] can be employed which performs multiple adjustments at each time instant. This technique uses M input vectors. By this repetitive use of the same input sequence, $\{\Delta F_n\}$, it is possible to get more information from each observation interval compared to the conventional LMS algorithm, and therefore to accelerate the convergence of the proposed algorithm. Table I shows the results obtained from the repetition method with M=512 input vectors.

Iteration Time (ms)	ELE (dB)	
	Proposed Met.	Repetition Met.
12	16.94	27.00
28	33.60	36.10
44	42.60	42.40

Table I

Nevertheless, this procedure requires large memory to store extra input/output vectors and faster computational resources. On the other hand, the complexity of the algorithm can be considerably reduced with a slight degradation in the performance results using subsampling method which is outlined next.

3.4.2. Subsampling Method

Instead of updating all coefficients at each instant, it is possible to adapt only ODD or only EVEN coefficients at each cycle [11]. This obviously reduces the complexity of computation by 2. At sampling rate of 64 kHz, only a slight degradation in performance is to be expected. The computer simulation results for this method are shown in Table II.

Iteration Time(ms)	ELE(dB)	
	Proposed Met.	Subsampling Met.
12	16.94	16.10
28	33.60	28.80
44	42.60	33.40

Table II

4. CONCLUSIONS

We have simulated a new method for an echo canceller employing both DM at 64 kbit/s and LMS filter and studied its performance for a broadband white noise. The effects of the coefficient wordlength were also analyzed. Similar simulations were carried out with an additional uncorrelated noise sequence. Further, two different approaches : first for speeding up the convergence rate, second for reducing the computational complexity were proposed. The simulation results show that the performance behaviour of the proposed method is comparable to the classical echo cancellers with PCM. However, due to its multiplier-free structure, the proposed method appears particularly amenable to a VLSI implementation and therefore has a good potential to be used in a wide range telecommunication equipment.

Nevertheless, it should be mentioned that LDM does not produce a satisfactory SNR at all possible input levels unless a very high sampling rate is used. For this reason, LDM should be replaced by an adaptive DM (ADM) in applications where input signals have large dynamic range. Hence, for the best performance, ADM codec is recommended [5].

ACKNOWLEDGEMENT

The authors thank J. MASSON for his advice and constructive criticism of this work.

REFERENCES

[1] Mitcher, D.W., and Elmasry, M.I.,'A CMOS Digital Echo Canceller', Proc.IEEE Conf. in Commu. Power, Montreal, Canada, 1980, pp.342-345.
[2] Duttweiller, D.L., 'A 12-Channel Digital Echo Canceller', IEEE Trans. Vol. COM-26, No.5, May 1978, pp. 647-653.
[3] Gritton, C.W.K. and Lin.D.W., 'Echo Canceller Algorithms', IEEE ASSP Magazine, 1984, pp. 30-38.
[4] Degryse, D., Druilhe, F., and Gilloire, A.,'A Multi-processor Structure for Signal Processing Application to Acoustic Echo Cancellation', Proc. of conf. in IEEE-ASSP Tampa, U.S.A., March 1985, pp. 1613-1616.
[5] French Patent, No.2569322, 1984.
[6] Bellanger, M.G. ' Digital Processing of Signals; Theory and Practice', (John Wiley, London, 1984).
[7] Abate, J.E.,'Linear-adaptive DM', Proc.of IEEE, Vol.55, No.3, March 1967, pp. 298-308.
[8] CCITT Recommendation, G.165, Red Book.
[9] Lassaux, J., Private Communication.
[10] Nagumo, J.I., and Noda, A.,'A Learning Method for System Identification', IEEE Trans. Vol.AC-12, No.3, June 1967, pp. 282-287.
[11] French Patent, No.2518853, 1981.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ON THE STABLE OPERATION OF FRACTIONALLY-SPACED ADAPTIVE EQUALIZERS IN VOICEBAND DATA MODEMS

L. Vergara-Domínguez, R. García-Gómez, F.J. Casajús-Quirós, R. Martín-Arcos.

GTTS. ETSI Telecomunicación-UPM, Cdad. Universitaria, 28040 Madrid, SPAIN
Tfno. 4495700 x 238

We deal in this paper with a classical problem found in the digital realization of fractionally-spaced adaptive equalizers: the tap wandering effect which eventually tends to oversaturate the weights and the equalizer output after a given time of operation. Some reported solutions are shown to be seriously involved with small word sizes and an alternative is proposed. We have simulated a digital V27 ter modem transmitter/receiver to show the performance of the proposed algorithm.

1. INTRODUCTION

Fractionally-spaced equalizers (FSE) [1], [2] are nowdays increasingly used in voiceband data modems due to a better performance with respect to conventional synchronous equalizers. Fractionally-spaced equalization implies sampling the received signal a number of times per symbol, reducing or even avoiding, the spectrum aliasing. This is always present in a synchronous equalizer where the signal must be sampled at the symbol rate, clearly lower than the Nyquist rate. Aliasing may eventually produce spectral nulls leading the equalizer to a very high gain and consequently increasing the noise level at the equalizer output.

In contrast with the above advantages, FSE presents some important problems, specially when a digital implementation is required [3], [4]. It has been observed that the FSE tap weights tend to continuously wander, eventually reaching magnitudes which result in partial sum or even coefficient overflows. An explanation for this phenomenon is based on the ill-conditioned situation that we have when the optimum FSE weights are searched through conventional gradient type algorithms. The large eccentricity of the equal mean-square-error (mse) contours (which simply reflects the high eigenvalue distribution and at the end the high spectral dynamic range to avoiding aliasing) allows a large number of tap weight vectors which result in similar mse values. Some of these vectors may include tap weights of an excessive magnitude for the finite precision word size used and may be reached when converging to the optimum solution. The probability of reaching those undesired vectors is increased if a systematic bias is applied to the adaptive algorithm correction term (for example if a two's complement type of quantizing is used), because the weights will wander around the steady-state mse contour, towards the overflow region.

In [3] the authors propose a simultaneous minimization of the mse and a measure (L_2 or L_1 norm) of the weight vector. The new solution is adaptively implemented by means of a systematic coefficient magnitude reduction. From another point of view the algorithm (called tap-leakage algorithm) is equivalent to adding an artificial white noise. Of course, this artificial noise produces an increase of the minimum attainable mse. We show in the next section that this can be a serious problem when a too small word size is used. Then, we have proposed in [5] an alternative algorithm simply by considering a norm constrained mse minimization, which can be adaptively realized by means of an alternative increase or reduction of the weights magnitude, trying to preserve the norm constrain. The latter can be estimated in practice at the beginning of the reception, for example after the FSE has been trained.

In this paper the norm-constrained proposed algorithm is revised and new insights are gained, specially referring to the algorithm robutness with respect to the optimum norm estimate errors.

2. THE TAP LEAKAGE ALGORITHM AND ITS LIMITATIONS

Assuming a baseband system it can be readily shown that the optimum tap weight vector (that which yields a minimum mse) is given by

$$\underline{c}_{OPT}=\underline{A}^{-1}\underline{h}=\sum_{i=0}^{L}\frac{\underline{v}_i\underline{b}}{\lambda_i}\underline{v}_i \qquad (1)$$

where $\underline{A}$ is the channel-correlation matrix, $\{\underline{v}_i\}$ and $\{\lambda_i\}$ are, respectively, the corresponding eigenvalues and eigenvectors, $\underline{h}$ is the channel vector, and L+1 the FSE length.
We can try to obtain $\underline{c}_{OPT}$ adaptively through a gradient-type algorithm, i.e.,

$$\underline{c}(n+1)=\underline{c}(n)-\alpha e(n)\underline{x}(n) \qquad (2)$$

where e(n) is the difference between the equalizer output and the decided symbol and $\underline{x}(n)$ is the signal value vector at the equalizer memory used to compute that output. Usually (2) is applied once per symbol period.
The tap leakage algorithm starts from a slightly modified optimality criterion. Now, instead of minimizing the mse solely, a simultaneous mse and weight magnitude minimization is imposed. The objective function will be

$$J=E+\mu M \qquad (3)$$

where E is the mse and M is a measure of the tap weight vector magnitude, for example $M=||\underline{c}||_2^2$ or $M=||\underline{c}||_1$ (L2 or L1 norm measures).
The FSE coefficients that minimize (3), assuming $M=||\underline{c}||_2^2$, can be readily obtained

$$\underline{c}_\mu=(\underline{A}+\mu\underline{I})^{-1}\underline{h}=\sum_{i=0}^{L}\frac{\underline{v}_i\underline{h}}{\lambda_i+\mu}\underline{v}_i \qquad (4)$$

Hence, the new optimality criterion is equivalent to adding an artificial white noise to the channel matrix.
From an adaptive point of view, we can implement (4) simply introducing a systematic reduction of the coefficient magnitude in (2)

$$\underline{c}(n+1)=\underline{c}(n)(1-\beta)-\alpha e(n)\underline{x}(n) \qquad (5)$$

That is the so-called tap leakage algorithm where β is directly related to μ, $\beta=\mu\alpha$. If a L1 norm is selected in (3) the corresponding algorithm will be

$$\underline{c}(n+1)=\underline{c}(n)-\beta\,\mathrm{sign}[\underline{c}(n)]-\alpha e(n)\underline{x}(n) \qquad (6)$$

In both cases, there exits a lower bound for a nonzero coefficient reduction and, hence, a lower bound for the artificial noise power. To clarify this point let us consider that 1 is the maximum representable magnitude and that we allow B+1 bits for finite precision representation (i.e., B bits for magnitude representation and 1 bit for the sign).
The minimum magnitude that could be represented with this word size will be 2^{-B}. If, for example, we want to guarantee through the use of (5), a systematic reduction of any coefficient greater than 2^{-D}, D<B, we need $\beta\geq 2^{-(B-D)}$, and hence, $\mu\geq 2^{-(B-D)}/\alpha$. Choosing D=4, B=12 and α=0.01 the lower bound for μ will be approximately 0.4, a large value if we take into account the maximum representable magnitude. Similarly, in (6) we need $\beta\geq 2^{-B}$, i.e., $\mu\geq 2^{-B}/\alpha$ and, with the same numerical values, we have $\mu\geq$.025, a smaller value than before, but note that with the algorithm (6) we always reduce the weights, even when their magnitudes are already small. This produces an effective increase of the artificial noise power, as will be confirmed by the examples in the final section.
It could be thought that increasing the step size α, would decrease the artificial noise power lower bound. This is theoretically true, but note that, in that case, an increase of the residual mse will accompany the step size increase, thus producing a practical bound in the maximum allowable step size.
Clearly, the above limitations will be of no importance if a large number of bits in the weight adaptation is possible.
We propose in the next section a different approach to overcome these limitations, even when a small word size is used.

3. THE PROPOSED ALGORITHM

Let us return to the objective function (3). We could have arrived to it starting from a different, but equivalent, way: minimize E, subject to the constraint, M=K. This constrain tries to preserve the tap weight magnitudes inside the non-overflow region.
Now μ plays the part of a Lagrange multiplier that should be determined by imposing the constrain. Let us concentrate on the $M=||\underline{c}||_2^2$ case. Comparing (4) with (1) we see that

$$\underline{c}_\mu=\sum_{i=0}^{L}\frac{\alpha_i}{1+(\mu/\lambda_i)}\underline{v}_i \qquad (7)$$

where α_i are the components of $\underline{c}_{OPT}$ over the space expanded by the eigenvectors $\underline{v}_i$, clearly

$$||\underline{c}_\mu||^2=\sum_{i=0}^{L}\frac{\alpha_i^2}{(1+(\mu/\lambda_i))^2} \qquad (8.a)$$

$$||\underline{c}_{OPT}||_2^2=\sum_{i=0}^{L}\alpha_i^2 \qquad (8.b)$$

Hence, imposing "a priori" a given value greater than zero for μ, leads to a norm reduction in the final solution. The amount of reduction depends on the μ/λ_i quotient.
In a practical situation there is a high probability of having a given number of eigenvalues near to zero [3], so, even with a small value of μ we could achieve an important reduction of the corresponding α_i. Note that (1) indicates that the α_i component

is inversely related to the eigenvalue λ_i; thus the "more important components" are "more reduced".
Let us assume for a moment that we could know "a priori" $||\underline{c}_{OPT}||_2^2=k_{OPT}$. And we impose this constrain in the final solution. Obviously this constrain in the final solution. Obviously this norm constrained problem will be equivalent to a direct minimization of E (without constrain) and hence μ=0.
What is more interesting is that in an adaptive implementation of the norm constrained problem, we can impose systematically a norm k_{OPT} after each adaptation is performed, thus avoiding the drift of the coefficients towards the overflow region. The gradient-type algorithm will be performed in two steps: a conventional gradient correction as in (2) and a posterior normalization

$$\underline{c}'(n+1)=\underline{c}(n)-\alpha e(n)\underline{x}(n) \qquad (9.a)$$

$$\underline{c}(n+1)=\underline{c}'(n+1)\left(\frac{K_{OPT}}{K(n+1)}\right)^{1/2} \qquad (9.b)$$

or, equivalently, in only one step

$$\underline{c}(n+1)=\underline{c}(n)\left(\frac{K_{OPT}}{K(n)}\right)^{1/2}-\alpha e(n)\underline{x}(n) \qquad (10)$$

The critical point is to know "a priori" K_{OPT}. Nevertheless, the tap weight wandering is a long term effect. Thus, we could use the algorithm (2) without control weight at the beginning (for example while the training sequence, if used, is being received) starting with $\underline{c}(0)=\underline{0}$. Once the equalizer has converged to $\underline{c}_{OPT}$ (of course with a residual mse) we can estimate K_{OPT} and then use the algorithm (10) to control the weight magnitudes.
This algorithm may be practically implemented with almost no computational increase, just observing that

$$\frac{K_{OPT}^{1/2}}{K^{1/2}(n)}=1+\frac{K_{OPT}^{1/2}-K^{1/2}(n)}{K^{1/2}(n)} \qquad (11)$$

but the second summand in the right side will be very small in practice, and hence the following algorithm is proposed

$$\underline{c}(n+1)=\underline{c}(n)[1+\beta\,\mathrm{sgn}[d]]-\alpha e(n)\underline{x}(n) \qquad (12)$$

where β is a small value and d is a measure of the difference between K_{OPT} and K(n). We could have simplified the computation of d, by taking only the most significant coefficients (for example).
Note that (12) is quite similar to (5), but now instead of performing a systematic reduction, the weight magnitude is increased or decreased depending on the measure d. In such a way the magnitudes will be mantained around a prescribed value, and the global effect will be a degradation lower than with the tap leakage algorithms. This will be experimented in the next section.

4. EXPERIMENTAL RESULTS

We have applied the above different adaptive algorithms to a 4800 bits/s (following the CCITT V27 ter norm) finite precision simulated modem. The phase and quadrature components are recuperated together with the channel equalization by means of the structure proposed in [6]. Thus, a sampling rate higher than the Nyquist frequency is necessary and then a FSE is implicitly used. The splitter-equalizer has a length of 31 tap weights and a value 0.01 was used for α. Coefficient actualization was performed once per symbol. Automatic gain control and carrier phase recovery were disconnected to isolate the equalization process. Table I summarizes the results obtained. A training sequence of 2000 symbols was used. During the training sequence, control of the coefficients was not performed (thus, all the algorithms were the same until the symbol number 2000). Once the training sequence has finished (the equalizer has clearly converged) we began to apply the different coefficient controls. In the norm constrained algorithm K_{OPT} was estimated in symbol 2000. The SNR is an average over the first 3000 symbols following the training sequence.
Clearly the tap leakage algorithms becomes involved for a 13 bit word size, while the

	no control (2)	tap leakage L2 $(5)_{\beta=2}$	tap leakage L1 $(6)_{\beta=2}$	norm constrained $(12)_{\beta=2}$	
16 bits	39	32.1	28.7	38	SNR (dB)
13 bits	28.7	15.7	15.6	27.4	

TABLE I

norm constrained algorithm practically does not degradate the no control operation. Note that the β values used in the tap leakage L2 and norm constrained algorithms guarantee a correction of any coefficient greater than or equal to 2^{-2} (a maximum representable magnitude equal to 1 is assumed). The tap leakage L1 uses a smaller value (in fact its lower bound is 2^{-2}) but it always corrects the coefficients, independently of their magnitudes, and the final SNR is even worse than with the tap leakage L2.

Next, we have tried to examine the robutness of the normalized algorithm with respect to errors in the estimate of the optimum norm. Figure 1 shows the SNR versus the percent norm error defined as $\hat{K}_{OPT}-K_{OPT}/K_{OPT}$. Thus, a negative error implies a global reduction of the coefficient norm and a positive one a corresponding norm increasing.

The curve in Figure 1 is clearly assymetrical with respect to the vertical line crossing the zero error point (a+30% error is still 3 db better than a -15% error). That means that

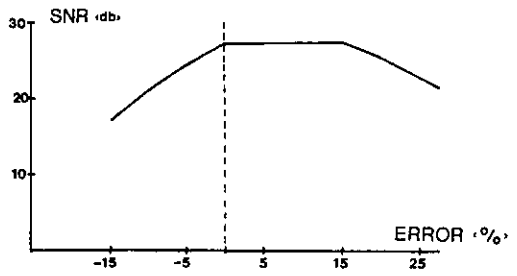


Figure 1.

the algorithm will be more sensible to an underestimated norm than to an overestimated one. Note that the tap-leakage algorithm always (indirectly) underestimates the norm (as long as μ is a positive number), and so Figure 1 is consistent with the fast degradation of the tap-leakage algorithm as the number of bits is reduced (i.e., as a higher norm reduction is achieved by the algorithm). In the other hand, the good performance of the norm constrained algorithm even for high positive percent error (until a +15%, almost no degradation is observed) is consistent with the tap-wandering effect itself, that allows uncontrolled increasing of the coefficient magnitude towards the overflow region without perceptible repercusion in the mse. Hence, imposing a norm slightly greater than that one actually estimated could guarantee a good performance in most practical cases.

5. CONCLUSIONS

Norm constrained algorithms offer an interesting alternative to tap leakage algorithms for the long term stable operation of a FSE. They may yield almost no degradation of the SNR at the equalizer output, under the assumption that a good previous estimate of the optimum weight vector norm is available. But this will be possible in practice, if we let converge the equalizer at the beginning of the reception with no control of the weight magnitude. In highly variant channels it will probably be necessary to repeat the above start-up procedure after a given interval of symbols. Nevertheless important errors in the optimum norm estimate may not necessarily lead to important SNR degradation.

Although focussed on the FSE stability problems, the above type of algorithms should be useful in other ill-conditioned situations where an adaptive estimation is desired.

REFERENCES

[1] R.D. Gitlin and S.B. Weinstein, "Fractionally-Spaced Equalization: An Improved Digital Transversal Equalizer", BSTJ, vol. 60, pp. 275-296, Feb. 1981.

[2] G. Ungerboeck, "Fractional Tap-Spacing Equalizer and Consequences for Clock Recovery in Data Modems", IEEE COM-24, pp. 856-864, Aug. 1976.

[3] R.D. Gitlin et al., "The Tap-leakage Algorithm: An Algorithm for the Stable Operation of a Digitally Implemented Fractionally Spaced Adaptive Equalizer", BSTJ, vol. 61, pp. 1817-1837, Oct. 1982.

[4] J.M. Cioffi and J.J. Werner, "Effects of Biases on Digitally Implemented Data Driven Echo Cancellers", BSTJ, vol. 64, pp. 115-139, Jan. 1985.

[5] L. Vergara-Domínguez et al., "Long Term Stabilty in Fractionally-Spaced Adaptive Equalizers". Proc. ICASSP'86, Tokyo, Japan, pp. 3007-3010, April 1986.

[6] K.H. Mueller and J.J. Werner, "A Hardware Efficient Passband Equalizer Structure for Data Transmission", IEEE COM-30, pp. 538-540, Mar. 1982.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ADAPTIVE CANCELLATION OF PHASE DISTORTED ECHOS: OPTIMIZING THE PHASE LOOP GAIN BY CONTROLLING THE POWERS

Odile MACCHI*, Kyu Ho PARK**

*Laboratoire des Signaux et Systèmes, ESE, 91190 Gif-sur-Yvette, FRANCE

**Korea Advanced Institute of Science and Technology, Seoul, KOREA.

In full duplex data transmission with classical cancellation of phase distorted echos, the loop gain for phase tracking is highly dependent on the powers P and S of the echo and far-end signal. To reduce this influence we present two different systems, either by normalizing the echo replica power at an intermediate stage (prior to phase tracking) or by normalizing the loop gain. Both methods yield an optimal gain depending only on (P/S), and are implemented with a few additions and binary shifts. Satisfactory robustness against power variations is gained in the sense that optimization of the loop for the worst possible case (maximum P/S) ensures a minimum satisfactory output SNR.

I. INTRODUCTION

We consider the problem of cancelling a phase-distorted echo

$$\sigma'_k = e^{j\Phi_k} C^T A_k \qquad (1.1)$$

that originates in the reflection and phase rotation of a random (known) data sequence a_k. In full-duplex data transmission systems such phase rotations (jitter and/or drift) are expected e.g. on the distant echo. In that case, the usual method [1] depicted in Fig. 1, is the phase adaptive echo canceller (EC). A long filter C_k adaptively identifies the true echo path response C. On the basis of the known input sequence $A_k=(a_k,\ldots,a_{k-N+1})^T$, it generates a phase free echo estimate u. A discrete phase locked loop (DPLL) then provides an estimation ϕ of the true phase distortion Φ and rotates the echo accordingly, generating the estimated echo

$$\sigma_k = u_k e^{j\phi_k} \; ; \; u_k = C_k^T A_k. \qquad (1.2)$$

Both adaptive stages are updated on the basis of the same error

$$e_k = y_k - \sigma_k = (\sigma'_k - \sigma_k) + s_k \qquad (1.3)$$

between a noisy reference $y = \sigma' + s$ of the phase distorted echo σ' and the estimated echo σ. In this problem all signals are complex-valued as in modern high rate data transmission systems. The signal s which acts as noise in (1.3) for the EC is indeed the useful far-end signal for the receiver (after EC) and includes some (proper) additive noise, with a given input SNR, taken at 20dB in the rest of the paper. Taking the example of a first order DPLL, the phase tracking algorithm is

$$\phi_{k+1} = \phi_k + \lambda \, \mathrm{Im}[e_k \sigma^*_k] \, , \qquad (1.4)$$

or one variant of it, where λ is the positive loop gain.

In this paper the dependency between the optimum gain λ, the power P of the echo σ' and the power S of the far-end signal s is first investigated (Section II). The degradation observed in the loop by not knowing P and S is exemplified through computer simulations. Two remedies are then described. In Section III, we present a system with three jointly adaptive stages where the original non rotated echo replica has controlled power. The DPLL then takes place; it is followed by a gain control. For this phase loop, the optimum gain depends only upon the ratio (P/S). It has a reduced

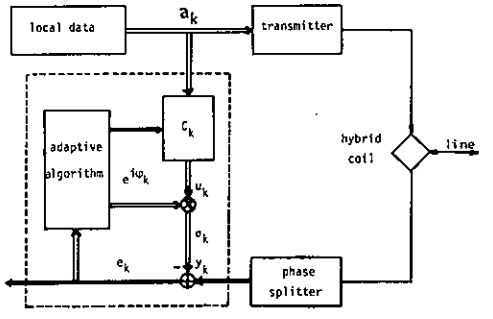


Fig. 1. Data transmitter and receiver with a phase adaptive echo canceller.

binary size for the EC taps. In Section IV, sensitivity to power variations P and S is further reduced by normalizing the gain in accordance with the increment amplitude of the phase loop (1.4). Both proposed systems involve only a slight additional computation burden.

II. PHASE LOOP SENSITIVITY TO POWER VARIATIONS

Because the true echo (1.1) has fixed linear path C and time-variable phase Φ_k, the time constant of the adaptive filter C_k can be made large enough so that $C_k=C$. Hence the DPLL (1.4) simply rotates the known (random) signal u_k in accordance with the noisy rotated

reference

$$y_k = e^{j\phi_k} u_k + s_k \qquad (2.1)$$

The theoretical evaluation of the optimum loop gain requires a statistical knowledge about the true phase variations $d_k \triangleq \phi_{k+1} - \phi_k$. In the case of phase jitter, an adequate model is a zeromean independent sequence d_k. In the case of frequency offset, a better model is the simple linear trend. In both cases, the average phase error $E(|\phi - \Phi|^2)$ is made of two additive contributions having opposite variations versus the loop gain λ [2]: the fluctuation error due to the additive noise s, is proportional to λ, while the lag error due to phase variations is inversely proportional either to λ (jitter case) or to λ^2 (frequency offset case). As a result there is always a trade-off gain that compromizes between both kinds of error and minimizes $E(|\phi - \Phi|^2)$. It is shown in [2], [3] that in the jitter case

$$P\lambda_{opt} = \sqrt{2}\, d\, (P/S)^{1/2} \qquad (2.2)$$

where d^2 is the variance of d_k. In the frequency offset case, one gets

$$P\lambda_{opt} = 2d^{2/3}\, (P/S)^{1/3},\ d \triangleq d_k. \qquad (2.3)$$

Both results are in agreement that λ_{opt} is not proportional to the inverse power $1/P$ of the signal u entering the DPLL. It depends on both P and S. Now the normalized loop gain

$$\delta_{opt} = P\lambda_{opt} = f(d, P/S)\ , \qquad (2.4)$$

depends only on the phase increment d, and on the echo to far-end signal ratio P/S, in an increasing way. For frequency drift (2.3), the sensitivity of δ_{opt} to power variations is reduced: it is in $(P/S)^{1/3}$ instead of $(P/S)^{1/2}$. Because frequency offset is a major impairment forbidding any transmission at all unless properly compensated, the rest of this paper is devoted to this latter case.

In order to illustrate the effects on the phase loop of varying the powers P and S, computer simulations have been run with a 2400 bit/s full duplex transmission system using 4-phases modulation and carrier frequency ν= 1800 Hz, with symbol interval T=1/1200s. The echo path is a cosine Nyquist filter with bandwidth [600 Hz, 3000 Hz]. The EC is implemented in a passband way so that modulation by the carrier wave is indeed included in the complex data $a = \pm 1 \pm j$. In this example, the echo frequency offset is f=1 Hz. The adaptive EC has K = 55 taps and the step-size is $\mu = 2^{-13}$. At the system output, the overall SNR ρ_o[=far-end signal/(residual echo + proper additive noise)] is plotted in Fig. 2 versus the (unnormalized) loop gain λ in three cases: (a) P=13dB, S=3dB; (b) P=3dB, S=-7dB; (c) P=S=3dB. This figure emphasizes the importandce of optimizing λ according to the variations of P and S, as predicted by the foregoing theory. It also shows that λ_{opt} is decreasing versus P and S, in agreement with (2.3).

Because the powers P and S are not known in advance, and moreover might be fluctuating, it is critical to improve the robustness of the phase loop (1.4).

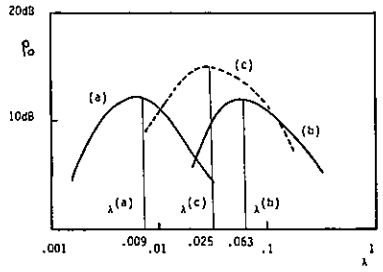


Fig. 2. Average output SNR, ρ_o of a phase adaptive EC for different powers S and P (average over 1000 iterations) versus loop gain.

III. CONTROLLING ECHO POWER PRIOR TO PHASE ADJUSTMENT

This corresponds essentially to a phase loop implementation of the type

$$\phi_{k+1} = \phi_k + \frac{\delta}{P}\, \mathrm{Im}[e_k \sigma_k^*]\ , \qquad (3.1)$$

which means that the gain λ has been normalized according to (2.4) using the echo power $P = E(|\sigma^2|)$. One way of implementing (3.1), depicted in Fig.3, is to split the system in three jointly adaptive stages using intermediate non phase rotated and rotated echo replica v and w, both with controlled unitary power, according to

$$v_k = F^T A_k\ ;\ E(|v_k|^2) = 1\ , \qquad (3.2)$$

$$w_k = e^{j\phi_k} v_k\ ;\ \sigma_k = \sqrt{P}\, w_k\ . \qquad (3.3)$$

Obviously the loop (3.1) can be written

$$\phi_{k+1} = \phi_k + \frac{\delta}{\sqrt{P}}\, \mathrm{Im}\,[e_k w_k^*]\ . \qquad (3.4)$$

Although unknown, the (normalized) echo path F and the echo amplitude $g = \sqrt{P}$ are involved in the system (3.2)-(3.4). So, in practice, F will be reached iteratively by the classical gradient algorithm of EC along

$$F_{k+1} = F_k + \mu\, e_k A_k^*\, e^{-j\phi_k}/g_k;\ \mu > 0\ . \qquad (3.5)$$

The phase loop (3.4) is approximated through

$$\phi_{k+1} = \phi_k + \delta\, \mathrm{Im}[e_k w_k^*]/g_k. \qquad (3.6)$$

In order to ensure the power, the amplitude of

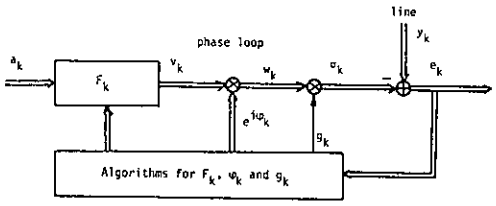


Fig.3 Phase adaptive EC with controlled echo power.

v (or w) must be tested. However, the control need not be as strict as it is in (3.2). For instance, power can simply be constrained within the range [½, 2] by implementing binary shifts for F and g according to

$$F'_{k+1} = 2F_{k+1};\ g'_{k+1} = \tfrac{1}{2}g_{k+1} \quad \text{if } E(|v_k|^2) < \tfrac{1}{2}$$
$$F'_{k+1} = \tfrac{1}{2}F_{k+1};\ g'_{k+1} = 2g_{k+1} \quad \text{if } E(|v_k|^2) \geq 2\ , \tag{3.7}$$

starting with $g_o=1$. Then g_k is of the form 2^{Pk} and the divisions in (3.5) and (3.6) occasion no arithmetical complexity. With this method, (3.1) is implemented with a gain δ adjusted up to a factor 2. Apart from the logical tests in (3.7), the additional computational burden resulting from power control is the simple binary shifts involved in the divisions or multiplication by g_k. This is negligible compared to the adaptive echo filter itself $v_k = F_k^T A_k$. Moreover it has been shown in [4] that this echo regulation reduces the bit precision of the EC coefficients by an amount of 3 to 4 bits.

It follows from (2.3) that in such a configuration, the optimum loop gain is

$$\delta_{opt} = 2\, d^{2/3}\, (P/S)^{1/3}\ . \tag{3.8}$$

It depends on the frequency offset f whose value is known to the user either because of hardware specifications or because f is in fact the residual drift of a higher-order DPLL which is able to compensate for frequency offset, up to a (known) finite accuracy.

In our computer simulations, f=1Hz which yields $d=2\pi fT=0.005$. For the same transmission system and echo path as in Section II, we have implemented a three stages echo canceller with coarse (binary) regulation of the intermediate echo replica along eq. (3.5)-(3.7). The fact that the performances depend on P and S only through their ratio, according to (3.8), is always true in the simulations. In real situations, P/S will scan a limited range. On the one hand, echos that are affected by frequency offset are produced in distant locations. Thus the far-end signal S cannot be considerably more attenuated than the echo P. On the other hand, it can be assumed that P/S>-αdB, the level α depending on the data diagram. Otherwise there is no need of an EC, the far-end signal being clearly apparent above the echo. So a plausible range is

$$-10\text{dB} \leq P/S \leq 10\text{dB} \tag{3.9}$$

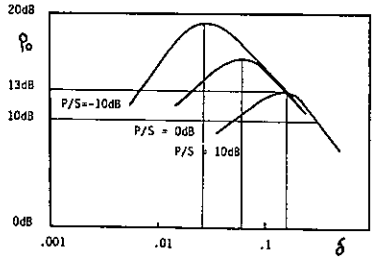


Fig.4 Output SNR for the phase adaptive EC with controlled power.

Fig. 4 gives the output SNR ρ_o versus the loop gain δ for three echo-to-signal levels P/S = -10,0,10dB. The optimum gain is given in Table 1; the agreement between analysis and computer simulations is quite satisfactory.

P/S	10dB	0dB	-10dB
δ_{opt}^{simul}	0.16	0.063	0.025
δ_{opt}^{theor}	0.13	0.060	0.028

Table 1. Optimum gain with controlled echo power.

Note that the curves of Fig.4 are superimposed for high gains. Therefore a preassigned level of ρ_o can be achieved independently of P/S by optimizing δ for the (worst) case of highest possible P/S, e.g., $\rho_o \geq 13$dB with $\delta=0.16$, independently of the echo and far-end signal powers. A desirable robustness w.r.t. the levels of P and S is thus attained by the system.

IV. NORMALIZING THE PHASE LOOP GAIN

Let us write (1.4) in the form

$$\phi_{k+1} = \phi_k + \gamma\, \mathrm{Im}[e_k \sigma_k^*]/E(|\mathrm{Im}(e_k \sigma_k^*)|)\ . \tag{4.1}$$

When the system is efficient, e and σ are independent. It can be assumed that $e\,\sigma^*$ has an even distribution in such a way that

$$E(|\mathrm{Im}(e_k \sigma_k^*)|) = \frac{2}{\pi}\, \alpha_s \alpha_p (SP)^{1/2}, \tag{4.2}$$

where the constants α_s and α_p depend on the modulation diagram on the transmission channel and echo path. In the case simulated, α_s and α_p remain very close to 1 and will be omitted in the sequel. It follows from (3.1), (3.8), (4.1) and (4.2) that the optimum gain for the new DPLL is now

$$\gamma_{opt} = \frac{4}{\pi}\, d^{2/3} (S/P)^{1/6}. \tag{4.3}$$

The comparison with (3.8) exhibits a reduced sensitivity since the exponent is divided by

two. Notice that γ_{opt} is now a decreasing function of P/S whereas δ_{opt} increases with P/S.

In order to implement the new DPLL (4.1), one must compute the expectation (4.2) and then perform the division. With the same transmission system, computer simulations were run in that way, taking the expectation over 16 samples. The resulting output SNR ρ_o is plotted in Fig. 5 in four cases: (a) (P/S)=10dB, P=0dB; (b) (P/S)=0dB, P=-10dB; (b') (P/S)=0dB, P=10dB; (c) (P/S)=-10dB, P=0dB.

The coincidence between curves (b) and (b') confirms the theoretical fact that the gain is merely a function of (P/S) and does not depend separately upon P and S. Secondly, we check that the optimum gain is an increasing function of (S/P). The quantitative agreement between theory and computer simulations is evident in Table 2. The main conclusion is that

P/S	10dB	0dB	-10dB
γ_{opt}^{simul}	0.023	0.035	0.055
γ_{opt}^{theor}	0.025	0.037	0.054

Table 2. Optimum gain with normalized phase loop

γ_{opt} is a very slow function of the signal-to-echo ratio S/P since the twenty decibels variations (3.9) of S/P induces only 3dB onto the optimum gain. We shall take advantage of this property to adjust the gain γ only approximately, inside a 3dB-range that is centred at γ_o=0.035, a value optimum for average ratio P/S=0dB.

A division-free suboptimal algorithm

In order to avoid division, the value $\lambda_o=\gamma_o E(|\ \mathrm{Im}(e_k\sigma_k^*)|\ )^{-1}$ is roughly tracked by a sequence λ_j and the phase algorithm is the classical DPLL (1.4) using the gain λ_j. For this, we compute for $|\ \mathrm{Im}(e_k\sigma_k^*)|$ two successive means M_j^1 and M_j^2 over identical consecutive intervals and we define the logical variables q_j^i= 1 if $\lambda_j M_j^i < \gamma_{o_i}$ and zero otherwise. When λ_j is close to λ_o, q_j^i takes the values 0 and 1 equally often. So we use the recursive scheme

$$\lambda_{j+1} = \begin{cases} \lambda_j\ , & \text{if } q_j^1 + q_j^2 = 1 \\ \lambda_j/2, & \text{if } q_j^1 + q_j^2 = 0 \\ 2\lambda_j\ , & \text{if } q_j^1 + q_j^2 = 2 \end{cases} \qquad (4.4)$$

which adjusts automatically the normalized gain γ of the phase loop (4.1) at the level γ_o, up to a factor that ranges between ½ and 2. The simulation results are shown in Table 3,

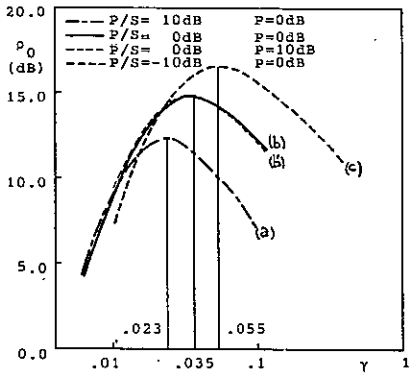


Fig.5 Output SNR for the phase adaptive EC with normalized loop gain.

P/S	10dB	0dB	-10dB
normalized loop (4.1) with γ_{opt}	12.2dB	14.8dB	16.4dB
division-free loop with γ_o=0.035	10.7dB	14.6dB	15.3dB

Table 3. Output SNR ρ_o with normalized phase loop.

compared with the truly normalized loop (4.1), using a basic averaging interval of width P=16. The division suppression is worth the slight degradation that is incurred. Notice that the choice of a smaller γ_o=0.023 (matched to the worst case P/S=10dB) ensures the minimum SNR of ρ_o=12.4dB independently of the level P/S.

References

[1] R.D. GITLIN, J.S. THOMSON, "A phase-adaptive structure for echo cancellation", IEEE Trans. Comm., Vol.COM-26, pp.1211-1220, Aug.1978.

[2] O. MACCHI, "Optimisation du gain d'une boucle à verrouillage de phase en mode de poursuite", in Proc. 10th GRETSI Colloq., Nice, pp.545-550, May 1985.

[3] E. EWEDA, O. MACCHI, "Tracking error bounds of adaptive nonstationary filtering", Automatica, Vol.21, N°3, pp.293-302, 1985.

[4] O. MACCHI, K.H. PARK, "An echo canceller with controlled power for frequency offset correction", IEEE Trans. Comm, Vol.COM-34, N°4, April 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Efficient Beamforming Based on Interpolation Over the Array Elements

Petri Jarske, Sanjit K. Mitra, fellow IEEE and Yrjö Neuvo, senior member IEEE

Abstract - A general approach to the analysis and design of subarray structures is presented. It is shown that subarray structures can be interpreted as interpolation functions for the beamforming weights and/or beamsteering delays. As a result of the interpolation a significant reduction in the number of beamforming weights and/or beamsteering delays (or phase-shifts) can be achieved in narrow beamwidth applications. Illustrative examples are given.

1. Introduction

Beamforming has become a powerful tool in e.g. radar, sonar and seismic signal processing where arrays of antennas or sensors are receiving or transmitting data coherently. The output of a conventional weighted delay-and-sum beamformer can be given by

$$b(t) = \sum_{i=0}^{N-1} w_i r_i(t - \tau_i) \tag{1}$$

where $r_i(t)$ is the i th receiver signal and w_i is the corresponding weight [1], [2]. Let the space-time signal be a plane wave

$$s(\mathbf{x}, t) = e^{j(\omega t - \mathbf{k}\cdot\mathbf{x})} \tag{2}$$

where $\mathbf{k} = (k_x, k_y, k_z)^T$ is the wavenumber vector and $\mathbf{x}$ is a position in the space. The position of the i th array element is $\mathbf{x}_i$. Assuming that the delays are set equal to

$$\tau_i = -\frac{\mathbf{k}_0 \cdot \mathbf{x}_i}{\omega}, \; i = 0, ..., N-1 \tag{3}$$

and the receivers are sampling $s(\mathbf{x}, t)$ ideally the output signal becomes

$$b(t) = e^{j\omega t} W(\mathbf{k} - \mathbf{k}_0) \tag{4}$$

The function

$$W(\mathbf{k}) = \sum_{i=0}^{N-1} w_i e^{-j\mathbf{k}\cdot\mathbf{x}_i} \tag{5}$$

is called the *array pattern.*

In the case of one dimensional array with equally spaced elements

$$\mathbf{x}_i = (iD, 0, 0)^T \tag{6}$$

and

$$\mathbf{k} = (k_x, k_y, 0)^T \tag{7}$$

where $k_x = \frac{\omega}{c}\sin\alpha$ and $k_y = \frac{\omega}{c}\cos\alpha$ are the x and y wavenumbers respectively. D is the sensor spacing.Then

$$b(t) = e^{j\omega t} W(k_x - k_{x0}) \tag{8}$$

where the array pattern

$$W(k_x) = \sum_{i=0}^{N-1} w_i e^{-jk_x iD} \tag{9}$$

Equation (9) can also be interpreted as the frequency response of a FIR filter in the k_x-domain with periodicity of $\frac{2\pi}{D}$. As a result many of the powerful synthesis methods derived for FIR digital filters can be applied to beamforming.

Beamforming and steering generally involve high data rates, large data storage requirements and heavy computations. Several methods have been derived for reducing this complexity.

Davies and Ward [3] have used multiplicative processing for achieving low-sidelobe patterns from thinned arrays. Their array configuration consists of two coincident subarrays the outputs of which are multiplied together. One of the subarrays is thinned and provides the desired aperture. A filled subarray removes the grating lobes. Because of the multiplicative configuration the output signal is demodulated. The results presented in this paper can also be used when designing multiplicative structures as expressions for the beam shape are similar.

Digital interpolation methods [4] have been applied to digital beamforming in order to reduce the high sampling rate needed when implementing digital beamsteering [5], [6]. This approach leads to reduction in the complexity of processing signals coming along each signal path from the sensors.

In this paper we take another approach, we merge by interpolation the signals from L sensors and then perform the actual signal processing tasks on this combined signal. This signal processing can be implemented using the various efficient methods developed for beamforming and steering.

The use of nonoverlapping and nonshaded subarrays as elements of an array is a special case of the interpolation concept to be analyzed in this paper. It is used in sonar and seismic applications. In this case the signals from the elements of the subarray are added together and the sum signals are taken as element signals of the main array. The way the subarrays are formed in this kind of an array corresponds to zero order interpolation over the array elements in the sense that the same signal processing branch is copied to two neighboring elements. It is well known (e.g. [1,p.306]) that in this case the overall array pattern is

$$W(\mathbf{k}, \omega) = W_1(\mathbf{k}, \omega) W_2(\mathbf{k}, \omega) \tag{10}$$

where $W_1(\mathbf{k}, \omega)$ and $W_2(\mathbf{k}, \omega)$ are the subarray and the array patterns respectively.

P. Jarske and Y. Neuvo are with the Department of Electrical Engineering, Tampere University of Technology, Tampere, Finland.

S.K. Mitra is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106.

A more general use of subarrays which can also be overlapping has been introduced by Gabel and Kurth [7]. Their approach combines time-delay and phase-shift beamsteering into a three-stage structure. In the first stage the phase-shift beamforming of each subarray is done using DFT which can be calculated efficiently. In the second stage a time-delay steering by interpolation is done for the reduced set of output signals from the first stage. The third stage combines the signals using again DFT. This way the beamformer bandwidth exceeds the limits of a conventional phase-shift beamformer while the computational complexity is reduced compared with the delay-and-sum beamforming by interpolation.

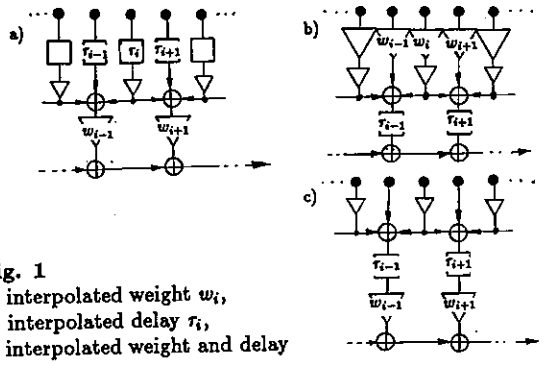


Fig. 1
a) interpolated weight w_i,
b) interpolated delay τ_i,
c) interpolated weight and delay

There are three basic alternatives to perform the interpolation as depicted in Figure 1 for the special case where one array signal is added to the two neighboring array signals. The interpolation can be made for the beamforming weights w_i, for the steering delays τ_i or for both. Naturally the last alternative is the most attractive from the implementation point of view. In the following we will derive equations describing the frequency-wavenumber properties of these alternative structures and illustrate their properties with examples.

Although the analysis of our method is done only for the weighted delay-and-sum structure it can be applied to other beamforming and steering structures as well including those mentioned above.

2. Interpolated Beamforming Weights

A new approach to implement efficient FIR digital filters called interpolated finite impulse response (IFIR) filters has been presented in [8] and [9]. These filters are usually realized as a cascade of two relatively simple filters. The first stage is called the shaping filter as it determines the passband shape. It has a thinned impulse response whose values are obtained by designing a dense model filter $H_M(\omega)$ with passband specifications correspondingly wider. The second stage interpolates the missing impulse response values and removes the grating lobes.

The interpolator can also be built into the shaping filter structure. An example of this is presented in [8,Fig.4] which is analogous to our special case of Figure 1a. The structure can be seen as a combination of three element subarrays separated by two sensor spacings. More generally an array consists of N subarrays having L elements each and center spacing MD where D is the sensor spacing. The total number of elements is then

$$K = (N-1)M + L \tag{11}$$

The output of an array consisting of subarrays can be calculated by first calculating the output of each subarray as in (8) and adding the outputs together with weights w_i. If the input of the array is a plane wave of (2) the output can be given by

$$\begin{aligned} b(t) &= \sum_{n=0}^{N-1}\sum_{i=0}^{L-1} w_{nM}a_i s\left((nM+i)D, t + \frac{k_{x0}(nM+i)D}{\omega}\right) \\ &= e^{j\omega t}\sum_{n=0}^{N-1} w_{nM}e^{-j(k_x-k_{x0})nMD}\sum_{i=0}^{L-1} a_i e^{-j(k_x-k_{x0})iD} \\ &= e^{j\omega t}W(M(k_x-k_{x0}))A(k_x-k_{x0}) \end{aligned} \tag{12}$$

The overall array pattern $W(Mk_x)A(k_x)$ is analogous to the IFIR filter frequency response

$$H(\omega) = H_M(M\omega)G(\omega) \tag{13}$$

where $H_M(M\omega) = H_S(\omega)$ is the frequency response of the thin shaping filter and $G(\omega)$ is the interpolator frequency response. Consequently the IFIR filter design procedures can be directly applied to the array pattern design. However when the interpolator is built into the array structure and reduction in the complexity of the beamformer is desired the subarray shading should be chosen as simple as possible as the interpolator is repeated N times. The design proceeds in the following way.

First choose M. In [8] and [9] there are some rules for the choice of the model filter thinning factor. Then design a simple subarray having the desired stopband attenuation at the grating lobes of $W(Mk_x)$ centered at $k_x = \frac{2n}{M}, n = 1, ..., \frac{M}{2}$. Design a model array $W(k_x)$ with beamwidth equal to MB where B is the desired beamwidth of the overall array pattern. In this case the model array can also be designed to compensate the attenuation of the subarray (see [8] and [9]). Increase the sensor spacing of the model array by factor M and replace each sensor with a subarray.

Figure 2. shows the patterns of an example design with $M = 3, L = 4, N = 14, K = 43$. The subarray shading weights are 1, 2, 2, 1, and the 14-element model array is designed to compensate the subarray attenuation on and around the main lobe.

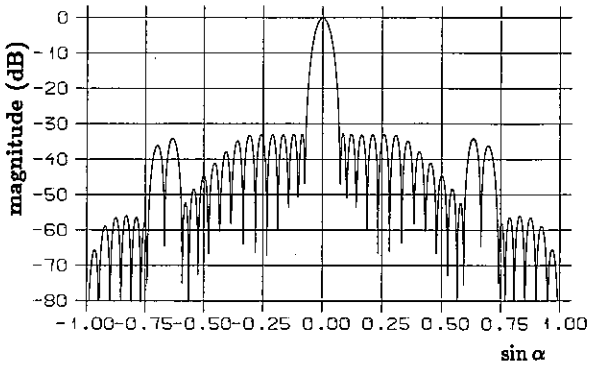


Fig. 2 an example array pattern with interpolated weights

3. Interpolated Beamsteering Delays

In the case of Figure 1b. the beam steering is done for the outputs of the subarrays while the beamforming weights have been designed for the whole array. The output of the overall array when the input is the ideal plane wave of (2) is given by

$$\begin{aligned} b(t) &= \sum_{n=0}^{N-1}\sum_{i=0}^{L-1} w_{nM+i}a_i s(x_n + x_i, t + \frac{k_{x0}x_n}{\omega} + \frac{k_{x1}x_i}{\omega}) \\ &= e^{j\omega t}\sum_{i=0}^{L-1} a_i e^{-j(k_x-k_{x1})x_i}\sum_{n=0}^{N-1} w_{nM+i}e^{-j(k_x-k_{x0})x_n} \end{aligned} \tag{14}$$

where $k_{x0} = \frac{\omega}{c}\sin\alpha_0$, $k_{x1} = \frac{\omega}{c}\sin\alpha_1$, $x_n = nMD$ and $x_i = iD$. The angles α_0 and α_1 are the steering angles of the main array and subarrays, respectively. The overall array pattern cannot be expressed as a product of two separate array patterns except in some special cases.

Let us consider the case $k_{x1} = \alpha_1 = 0$ which means that the steering delay is constant over each subarray. Figure 3. shows the pattern of a 35-element array having the structure of figure 1b. The patterns have been drawn for steering angles from 0 to 20 degrees with 4 degrees increment. The usable range of the steering angle is clearly limited because the sidelobe level increases and the mainlobe level decreases with increasing steering angle.

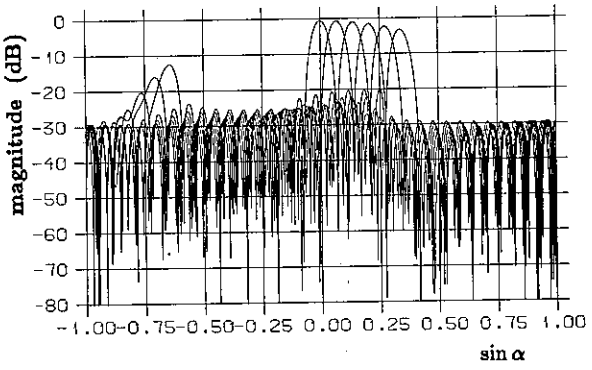


Fig. 3 an example array pattern with interpolated delays

To overcome the limitation of the steering angle a coarse steering can be applied to each subarray. Again if we want to reduce the complexity of the overall beamformer the subarray steering should be made as simply as possible. If the sensor spacing is set equal to $D = \frac{\lambda}{2}$ a relatively simple steering results if α_1 is chosen in the following way.

$$\alpha_1 = \begin{cases} -30° & \text{for } \alpha_0 < -14.5° \\ 0° & \text{for } -14.5° < \alpha_0 < 14.5° \\ 30° & \text{for } \alpha_0 > 14.5° \end{cases} \tag{15}$$

Remembering that

$$k_{x1} = \frac{\omega}{c}\sin\alpha_1 = \frac{2\pi}{\lambda}\sin\alpha_1 = \frac{\pi}{D}\sin\alpha_1 \tag{16}$$

the term $e^{-j(k_x-k_{x1})iD}$ in (14) becomes

$$e^{-j(k_x-k_{x1})iD} = e^{-jk_xiD}e^{ji\pi\sin\alpha_1} \tag{17}$$

where

$$e^{ji\pi\sin\alpha_1} = \begin{cases} (-j)^i & \text{for } \alpha_1 = -30° \\ 1 & \text{for } \alpha_1 = 0° \\ (j)^i & \text{for } \alpha_1 = 30° \end{cases} \tag{18}$$

In other words the phase-shift beam-steering of the subarrays can be done using only coefficients $\pm j$ and ± 1. When this steering is applied to the case of Figure 3 the result of Figure 4 is obtained. Naturally finer steering can also be used for the subarrays with the expence of increased complexity.

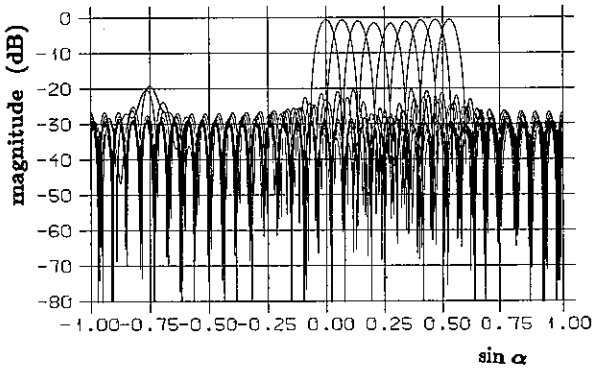


Fig. 4 the pattern of fig. 3 with a coarse steering on the subarrays

4. Interpolated Weights and Delays

Both the beamforming weights and steering delays (or phase-shifts) can be interpolated as shown in Figure 1c for a special case. The overall array pattern for a general case can be derived as in the previous cases by calculating the output of the array for a plane wave input (2).

$$\begin{aligned} b(t) &= \sum_{n=0}^{N-1}\sum_{i=0}^{L-1} w_{nM}a_i s(x_n + x_i, t + \frac{k_{x0}x_n}{\omega} + \frac{k_{x1}x_i}{\omega}) \\ &= e^{j\omega t}\sum_{n=0}^{N-1} w_{nM}e^{-j(k_x-k_{x0})x_n}\sum_{i=0}^{L-1} a_i e^{-j(k_x-k_{x1})x_i} \\ &= e^{j\omega t}W\big(M(k_x-k_{x0})\big)A(k_x-k_{x1}) \end{aligned} \tag{19}$$

where $x_n = nMD$ and $x_i = iD$.

The overall pattern is again of the form which is analogous with the IFIR filter frequency response. If $k_{x0} = k_{x1}$ which means that the main array and the subarrays are steered to exactly the same direction this case is identical with the first case where only the beamforming weights were interpolated. In fact the beamsteering is sometimes also in practice done by first steering certain parts or subarrays of the beamformer to the desired direction and then delaying the output of each subarray in order to delay each signal correctly. Generally however k_{x0} and k_{x1} need not be equal.

Let us again look first at the case $k_{x1} = 0$. As an example we use a 36-element array. The parameters describing the structure of the array are $N = 17$, $L = 4$, $M = 2$, and $K = 36$. The 4-element subarray pattern is the same as in Figure 2. The 17-element model array is designed to have the sidelobe level approximately equal to that of the subarrays. The overall array pattern is presented in Figure 5. When the main array is steered to different angles (Fig. 5) the sidelobe

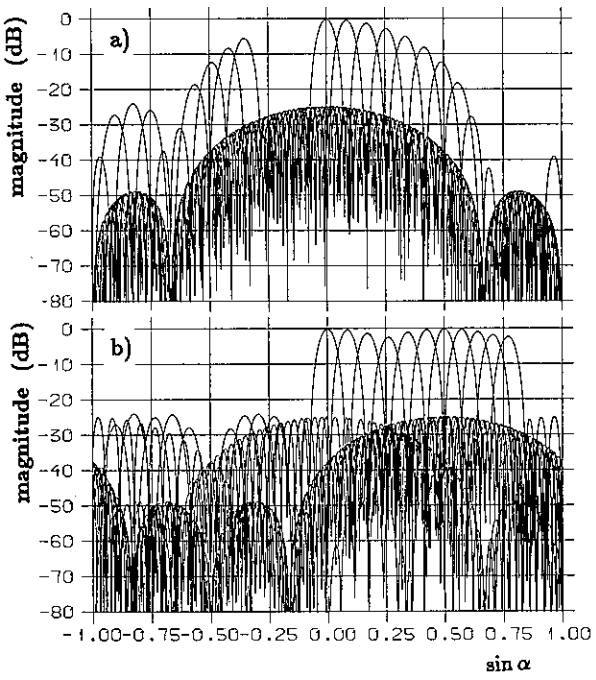


Fig. 5 a) an example array pattern with interpolated weights and delays, b) the pattern of fig. 5a) with a coarse steering on the subarrays

level of the overall array follows the pattern of the subarrays. The subarray pattern can therefore be used for controlling the sidelobe behavior of the array. In Figure 5, for example, the sidelobe level is less than -20 dB up to approximately 30° steering angle. Unfortunately the mainlobe level also follows the pattern of the subarrays resulting at a loss in the mainlobe gain. For instance, at ±30° steering angles the mainlobe gain is approximately 12 dB less than at 0°.

The subarray steering method of equation (15), as well as any kind of steering, can also be applied to this case in order to widen the steering range. The result for our 36-element example array is shown in Figure 5. In this case the difference between the mainlobe and the sidelobe level is more than 20 dB up to approximately ±50° steering angles.

5. Subarray Structures

The choice of the subarray structure and weights is usually the most difficult part of the array design. As the subarray is repeated N times the structure should be rather simple in order to avoid increased complexity of the overall array compared with the conventional direct form structure. This in turn means that the subarray pattern cannot meet very strict requirements. The subarray pattern should at least attenuate the grating lobes of the thinned model array to the desired sidelobe level while retaining the main lobe unattenuated.

In our examples the grating lobes are attenuated by designing a zero of the subarray on each grating lobe. A more general approach for subarray design is to allow them to have more zeros around the grating lobes and yet require the shading coefficients to be simple integers.

The cyclotomic polynomials [10] in general and especially the low-order polynomials are good candidates for subarray structures because their coefficients are ±1 or 0 and because their zeros are on the unit circle. However the relatively high sidelobe level of these polynomials, over -15 dB for all orders, may be disadvantageous in some applications. As the next step to improve the subarray properties a product of two or more cyclotomic polynomials can be chosen. In our examples in the previous chapters the subarray patterns were products of two low-order cyclotomic polynomials. In the case of Figure 1 the subarray pattern is

$$A(z) = 1 + 2z^{-1} + z^{-2} = (1 + z^{-1})^2 \qquad (20)$$

and in the case of Figure 2

$$A(z) = (1 + z^{-1})(1 + z^{-1} + z^{-2}) \qquad (21)$$

where $z = e^{jk_x D}$.

6. Concluding Remarks

In this paper a general framework for designing and implementing arrays consisting of possibly overlapping subarrays was given. Equations describing the frequency-wavenumber behavior of the arrays have been derived for three different types of interpolation.

By carefully designing the array and by introducing a simple coarse steering to the subarrays promising array structures can be achieved with greatly reduced number of actual beamforming and steering elements. The subarray structures introduced in the examples can be implemented with simple analog or digital hardware.

The approach can be used with any beamforming and beamsteering method.

Acknowledgements

This work was supported in part by the Academy of Finland, Hollming Ltd. Electronics and University of California Micro Grant with matching support from Intel and Rockwell.

References

[1] Dudgeon, D.E., Mersereau, R.M., *Multidimensional Digital Signal Processing.* Prentice-Hall, 1984.

[2] Dudgeon, D.E., *Fundamentals of Digital Array Processing.* Proc. IEEE, vol. 65, pp. 898-904, June 1977.

[3] Davies, D.E.N., Ward, C.R., *Low Sidelobe Patterns from Thinned Arrays Using Multiplicative Processing.* Proc. IEE, vol.127, No. 1, pp. 9-15, Feb. 1980.

[4] Crochiere, R.E., Rabiner, R.L., *Interpolation and Desimation of Digital Signals - A Tutorial Review.* Proc. IEEE, vol. 69, No. 3, pp. 300-331, Mar 1981.

[5] Mucci, R.A., *A Comparison of Efficient Beamforming Algorithms.* IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-32, pp. 548-558, June 1984.

[6] Knight, W.C., Pridham, R.G., Kay, S.M., *Digital Signal Processing for Sonar.* Proc. IEEE, vol. 69, pp. 1451-1506, Nov. 1981.

[7] Gabel, R.A., Kurth, R.R., *Hybrid Time-Delay/Phase-Shift Beamforming for Uniform Collinear Arrays.* J. Acoust. Soc. Am. 75(6), pp. 1837-1847, June 1984.

[8] Neuvo, Y., Dong, C., Mitra, S.K., *Interpolated Finite Impulse Response Filters.* IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-32, No. 3, pp. 563-570, June 1984.

[9] Saramäki, T., Neuvo, Y., Mitra, S.K., *Efficient Interpolated FIR Filters.* Proc. of the ISCAS-85, pp.1145-1148.

[10] McClellan, J.H., Rader, C.M., *Number Theory in Digital Signal Processing.*, Prentice-Hall, 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOME PROPERTIES OF FAST PROJECTION METHODS OF THE HUNG-TURNER TYPE

U. Nickel

FFM-FGAN, Electronics Department D-5307 Wachtberg 7, F.R. Germany

Eigenvector projection methods are well known for high resolution angle estimation or interference suppression for radar or sonar. We present results with the Hung-Turner projection method which overcomes the large amount of computations associated with the eigensystem computation. We show that the method performs reasonably if "averaging over the dimension" is done. An empirical rule for testing the dimension of the source space is presented. The method is only of interest for arrays of many elements.

1. Introduction.

Projection methods have proven to be very effective for interference suppression and spectral estimation with data from a time series or from an array of sensors. For the resolution enhancement application the essential of this method is described e.g. in the paper of Bienvenue/Kopp [1]. The method is known under different names like "orthogonal beamforming", "MUSIC-algorithm", and others. The application to interference suppression is mentioned in [2,3].
The idea is to separate the array output vector into a signal and a noise part. This is done by applying a projection matrix $\underline{P} = \underline{I} - \underline{X}\underline{X}^*$, where $\underline{X} = (\underline{x}_1,..\underline{x}_M)$ is a (N×M)-matrix of M orthogonal vectors describing the signal (or interference) space. N is the number of array elements, which is the dimension of the space of observations. Vectors and matrices are underlined, the asterisk means conjugate transpose. $\underline{P}$ is a projection on the complement of the space spanned by the columns of $\underline{X}$. Applying $\underline{P}$ to the measured array ouput vector $\underline{z}$ then means to suppress the interference because $\sum_i (\underline{x}_i^* \underline{z})\underline{x}_i = \underline{X}(\underline{X}^* \underline{z})$ is the estimated interference which is subtracted from the data : $\underline{P}\underline{z} = \underline{z} - \underline{X}(\underline{X}^* \underline{z})$.
It can be shown [2,3] that the projection $\underline{P}$ is the limit of the inverse covariance matrix for infinite signal-to-noise ratio, i.e. the projection matrix is a special case of the optimum interference suppression. In the case of angle estimation, subtracting the estimated signal means a filtering procedure and we take the inverse transfer function of this filter $1/(\underline{a}^* \underline{P} \underline{a})$ as an estimate of the angular spectrum. $\underline{a} = \underline{a}(\omega)$ then is the beam steering vector for a direction ω. This gives a very peaky estimate of the signal spectrum. The signal space is in general estimated using the eigenvectors corresponding to the largest eigenvalues of the sample covariance matrix. This imposes a heavy computational expense which makes this method sometimes useless, especially for large phased array radars, where weight determination has to be fast. The method of Hung and Turner [4] offers the possibility to use projection methods without calculating eigenvectors and even a covariance matrix.
As the ideas of interference suppression and high resolution angle estimation are closely related, we will use for simplicity only the terminology of interference suppression. The presented results can easily be translated to the angle estimation problem.

2. Description of the Hung-Turner method.

Hung and Turner [4] suggested a simple and fast method to estimate the interference space: they assume that the array ouput vectors contain only interference, which in fact is only true in the noise-free case, and then compute the projection matrix described above by orthogonalising these data vectors: Let $\underline{z}_1,..\underline{z}_M$ be the complex vectors of array outputs at times t= 1,..M. These are orthonormalised resulting in a matrix of orthonormal columns $(\underline{x}_1,..\underline{x}_M) = \underline{X}$. If there are M sources plus noise present, we can use the following data model : $\underline{z}(t) = \underline{A}\,\underline{b}(t) + \underline{n}(t)$, where $\underline{A} = (\underline{a}(\omega_1),..\underline{a}(\omega_M))$ and ω denotes the direction. For radar applications we can write
$a_i(\omega_k) = d_i \exp(j2\pi/\lambda(x_i u_k + y_i v_k)$

with $\omega = (u,v)$ denoting the direction sines/cosines of the sources, x_i, y_i the coordinates of the array elements, and d_i is some aperture amplitude tapering. $b_k(t)$ denotes the complex amplitude of the k-th source. If there is no noise, the vectors $\underline{s}(t) = \underline{A}\,\underline{b}(t)$, t=1,..M, span the same space as $\underline{a}(\omega_1),..\underline{a}(\omega_M)$, provided the vectors of complex amplitudes $\underline{b}(t_1),...\underline{b}(t_M)$ are linearly independent. For sufficiently fluctuating random amplitudes this is almost surely true. However, for completely correlated sources (e.g. due to multipath) the complex amplitudes are linearly dependent. The method is attractive for numbers of sources M much smaller than the number of sensors, e.g. for large phased array antennas. Arrays consisting of hundreds of elements prohibit the use of the covariance matrix. In practice, when there is noise present, the Hung-Turner method is only a very rough estimate of the signal space resulting in a randomly fluctuating interference cancellation.

3. Computation of the Hung-Turner projection.

The weight for interference suppression is $\underline{w} = \underline{P}\underline{a}_0$, where $\underline{a}_0 = \underline{a}(\omega_0)$ is the steering vector in a given look direction ω_0, [2]. If we orthogonalise the data vectors by the Gram-Schmid procedure, we can update the weight vector, because
$\underline{w} = \underline{a} - \sum_i \underline{x}_i \underline{x}_i^* \underline{a}$, so using $\underline{w}_0 = \underline{a}$, we have

$$\underline{w}_i = \underline{w}_{i-1} - (\underline{x}_i^* \underline{a})\underline{x}_i \quad , \quad (i=1,..M)$$

Another possibility is to compute the projection explicitly, using the identity $\underline{X}\underline{X}^* = \underline{Z}(\underline{Z}^*\underline{Z})^{-1}\underline{Z}^*$, if $\underline{Z} = (\underline{z}_1,..\underline{z}_M)$.

The computation of $(\underline{Z}^*\underline{Z})^{-1}$ can be done via Cholesky decomposition.
The most elegant way to compute a projection is to use the QR-decomposition, especially in the case of angle estimation, where we have to compute quadratic forms $r = \underline{a}^* \underline{P}\underline{a}$. Recognising that r is just the residual of a least-squares problem, because

$$r = \underline{a}^* \underline{P}\underline{a} = \| \underline{P}\underline{a} \|^2$$

$$= \| \underline{a} - \underline{Z}(\underline{Z}^*\underline{Z})^{-1}\underline{Z}^* \underline{a} \|^2$$
$$= \min_{\underline{c}} \| \underline{a} - \underline{Z}\underline{c} \|^2 .$$

We can apply the QR-decomposition to the matrix $\underline{Z}$ to compute the residual r or, if necessary, the residual vector $\underline{P}\underline{a}$. This is a standard technique of numerical analysis. The point is that there exist very fast algorithms for QR-decomposition for systolic array processors. These are the Kung-Gentleman-McWhirter algorithms [5], which are of special importance to this method.

4. Simulation results.

The behaviour of the Hung-Turner method was analysed by simulations. A good measure of the interference suppression is the signal-to-noise ratio

$$SNR = E\{ |\underline{w}^* \underline{s}|^2 \} \; / \; E\{ |\underline{w}^* \underline{n}|^2 \}$$

$$= \underline{w}^* \underline{s}\underline{s}^* \underline{w} \; / \; \underline{w}^* \underline{Q}\, \underline{w}$$

for given $\underline{w}$, where $\underline{Q}$ is the interference-plus-noise covariance matrix and $\underline{s}$ a given signal in the look direction of the antenna. For the Hung-Turner method $\underline{w} = \underline{P}\underline{a}(\omega)$ is a random vector and we plot ensembles of SNR-curves to get an impression of the distribution. The evaluation of analytical expressions of SNR for a random projection $\underline{P}$ proves to be rather difficult. The current statistical literature does not treat the distribution of such projections.
Figure 1 shows SNR plots for a linear 64-element array at $\lambda/2$-spacing. Two sources are present with a single element single source-to-noise ratio of 20dB located at u=±0.04 (azimuth angle ±2.3 degrees). The beamwidth is 1.8 degrees. To overload not the plot, only a sector of u= -0.8...0.2 is shown.

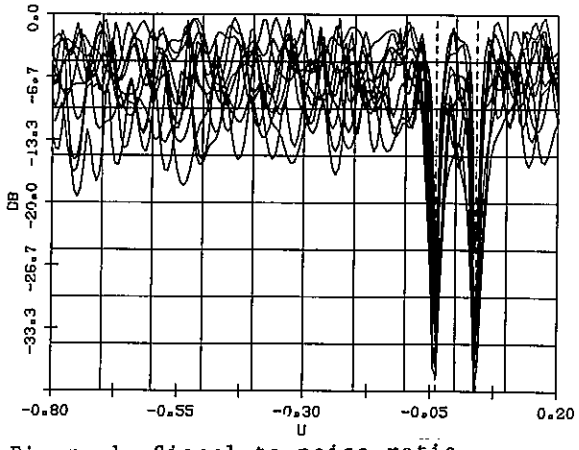


Figure 1. Signal-to-noise ratio
Two sources at -0.04, 0.04, K=2.

Two sample vectors were taken for the projection, because the interference space is of dimension 2. One can see that this is not satisfying. We have on the average a loss of 10dB compared to the optimum SNR (=0 dB, the grid indicates 5dB steps). The eigenvector projection would give the same distribution, because the sample covariance matrix has rank 2 in this case. Hung

and Turner concluded that their method is only appropriate for high interference-to-noise ratio, when the sample vectors approximate better the source space. But for higher interference-to-noise ratios the plots look very similar to fig.1. This is because strong interferences require also much higher suppression, although the interference space itself is described more accurately by the projection.
The problem is to achieve some averaging for the interference space. Hung and Turner therefore also suggested to use M columns of the sample covariance matrix for orthogonalisation instead of the sample vectors. This is of computational disadvantage as we want to avoid the computation of the covariance matrix. If the observation space is sufficiently large, N>>M, then we can simply take more than M data vectors for the projection. The effect of doing this is shown in figure 2. We have the same antenna and interference situation as in figure 1, but the projection is calculated from 6 data vectors. One can see that on the average we have a small loss to the optimum SNR of only 2-3dB.

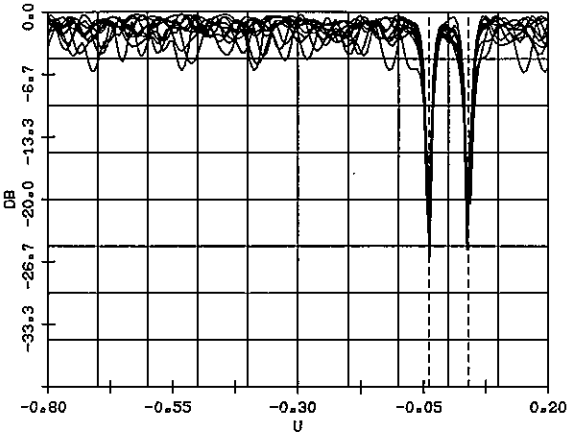


Figure 2. Signal-to-noise ratio
Two sources at -0.04, 0.04, K=6.
For stronger sources the plots look the same. This means that it is possible to average over the dimension, if the dimension of the whole space is large enough. The problem is how many samples we should take. If we take the interference space too large, then the plots look similar to fig.2, but the average level of SNR goes uniformly down. Our simulations showed that as a rule of thumb this happens for a number of samples K>N/2. On the other hand sufficient averaging was achieved only for K>2M. As we are interested in a minimum number of computations we are thus faced with the test problem of determining the optimum number of data vectors K.

5. The test problem.

It is difficult to define this test problem in terms of standard test theory. We have therefore studied the problem empirically by defining heuristically motivated test statistics which have certain invariance properties. Hung and Turner suggested for testing the dimension of the interference space the length of the residual vector after projection

$$T_{1,K} = \|\underline{P}\underline{z}_K\|^2 = \underline{z}_K^* \underline{P} \underline{z}_K$$

If T_1 is small this means that $\underline{z}_K$ is approximately in the subspace spanned by $(\underline{z}_1, .. \underline{z}_{K-1}) = \underline{Z}$, where

$$\underline{P} = \underline{I} - \underline{Z}(\underline{Z}^*\underline{Z})^{-1}\underline{Z}^*.$$

Other useful test statistics are

$$T_{2,K} = T_{1,K}/\|\underline{z}_K\|^2 = \|\underline{P}(\underline{z}_K/\|\underline{z}_K\|)\|^2$$

which is the normalised residual length or the cosine of the angle between the unit vectors of $\underline{z}_K$ and $\underline{P}\underline{z}_K$.

$$T_{3,K} = \det(\underline{Z}^*\underline{Z})/(1/K\ \mathrm{tr}(\underline{Z}^*\underline{Z}))^{1/K}$$

is a measure of the estimated volume of the parallelogram spanned by the columns of $\underline{Z} = (\underline{z}_1, .. \underline{z}_K)$, (the denominator is the Gram-determinant), normalised by the average length of the vectors $\|\underline{z}\|^2 = 1/K\ \mathrm{tr}(\underline{Z}^*\underline{Z})$. This statistic resembles very much the statistic for testing equality of the eigenvalues of a sample covariance matrix (sphericity test).

$$T_{4,K} = \det(\underline{Z}^*\underline{Z}) \Big/ \prod_{k=1}^{K} \|\underline{z}_k\|^2$$

is also a measure of the volume of the parallelogram spanned by $\underline{Z}$, but we have used only unit vectors, because $T_4 = \det(\tilde{\underline{Z}}^*\tilde{\underline{Z}})$ with $\tilde{\underline{Z}} = (\underline{z}_1/\|\underline{z}_1\|, .. \underline{z}_K/\|\underline{z}_K\|)$.

So we have $T_4 \leq 1$.
Using some rules for determinants and matrix inversion, we get the following relations:

$$T_{4,K} = \prod_{i=1}^{K} T_{2,i} \quad \text{or} \qquad (1)$$

$$T_{2,K} = T_{4,K}/T_{4,K-1} \qquad (2)$$

If we denote the Gramian matrix $\underline{Z}^*\underline{Z}$ by $\underline{G}$, we have

$$T_{1,K} = \det(\underline{G}_K)/\det(\underline{G}_{K-1}) \qquad (3)$$

i.e. T_1 measures also the change of the volume of $\underline{Z}$.

$$T_{1,K} = \left((\underline{G}_K^{-1})_{K,K} \right)^{-1} \qquad (4)$$

where $C_{k,k}$ means the (k,k)-th element of a matrix $\underline{C}$.
A good test statistic should be invariant against transformations possibly induced by the measurement system. We can require that the statistic is in-

variant against unitary transformations:

$\underline{z} \rightarrow \underline{U}\underline{z}$, where $\underline{U}^*\underline{U} = \underline{U}\,\underline{U}^* = \underline{I}$.

This implies that the statistic is insensitive to the chosen coordinate system. It is easy to show that the four statistics are all unitary invariant. Another desirable property is scale invariance:

$\underline{z}_k \rightarrow d_k \underline{z}_k$, for any sequence d_k.

If d is complex and of magnitude 1 this is a special case of unitary invariance and means independence of the initial phase. For d real this means independence of variations of the received power. One finds out that only T_2 and T_4 are scale invariant. We then investigated the distribution of T_2, T_4 by simulations. It turned out that T_4 always showed a more concentrated density than T_2. We therefore considered T_4 as the best of these four statistics.

Figure 3 shows SNR patterns for the same source distribution as in fig.1, but the number of sample vectors for each trial was determined using $(T_4)^{1/2K}$, which denotes the length of the edge of a cube of volume equal to T_4. Relation (1) shows that this statistic is just the geometric mean of T_2. Increasing the interference space was stopped if $(T_{4.K})^{1/2K} < 0.2$. The decrease of this test statistic is also shown in fig.3. The number K varies between 6 and 9, and is 7.4 on the average.

This test problem clearly needs further investigation, but the proposed method is able to produce reasonable results.

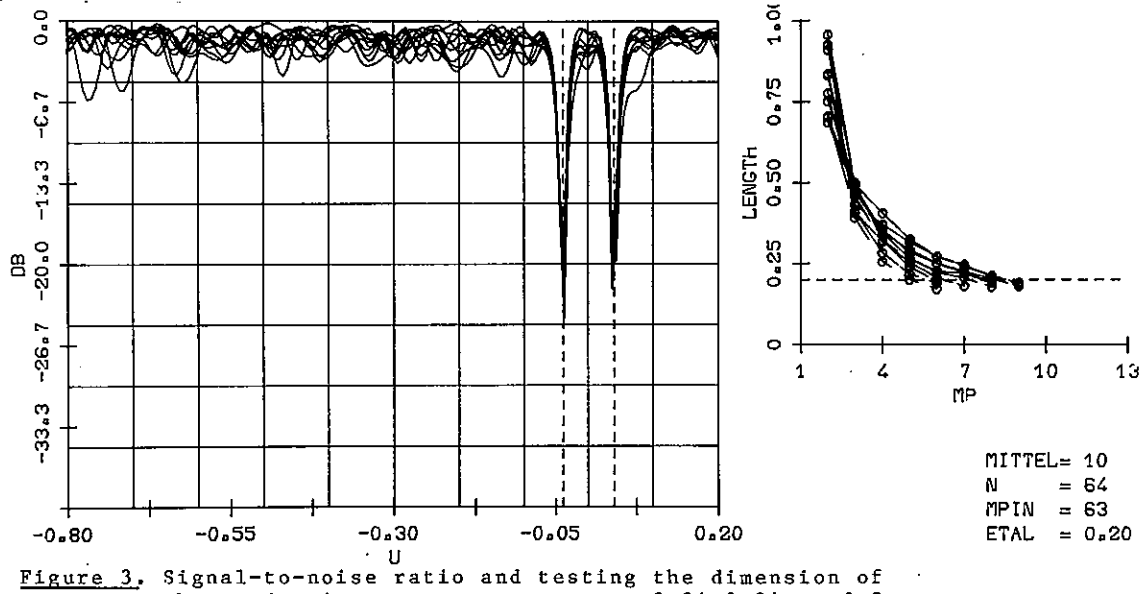


Figure 3. Signal-to-noise ratio and testing the dimension of the projection space, sources at -0.04,0.04, η=0.2.

6. Conclusions.

We have shown that the Hung-Turner projection method gives reasonable results independent of the interference-to-noise ratio if averaging over the dimension is done. This works only if the number of array elements is much larger than the number of sources. An empirical test for the projection subspace has been given, which performed satisfactory in simulations. The advantage of the Hung-Turner projection is the easy computation, which makes the method much faster than eigenvector projection.

References.

[1] Bienvenu,G.,Kopp,L.: Optimality of high resolution array processing using the eigensystem appraoch. IEEE Trans.-ASSP-31,No.5,Oct.1983.

[2] Hudson,J.E.: Adaptive array principles. IEE series, Peter Peregrinus, U.K.,1981.

[3] Keating,P.N.: A rapid approximation to optimal array processing for the case of strong localized interferences. J.Acoust.Soc.Am. 65(2), Febr.1979.

[4] Hung,E.K.L., Turner,R.M.: A fast beamforming algorithm for large arrays. IEEE Trans.AES-19,No.4, July 1983.

[5] Ward,C.R.,Robson,A.J.,Hargrave,P.J. McWhirter, J.G.: Application of a systolic array to adaptive beamforming. IEE Proc.,Vol.131,Pt.F,No.6,Oct.1984.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SIGNAL PROCESSING IN AN FMCW RADAR FOR DETECTING VOIDS AND HIDDEN OBJECTS IN BUILDING MATERIALS

Dr L.G. Cuthbert, Prof A.D. Olver, T-F. Liau and Y. Liu*

Department of Electrical and Electronic Engineering,
Queen Mary College (University of London)
Mile End Road, London E1 4NS, UK

This paper describes the developments in signal processing that have enabled research workers at Queen Mary College (QMC) to successfully produce a radar system for detecting non-metallic hidden objects, or voids in building materials. Techniques that are described in the paper include the use of template matching, maximum entropy methods for spectral analysis and the formulation of the return as an image.

1. INTRODUCTION

1.1. Background

Research into the use of microwave radar for the detection of non-metallic hidden objects has been taking place at Queen Mary College (QMC) for a number of years. Applications for a radar system of this kind range from the detection of underground services (such as gas pipes) to the investigation of construction materials. The latter are of particular interest because many common materials used in building have a low attenuation at frequencies suitable for radar. This paper describes the signal processing methods developed at QMC to enable such radars to be of practical use.

1.2. FMCW Radar System

The theory of FMCW radar is well known [1] but it is useful to include a summary here.

Figure 1 shows the basic arrangement for the FMCW radar. Repeated microwave chirp signals are transmitted, each chirp sweeping linearly between two preset frequencies. For simplicity it can be assumed here that the envelope of the chirp has a constant amplitude, but in practice a shaped envelope is used to reduce sidelobe levels in the frequency domain. Signals reflected from the target are received using the same antenna and mixed with the transmitted signal to form a difference frequency.

In the case of multiple targets the resulting mixed signal consists, in principle, of one frequency component for each target, with the frequency of each component being directly proportional to the range of the corresponding target. With a short-range radar of this type

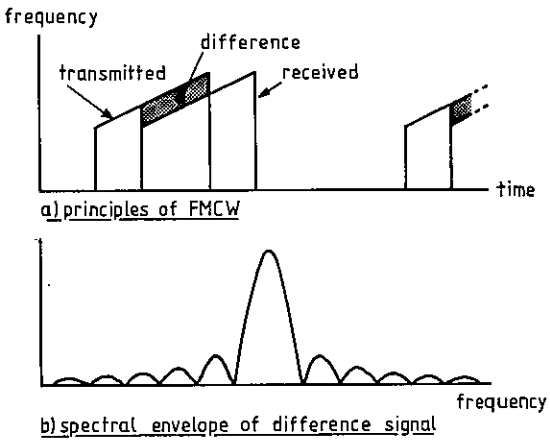


Figure 1 FMCW principles

there is an added complication in that there will also be multiple reflections between targets causing spurious target returns.

An advantage of this type of radar is that the sweep rate and the period can be arranged to produce difference frequencies in the audio range: the mixed signal can be sampled and converted into a digital signal for subsequent processing without undue difficulty.

For analysis, the mixed signal is converted to the frequency domain. In the ideal case each target would produce a single frequency component but the finite duration of the sweep and the constant amplitude envelope lead to a sinc(x) form for the envelope in the frequency domain.

*On study leave from Jianghan Petroleum College, Hubei 434102, Peoples Republic of China

1.3. Operating frequency

A critical choice in the design of the overall system is the choice of operating frequencies: as the frequency is increased the attenuation of the transmission medium generally increases but the resolution improves. Also, as the operating frequency is increased the antenna becomes smaller and lighter and the frequency band swept becomes proportionally smaller.

Experiments at QMC have concentrated on three different frequency bands: 1-2 GHz, 2-4 GHz and 9-11 GHz. Each of these bands has been found to be suitable for a particular type of use. For investigating buildings the 9-11 GHz radar shows a great deal of promise whereas the lower-frequency versions have advantages where a greater penetration of lossy materials is required and where the target is big enough to be detected by the longer wavelengths.

2. SIGNAL PROCESSING: FFT METHODS

2.1. Principles

The mixed signal from the radar is a short burst containing frequency components corresponding to the targets. In order to analyse the signal it is first converted to a frequency spectrum: in the earliest work at QMC this process was carried out using a commercial spectrum-analyser, but all the recent work has used a Fast Fourier Transform (FFT) algorithm on a computer. Lately the FFT has been replaced by maximum entropy methods which are more suited to the relatively short bursts of signal obtained: these methods are discussed later in this paper.

A template-matching technique has been used to to pick out targets automatically from the frequency spectrum of the return [2]; it has also proved possible to distinguish between different types of target [3].

One of the difficulties found when processing the return is that there is a considerable amount of system generated clutter. The effect of this can be reduced by subtracting a stored replica of the clutter from the actual signal. This subtraction is generally performed in the time domain and can be extended to subtract out the unwanted return from the front surface of the area under investigation. However, care must be taken if this return is subtracted out as irregularities in the surface may cause the return to vary. When scanning across a surface that is fairly uniform, and where targets are expected to occur at intervals, it is the difference between sweeps that is really of interest. In this case the previous return can be subtracted from the current return to highlight differences.

Limitations inherent in the FFT also cause problems because the duration of the burst of mixed signal is short in terms of the number of cycles of signal present: hence, the resulting spectrum has a broad main lobe and a high sidelobe level. Such a return makes it difficult to distinguish targets that are close together.

One way to reduce the sidelobe level is to shape the envelope of the time signal using Hamming weighting, or a similar scheme. In the QMC radar, weighting is used and has been found to be effective. However, the situation with a radar used for detecting hidden objects is more complicated than the ideal case described above: the medium through which the signal travels will have an attenuation that varies with frequency and will, moreover, also have a non-linear phase characteristic. Experiments have been carried out at QMC to determine the optimum weighting shape to use in a practical situation and it was found that the an optimal window gave minimal advantages over a simple raised-cosine.

2.1. Template matching

This technique is used for automatic detection of targets from the frequency spectrum of the return: the process is illustrated in figure 2.

The frequency spectrum is filtered using a low-pass FIR digital filter algorithm to smooth the shape of the spectrum and to make it more amenable to computer matching. An optimisation routine is then used to fit a model comprising several templates to the filtered spectrum. The optimisation adjusts the positions and amplitudes of the templates in the model to get the best fit and the position of each template is then taken to be the position of a target.

Figure 2a shows the principle of the method: it should be noted that it is necessary to include a penalty function to ensure that redundant templates are forced to zero, rather than being moved beyond the edge of the range.

It is of course necessary to ensure that a representative template is used in the model. This is obtained by using the return from a standard object, such as a metal plate; an example is shown in figure 2b.

Results from using this technique are shown in figure 2c. Here the dotted line is the actual return from the radar and the solid line the result of optimisation. Here the technique has correctly identified the two targets that could not be seen by eye from the return.

This approach gives good results but it is slow as the optimisation of the model to the actual return requires a great deal of computation. A further problem is that the returns from different types of target vary considerably, although this property can be used to distinguish between different types of target.

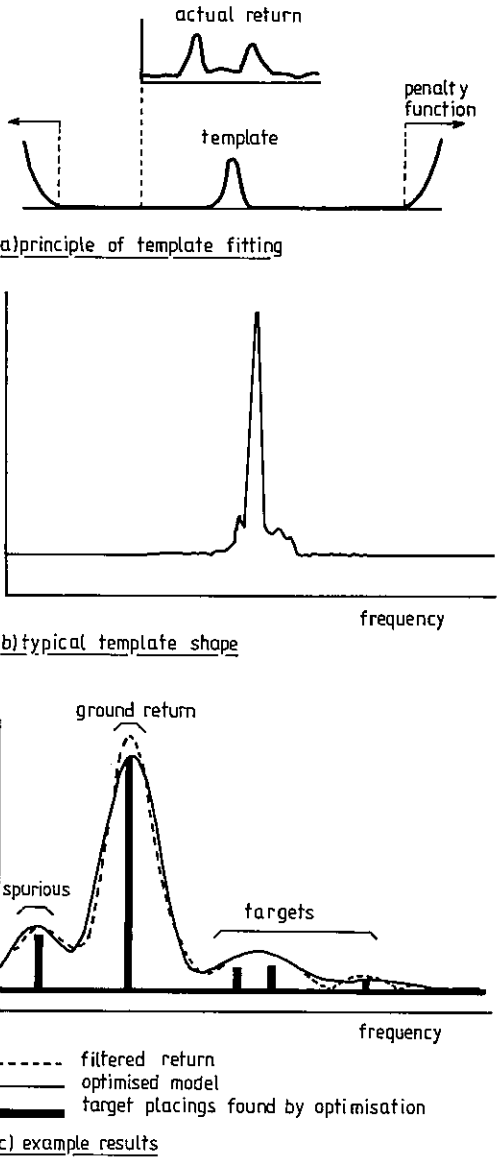


Figure 2 Template matching

2.2. Distinguishing between types of target

The method described in the previous section can be extended by building a model containing differently shaped templates corresponding to a range of different types of target. Using this approach it has been shown by the QMC group [3] that computer-recognition of a limited range of targets is possible.

Figure 3 compares the returns obtained from two different targets; from the difference in shape of the return it is obvious that an automatic recognition technique is possible.

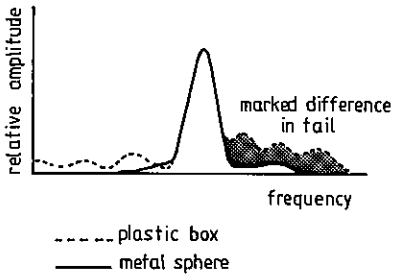


Figure 3 Variation of return with target shape

The return shown in figure 3 from the plastic box is typical of that from hollow (in radar terms) objects: the multiple lobes are caused by the reflection from the back of the box and by the multiple reflections within the box. This property is also seen in the image described in a later section.

In practice, the wide variations in shape of the returns makes this technique impracticable, except where the range of possible targets is very limited.

3. NON-FOURIER METHODS

These methods, originally developed for speech processing, are more suitable for short bursts of signal than is the Fourier Transform. They produce sharper returns without high sidelobe levels.

The basis of these methods [4] is the formation of a digital filter with transfer function:

$$H(z) = [1 + \sum_{n=1}^{p} a_n z^{-n}]^{-1}$$

The coefficients, a_n, are selected to produce an impulse response that is the same as the time signal being analysed. An all-pole model is used here because there is a number of deterministic methods available for computing the coefficients to a good approximation. Having evaluated the coefficients it is easy to determine the frequency response by evaluating the expression around the unit circle.

Figure 4 shows the results of using a maximum-entropy method, the autocorrelation method. It can be seen from a comparison of this result with that obtained from the FFT (figure 4b) that these techniques produce a spectrum from which it is much easier to identify targets: the main lobes are much narrower and sidelobes do not exist in the usual sense.

However, these methods do have the problem of choosing the best number of coefficients to be

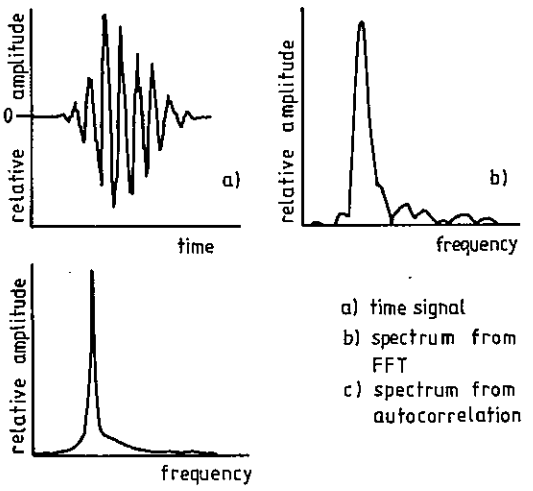


Figure 4 Comparison of autocorrelation and FFT

used, although in this particular application this has not been a serious problem.

The sharper target returns produced in the frequency domain by the use of maximum entropy methods are particularly important when target detection is being done by eye from the spectrum. This is illustrated in figure 5 where it can be seen that it is easier to pick out two objects close together with the non-Fourier method: the second target is less likely to be mistaken as a sidelobe.

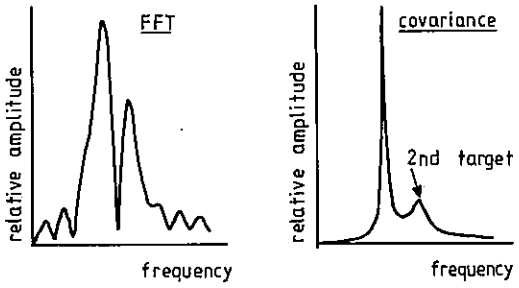


Figure 5 Advantage of using non-Fourier method

4. USE OF IMAGES

The object of treating the target return as an image is to enable more information to be obtained about the nature of the target found by the radar system. An image is obtained by scanning the radar across the surface to be investigated and storing a time return at each scan point.

Figure 6 shows here the return obtained from a

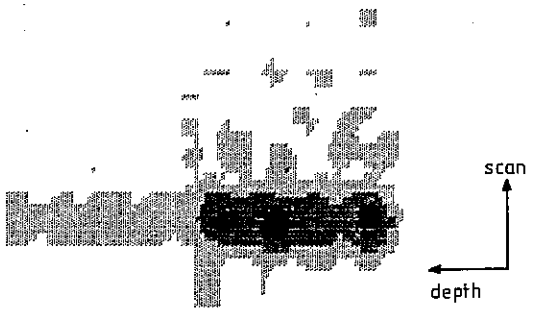


Figure 6 Image from a pipe behind a wall

plastic pipe behind a brick wall. The time returns from 15 scanning points were obtained in each case and a two-dimensional FFT performed, followed by two-dimensional low-pass digital filtering. Finally a one-dimensional FFT was performed along the scan direction to re-convert that dimension to scanning distance. Implicit in this process is interpolation and the final display is a 42 by 42 point image.

In the images shown in figure 6 the ghost targets behind the main peaks in the return from the pipe are readily visible. With this presentation it can be seen that the the operator of the radar has considerably more information about the target.

5. CONCLUSIONS

This paper has summarised the various signal processing techniques used to enhance the detection of hidden objects with a microwave FMCW radar. Without the use of signal processing these applications would not be feasible and research is continuing at QMC to determine even more effective ways of developing these techniques.

REFERENCES

[1] Skolnik M.I., "Introduction to radar systems", McGraw-Hill, 1969

[2] Carr A.G., Cuthbert L.G., and Olver A.D., "Digital signal processing for target detection in FMCW radar". Proc IEE part F, 1981, 128, 331-336

[3] Farmer G., Cuthbert L.G., Olver A.D. and Botros A.Z., "Distinguishing between types of hidden objects using an FMCW radar", Electronics Letters, Vol 20, No 20, 824-825

[4] Linkens D.A., "Non-Fourier methods of spectral analysis", Digital Signal Processing, Jones N.B. (Ed), Peter Peregrinus, 1982

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW METHOD OF ARRAY PROCESSING (Blind Reception Beamformer)

B.W. Dahanayake and K.M. Wong

Department of Electrical and Computer Engineering
McMaster University
Hamilton, Ontario, Canada
L8S 4L7

A new super-precision adaptive array processing technique is presented. This technique is based on the process of joint bearing estimation and interference cancellation, and it provides an exact cancellation of interferences. It directly operates on the input data adaptively and provides optimal solution at each snapshot. The beamformer can receive signals coming from different and unknown directions separately (listening device) and hence it is named as Blind Reception (BR) beamformer. BR beamformer is robust to uncertainties in the desired signal direction, strength of the interferences, and random sensor motion in the plane of the array. A systolic array structure suitable for the VLSI implementation of the BR beamformer is also presented.

INTRODUCTION

In general, the classical problem of array processing can be divided into three categories, viz.,

1. Location of sources or targets,
2. Reception of a signal coming from a known direction while suppressing all the other interferences,
3. Reception of all the signals coming from different and unknown directions separately while suppressing all the interferences (listening device).

Conventional array processing techniques can provide an approximate solution to the first and second tasks described above, but generally the knowledge of the desired direction is required and also an exact cancellation of interferences may not be achieved. Further, the conventional techniques do not provide any control over the interference cancellation capabilities. None of the available techniques provide a solution to the third task given above, due to the fact that the desired direction is unknown.

Most of the techniques available for computing the beamformer filter weights, do so by maximizing the overall signal-to-noise ratio at the system output, where noise includes sensor noise and directional interferences. This overall signal-to-noise ratio maximization provides an optimal solution as long as the sensor noise and directional interferences are equally detrimental to the system performance. However, there do exist situations where the jamming signals may be more damaging to the performance than sensor noise (e.g. war time jamming). The true measure of the system performance is our ability to demodulate the received signal and determine the information being sent and this performance measure is not a function of array SNR alone. Sometimes the structure of the interfering signals is also germane. For these cases, it is necessary for modeling interferences (or jammers) and sensor noise (or thermal noise) separately [1,2].

In this paper we propose a new method named Blind Reception (BR) beamformer which provides solutions to all the array processing tasks described above by treating interferences and sensor noise separately and placing exact or required amount of nulls at the interferences. BR beamformer does not demand the knowledge of the desired signal and hence it can be used to receive any signal which is present at the array without being affected by the other signals (interferences). Therefore, the BR beamformer acts a listening device. BR beamformer is specially suitable for removing directional interferences from point-to-point communications systems.

Signal Model:
We consider a linear array consisting of M number of elements. The output of the m^{th} element of the array $x_m(t)$ due to the incoming signals, θ_i, i = 1,2, ..., K can be written as,

$$x_m(t) = \sum_{i=1}^{K} s(m, \theta_i, t) + v_m(t),$$

$$s(m, \theta_i, t) = A_i \exp\left[j\left\{ \omega_c t + \frac{2\pi(m-1)d}{\lambda} \sin\theta_i + \psi_i \right\} \right]$$

$$m = 1, 2, \ldots, M$$

where

ψ_i is the phase of the i^{th} signal,
$s(m,\theta_i,t)$ is the output of the m^{th} element due to the signal coming from the direction θ_i,
λ is the wave length,
d is the sensor separation (nominal distance = 0.5 λ)
A_i is the amplitude of the i^{th} signal,
ω_c is the angular frequency
$v_m(t)$ is the sensor noise.

We assume that the incoming signals are uncorrelated. Consider the n^{th} block of data $x_m[(n-1)L + \ell]$, where $\ell = 0,1,\ldots,L-1$ and $n = 1,2,\ldots,N$. Taking the fast Fourier transform of this n^{th} block of data and extracting the amplitude and phase information of the frequency of interest ω_c, and denoting it by $u_m(n)$, where

$$u_m(n) = \sum_{i=1}^{K} A_i \exp\left[j\left\{ \omega_c t + \frac{2\pi(m-1)d}{\lambda} \sin\theta_i + \psi_i \right\} \right] + v_m(n),$$

we obtain an array u(n) at n^{th} snapshot such that

$$u(n) = [u_1(n), u_2(n), \ldots, u_M(n)]^T.$$

where $[\cdot]^T$ denotes the transpose. If we have several frequencies of interest, i.e. if the incoming signal is broad band, we have to consider several arrays. Interference cancellation will be done in frequency domain utilizing the array u(n) for all the frequencies of interest separately and finally reconstruction will be done by inverse fast Fourier transform (IFFT).

Blind Reception Beamformer:
The input data matrix at n^{th} snapshot, U(n) is given by,

$$U(n) = [u(1), u(2), \ldots, u(n)]^T.$$

U(n) is n×M dimension data matrix and is growing in size with time. U(n) also contains all the information about the incoming signals and noise. Since U(n) is growing in size, direct use of it is not suitable for adaptive array processing realization. Therefere, we first transform the floating dimension matrix U(n) to a fixed dimension upper triangular matrix R(n) adaptively starting from the first snapshot to n^{th} snapshot by using the Jacobi/Givens rotation Q(n). This can be performed adaptively by using the systolic array structure given by Gentleman and Kung [3] for matrix triangularization.

Let the filter weights at n^{th} snapshot be $w^*(n)$, then the output vector of the beamformer can be written as,

$$e_n(n) = U(n)\ w^*(n) \tag{1}$$

By applying Jacobi/Givens rotation Q(n) to the eqn. 1, we obtain

$$Q(n)\ e_n(n) = Q(n)\ U(n)\ w^*(n) \tag{2}$$

where, Q(n) is a unitary matrix.

We select Q(n) such that,

$$Q(n)\ U(n) = \begin{bmatrix} R(n) \\ 0 \end{bmatrix} \tag{3}$$

where, R(n) is an upper triangular matrix of order M×M. Eq. (2) can be written as,

$$Q(n)\ U(n) = \begin{bmatrix} R(n) \\ 0 \end{bmatrix}\ w^*(n) \tag{4}$$

The beamformer output at m^{th} snapshot, e(n) is given by,

$$\begin{aligned} e(n) &= w^H(n)\ u(n) \\ &= e_n(n). \end{aligned}$$

Now, by making the assumption that the spatial noise cross spectral density matrix at the frequency of interest is approximately equal to the spatial cross spectral density matrix of the adjacent frequency bin [4,5], we can obtain spatial noise cross spectral density matrix from the noise data matrix $U_v(n)$, obtained by the off-frequency bin at n^{th} snapshot. As we did for the data matrix, by applying the Jacobi/Givens rotation we transform $U_v(n)$ to an upper triangular matrix $R_v(n)$, i.e.

$$Q_v(n) \cdot U_v(n) = \begin{bmatrix} R_v(n) \\ 0 \end{bmatrix} \tag{5}$$

Since Q(n) and $Q_v(n)$ are unitary matrices, the information content of the matrices R(n) and $R_v(n)$ will be same as the information content of the data matrices U(n) and $U_v(n)$. Therefore R(n) and $R_v(n)$ can be used for further processing instead of U(n) and $U_v(n)$. Now let us consider the procedure for the Blind Reception beamformer.

In the Blind Reception beamformer, we perform the processing in two steps at each snapshot:

1. Estimate the bearings of the incoming signals by using the upper triangular matrices R(n) and $R_v(n)$.
2. Use multiple constraint minimum variance distortionless response (MCMVDR) method to place exact or a required amount of nulls at the interferences. (If the desired signal is known approximately, estimated bearing closer to that can be used as the desired signal. If no a priori knowledge of the desired signal direction is available, several beamformers have to be designed to receive the signals coming from each direction separately).

Bearing Information Extraction:
In order to implement the Blind Reception beamformer, first, we have to estimate the bearings of the incoming signals at each and every snapshot based on the upper triangular matrices R(n) and $R_v(n)$. Even though there are several bearing estimation techniques available, POP-MUSIC-LP (or POP-MUSIC-ML) [5] is more suitable due to its super resolution capabilities over other techniques [2]. In the POP-MUSIC-LP (or POP-MUSIC-ML) we construct a polynomial, the roots of which on or near the unit circle, correspond to the bearings of the incoming signals. POP-MUSIC is based on subspace decomposition and hence the generalized singular value decomposition (GSVD) on R(n) can be used to obtain the singular vectors which describe the signal and noise subspace basis vectors. GSVD of R(n) and $R_v(n)$ is computationally more stable than eigenvalue/eigenvector decomposition of cross spectral density matrix. The generalized singular value decomposition can be written as:

$$R^H(n) \cdot R(n) \cdot v = \lambda \cdot R_v^H(n) \cdot R_v(n) \cdot v \tag{6}$$

where $[\cdot]^H$ denotes the Hermititan transpose.

Van Loan [6,4] has given a procedure for obtaining the generalized singular value decomposition from the input data matrices. As we mention before, the use of input data matrices are computationally more involved and not suitable for adaptivity. Therefore, we use the same algorithm given by Van Loan [6,4], but instead of the input data matrices we use the upper traingularized matrices with fixed dimensions and hence we reduce computation, and achieve better structure for adaptivity. The GSVD of the eqn. 6 can be obtained by using the following procedure [4,2]:

1. Compute the singular value decomposition of the matrix G,

$$G = \begin{bmatrix} R(n) \\ R_v(n) \end{bmatrix} = \begin{bmatrix} Q(n) \\ Q_v(n) \end{bmatrix} \Sigma\ Z^H(n)$$

where $\Sigma = \text{diag}(\sigma_1, \sigma_2 \ldots, \sigma_n)$, Z(n) is unitary and Q(n) and $Q_v(n)$ satisfy the equality,

$$Q^H(n) \cdot Q(n) + Q_v^H(n) \cdot Q_v(n) = I$$

where I is an unitary matrix.

2. Compute unitary matrices U(n), $U_v(n)$ and Y(n) such that,

$$U^H(n) \cdot Q(n) \cdot Y(n) = \text{diag}(c_1, c_2, \ldots, c_M), \quad c_i \geq 0$$

$$U_v^H(n) \cdot Q_v(n) \cdot Y(n) = \text{diag}(s_1, s_2, \ldots, s_M), \quad s_i \geq 0$$

where $c_i^2 + s_i^2 = 1, \quad i = 1, 2, \ldots, M$

3. Set

$$V(n) = [v_1(n), v_2(n), \ldots, v_M(n)] = Z(n) \cdot \Sigma^{-1}\ Y(n)$$

and

$$\lambda_i^{1/2} = c_i / s_i$$

Then $\lambda_i^{1/2}$ denotes the generalized singular values and the column of V(n) consist of the generalized singular vectors at n^{th} snapshot. The matrix V(n) that contains the basis vectors of the signal and noise subspaces can be separated into its counterparts, i.e.

$$V(n) = [V_S(n), V_N(n)],$$

where $\mathbf{V}_s(n)$ and $\mathbf{V}_N(n)$ represents the signal subspace and noise subspace respectively. The number of signals, K, can be determined by using information theoretic criteria such as Akaike information criterion (AIC) or minimum description length criterion (MDLC) [7]. Once K is determined we choose the singular vectors corresponding to the K largest singular values as the signal subspace basis whereas the remaining M-K singular vectors form the basis for the noise subspace. Once the subspaces $\mathbf{V}_s(n)$ and $\mathbf{V}_N(n)$ are found, the bearings of the incoming signals can be estimated using the roots of the polynomials $G(z^{-1})$ or $H(z^{-1})$ which correspond to the POP-MUSIC-ML and POP-MUSIC-LP respectively [2]. $G(z^{-1})$ and $H(z^{-1})$ can be written as,

$$G(z^{-1}) = \sum_{i=-M+1}^{M-1} g_i(n) \cdot z^{-i}$$

$$H(z^{-1}) = \sum_{i=1}^{M} h_i(n) \cdot z^{-(i-1)} \tag{7}$$

where

$$g_{i-1}(n) = \sum_{j=1}^{M} p_{N_{j,i+j-1}}(n)$$

$$g_{-i}(n) = g_i^*(n) \qquad \text{for } i > 0,$$

$p_{i,j}(n)$ is the $(i,j)^{th}$ element of thematrix $\mathbf{P}_N(n)$ given by

$$\mathbf{P}_N(n) = \mathbf{V}_N(n) \cdot \mathbf{V}_N^H(n).$$

and $\mathbf{h}(n)$ is given by

$$\mathbf{P}_N(n) = \begin{bmatrix} \mathbf{h}^T(n) \\ \mathbf{P}'_N(n) \end{bmatrix}$$

$G(z^{-1})$ has $2(M-1)$ roots which occur in conjugate reciprocal pairs, i.e. if z_k is a root of $G(z^{-1})$, then $(z_k^*)^{-1}$ is also a root of $G(z^{-1})$. The roots of $G(z^{-1})$ or $H(z^{-1})$ on or near within the unit circle at $\exp(j(2\pi d/\lambda)\sin\theta_k)$ correspond to the signal arrival from the direction θ_k [2].

Lemma-1 [2]

The POP-MUSIC-LP polynomial coefficient $\mathbf{h}(n)$ of eqn. 7 has the relationship

$$\mathbf{h}(n) = \|\mathbf{v}_N(n)\|^2 \begin{bmatrix} 1 \\ \|\mathbf{v}_N(n)\|^{-2} \mathbf{V}'^*_N(n) \cdot \mathbf{v}_N(n) \end{bmatrix}$$

$$= (1 - \|\mathbf{v}_s(n)\|^2) \begin{bmatrix} 1 \\ -(1 - \|\mathbf{v}_s(n)\|^2)^{-1} \mathbf{V}'^*_s(n) \cdot \mathbf{v}_s(n) \end{bmatrix}$$

where

$$\mathbf{V}_s(n) = \begin{bmatrix} \mathbf{v}_s^T(n) \\ \mathbf{V}'_s(n) \end{bmatrix}, \quad \mathbf{V}_N(n) = \begin{bmatrix} \mathbf{v}_N^T(n) \\ \mathbf{V}'_N(n) \end{bmatrix},$$

and $[\cdot]^*$ denotes the conjugate.

Proof is given in [2].

From the lemma-1, it is obvious that the so-called minimum norm technique proposed by Kumaresan and Tuft [6] has no difference from the MUSIC method even though some authors have treated it differently.

Multiple Constraint Minimum Variance Distortionless Response (MCMVDR):

Once the bearings of the incoming signals are obtained our next step in the BR beamformer is to find the filter weight in receiving a selected signal while placing nulls at all the other interferences. This can be achieved by using the multiple contraint minimum variance distortionless response (MDVDR) formulated in lemma-2, in a form suitable for systolic array implementation.

Lemma-2: [2]

Let $\mathbf{D}(\theta)$ be the direction vector matrix given by

$$\mathbf{D}(\theta) = [\mathbf{d}(\theta_T), \mathbf{d}(\theta_{I1}), \ldots \mathbf{d}(\theta_{Ik-1})]$$

If we want to minimize the quantity $\|\mathbf{Q}(n)\mathbf{e}_n(n)\|^2$ with respect to the weight vector $\mathbf{w}(n)$ subject to the constraint,

$$\mathbf{D}^H(\theta) \cdot \mathbf{w}(n) = \mathbf{c}$$

then the optimal weight vector $\mathbf{w}_{opt}(n)$ is given by

$$\mathbf{w}_{opt}(n) = \mathbf{D}_2(\theta) \cdot \mathbf{c}_1$$

where the following relationships define the various quantities,

$$\mathbf{c} = (1, \delta, \delta, \ldots, \delta)^T$$
$$\mathbf{c}_1 = \mathbf{A}^{-1} \cdot \mathbf{c}$$
$$\mathbf{A} = \mathbf{D}_1(\theta)^H \cdot \mathbf{D}_1(\theta)$$
$$\mathbf{D}_1(\theta) = [\mathbf{a}(n,\theta_T), \mathbf{a}(n,\theta_{I1}), \ldots, \mathbf{a}(n, \theta_{IK-1})]$$
$$\mathbf{D}_2(\theta) = [\mathbf{b}(n,\theta_T), \mathbf{b}(n,\theta_{I1}), \ldots, \mathbf{b}(n, \theta_{IK-1})]$$
$$\mathbf{R}^H(n) \cdot \mathbf{a}(n,\theta) = \mathbf{d}(\theta)$$
$$\mathbf{R}(n) \cdot \mathbf{b}(n,\theta) = \mathbf{a}(n,\theta),$$

K is the number of signals and δ is a small number chosen depending on the amount of null required.

Proof is given in [2].

By selecting $\delta = 0$, one can cancel the interference exactly. The value of the δ can be chosen according to the interference rejection required. In many cases interferences and sensor noise reduction can be compromised by selecting δ such that the system performance is optimized. BR beamformer also provides an ability to control the null amplitude of the interferences according to the user specification over all the other existing beamformers. Several beamformers can be designed in order to receive signals that appear at the array separately. Systolic array structure for the POP-MUSIC bearing estimation is given in [2]. Further, the BR beamformer based on joint bearing estimation and interference cancellation given in Lemma-2 is in a suitable form for systolic array implementation and can be implemented using some simple structures such as matrix-vector multiplication, triangular arrays, and linear arrays for forward and backward substititution [2]. The performance of the beamformer depends on the resolution capabilities of the bearing estimation technique used. We have shown in [2] that the POP-MUSIC spectral estimation technique is insensitive to the random sensor motion in the plane of the array and hence the BR beamformer. In general, subspace rotation that appears due to the errors introduced by various effects such as sensor perturbation, finite snapshots and quantization can limit the resolution capabilities of the POP-MUSIC. This limitation can be overcome by using the Rotational Correction-Multiple signal classification (ROC-MUSIC) we have proposed in [2]. ROC-MUSIC provides superior resolution capabilities overall the existing techniques. The schematic diagram of the BR beamformer is shown in Fig. 1. BR beamformer is not limited to the situation where signals are temporarily uncorrelated. In the case of correlated signals, BR beamformer followed by adaptive spatial data

smoothing procedure (ASDSP) given in [2] provide successful performance.

Example:
For computer simulation, the following signal and noise environments are used:

(a) Number of signals = 2
(b) Incoming signals are uncorrelated
(c) Direction of the signals = $\sin^{-1}(0.2)$, $\sin^{-1}(-0.4)$
(d) Signal-to-noise ratio (SNR) = 20 dB
(e) Number of elements in the array = 5
(f) Separation of the elements = 0.5 λ
(g) Noise is spatially uncorrelated
(h) Number of snapshots = 100 and $\delta = 0$
(i) Interference-to-noise ratio (INR) = 40, 30, 20 dB.

Spatial response $20\log|w^H(n)\cdot d(\theta)|$ for the BR beamformer at the 100th snapshot is shown in Fig-2 which clearly indicates that the interference at $\sin^{-1}(-0.4)$ is exactly cancelled within the computational limitations while the signal at the $\sin^{-1}(0.2)$ is unaltered.

Discussion:
We have presented a super-precision blind reception beamformer and a pipeline, modular systolic array structure suitable for VLSI implementation. BR beamformer is capable of eliminating the interferences exactly or to the required amount and it recovers the desired signal without altering them. The performance of the BR beamformer does not demand the knowledge of the desired signal and hence it can be used to receive unknown signals separately (listening device). Since the POP-MUSIC method is robust to the random sensor motion in the plane of the array, BR beamformer is also robust to it. Real time processing can be made possible by designing the beamformer to perform a cycle of operation within the time interval used to obtain a single snapshot. The BR beamformer and the concept of beamforming as a joint bearing estimation and interference cancellation process will provide a new direction towards super-precision adaptive array processing.

References:

[1] Citron, T.K. and T. Kailath, (1983), "Eigenvalue Methods and Beamforming: A First Approach", IEE, ICASSP.
[2] Dahanayake, B.W. and K.M. Wong, (1985), "High Resolution Adaptive Array Processing Techniques", Research Report CRL-148, Dept. of Electrical and Computer Eng., McMaster University.
[3] Gentleman, W.M. and H.T. Kung, (1981), "Matrix Triangularization by Systolic Arrays", SPIE, vol. 198, Real-Time Signal Processing IV.
[4] Speiser, J.M. and C. Van Loan, (1984), "Signal Processing Using the Generalized Singular Value Decomposition", SPIE, vol. 495, Real-Time Signal Processing VII.
[5] Dahanayake, B.W. and K.M. Wong, (1986), "Proper Orthogonal Projection-Multiple Signal Classification (POP-MUSIC)", IEEE, ICASSP.
[6] Van Loan, C.F., (1976), "Generalizing the Singular Value Decomposition", SIAM J. Numer. Anal. vol. 13, No. 1.
[7] Wax, M. and T. Kailath, (1984), "Determining the Number of Signals by Information Theoretic Criterion", IEEE, ICASSP.
[8] Kumaresan, R. and D.W. Tuft., (1983), "Estimating the Angles of Arrival of Multiple Plane Waves", IEEE Trans. Aerospace and Electronic Systems, vol. AES-19.

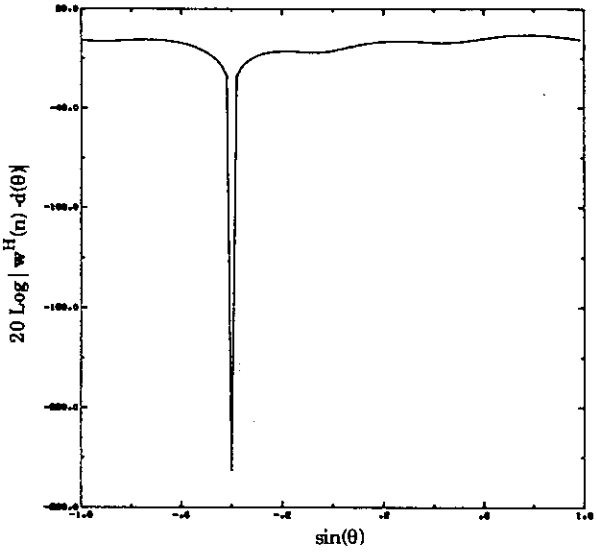


Fig. 2: Spatial Response of the Blind Reception Beamformer

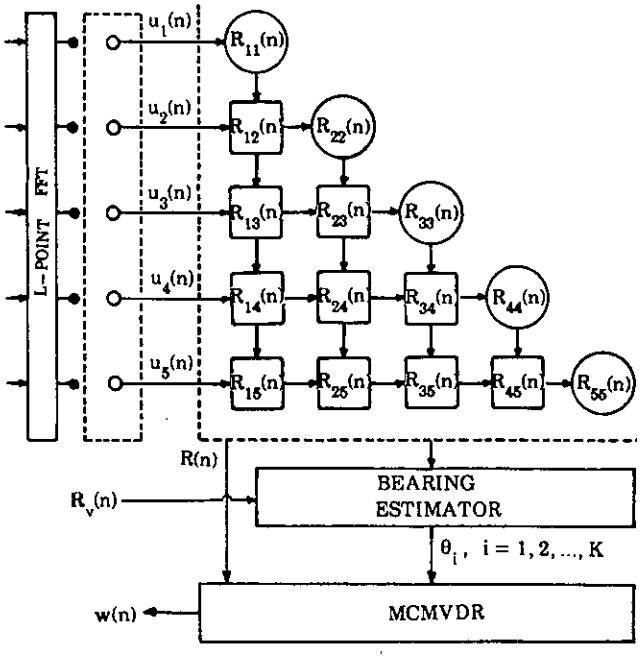


Fig. 1: Schematic Diagram of BR Beamformer

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

RADAR TARGET DETECTION WITH MTD PROCESSOR IN EXPONENTIALLY AND LOG NORMALLY DISTRIBUTED CLUTTER

Hermann Rohling

AEG Aktiengesellschaft, Research Institute, D-7900 Ulm

The probability of target detection in a white noise process can be calculated on the basis of P. Swerling's target fluctuation models /1/. In the present paper, statistical models with spatially non-homogeneous clutter reflectivity will also be studied. Measurements show that each radar cell in the range azimuth area of interest has a different mean clutter power. This situation is described in the present paper by a random process with log normally or exponentially distributed mean clutter power values.

1 INTRODUCTION

The performance of a radar system is determined by its probability of target detection (Pd) for a given constant probability of false alarm (Pfa). P. Swerling /1,2/ has studied target detection in a white noise environment in detail, including different target fluctuation models. In this analysis the radar signal r(t) consists of two terms, target echo s(t) and noise n(t).

$$r(t) = s(t) + n(t) \qquad (1)$$

Real radar environment exists not only in the presence of noise but also in the presence of a clutter background with varying behavior, (see /3-6/). In this case the radar signal r(t) consists of three terms: target echo s(t), clutter with different statistical behavior c(t), and noise n(t).

$$r(t) = s(t) + c(t) + n(t) \qquad (2)$$

In this paper, both steady and fluctuating targets (Swerling 0 and 1), and both a spatially *homogeneous* and *non-homogeneous* clutter environments are investigated.

2 MTD PROCESSOR

Modern radar digital signal processing units, e.g. a Moving Target Detector (MTD), use a Doppler filter bank, a CFAR circuit, and a high resolution ground clutter map (GCM) (Figure 1) in the automatic detection process for adaptation to the background clutter /7/.

2.1 Doppler filter bank

The Doppler filtering process is a very important tool in the MTD processor. Multiple narrow-band and overlapping Doppler filters cover the whole Doppler interval. The main task of the Doppler filter bank is target detection in heavy ground clutter, which means separating strong clutter from small target echoes.
Let us consider, as an example, a Doppler

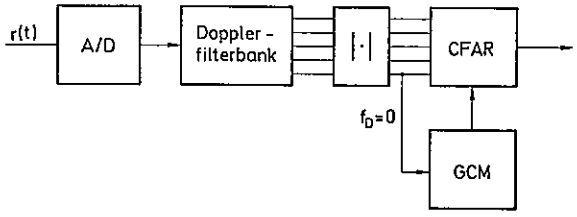


Figure 1: Moving Target Detector (MTD) with Doppler filter bank, CFAR circuit, and a high resolution ground clutter map (GCM).

filter bank consisting of 8 separate filters. Each filter coherently integrates 8 radar pulses. Figure 2 shows four typical filter transfer functions of the filter bank used for this task. The other four filters are mirror images thereof. These filters are designed with low sidelobe levels, especially in the ground clutter region (fD = 0), to prevent clutter signals from being coupled into the target signal filters. The complex valued filter coefficients are stored in ROMs and can be adapted and modified to a specific application.

Radar signals with low Doppler frequency, especially ground clutter signals, are detected by filters 1 and 8. Filters 2 - 7 (target signal filters) are designed to detect target signals with high Doppler frequency, and to suppress all low Doppler frequency signals.

According to eq. (2), the input signal consists of three different terms with mean clutter power C_{in}, signal power S_{in} and noise power N_{in} The clutter process is fully known, except for the clutter-to-noise ratio $(C/N)_{in}$. The unknown parameters of the target signal are the signal-to-noise ratio $(S/N)_{in}$ and the Doppler frequency fD.
Let w be the Doppler filter coefficient vector, and K_s, K_c, K_n the covariance matrices of the

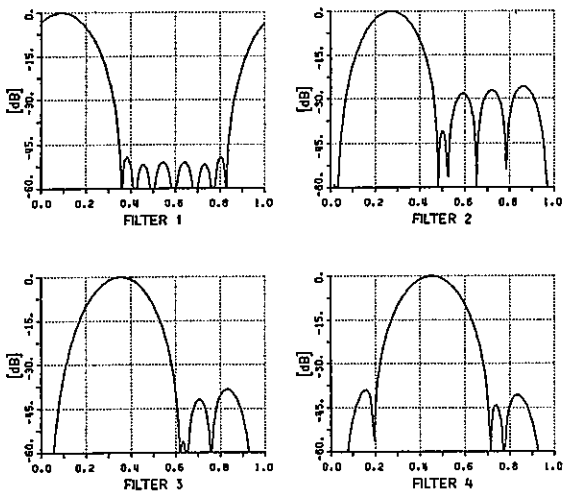


Figure 2: Typical filter transfer functions used in the Doppler filter bank with 8 separate filters. The other four filters are mirror images thereof.

target, clutter, and noise process, respectively. The corresponding signal parameters (mean power) at the filter output are

$$S_{out} = w^* K_s w \qquad C_{out} = w^* K_c w \qquad N_{out} = w^* K_n w$$

where w^* is the complex conjugate and transpose vector of w. We assume a normalized input noise level $N_{in} = 1$ (in this case the matrix K_n is the identity matrix I) and normalized filter coefficients $w^* w = 1$ which result in a normalized output noise power $N_{out} = 1$.

The relation between output and input signal parameters can be described as follows:

$$(S/N)_{out} = w^* K_s w / w^* K_n w = (S/N)_{in} \; ISN(w) \qquad (3)$$

$$(C/N)_{out} = w^* K_c w / w^* K_n w = (C/N)_{in} \; ICN(w) \qquad (4)$$

where ISN(w) and ICN(w) indicate two Doppler filter performance measures, namely the improvements in signal-to-noise and clutter-to-noise ratios, respectively.

$$ISN(w) = (S/N)_{out} \; / \; (S/N)_{in} = w^* K_s w / S_{in} \qquad (5)$$

$$ICN(w) = (C/N)_{out} \; / \; (C/N)_{in} = w^* K_c w / C_{in} \qquad (6)$$

Table 1 shows these Doppler filter performance measures for the filters shown in Figure 2. As Table 1 shows, the target Doppler filters 2-7 have very strong clutter suppression rates. In filters 3 - 6, the clutter residue will almost always be lower than the noise power; only filters 2 and 7 will have larger clutter residues in certain cases with heavy ground clutter in the input signal.

Filter number	ICN(w) /dB/	ISN(w) /dB/
1 and 8	4.2	7.5
2 and 7	-48.8	8.1
3 and 6	-74.9	7.5
4 and 5	-87.1	7.5

Table 1: Doppler filter Performance measures

2.2 Ground clutter map

Since there are different clutter background situations in the observation area, modern radar processing units use different CFAR systems in the automatic detection process for local adaptation to the background clutter, and in order to control their false alarm rate, viz.:

- constant amplitude threshold T1 in noise situations;
- high resolution clutter map (GCM) in ground clutter situations;
- range CFAR in local weather clutter situations.

A high resolution ground clutter map stores the mean clutter power in each range azimuth cell. The amplitudes at the output of filters 1 and 8 are averaged in each range azimuth cell and over several antenna scans. This estimation value is updated in each scan and each cell by a recursive algorithm. The threshold T2 is calculated in each range azimuth Doppler cell by a product of three terms, the mean clutter power C stored in the GCM, the ICN(w) value of the Doppler filter of interest, and a scale factor F2 used to adjust the false alarm rate.

$$T2 = F2 \cdot C \cdot ICN(w) \qquad (7)$$

2.3 Range CFAR

In the weather clutter case, the data from a single cell over several scans can be used with reservation only for estimation of the statistical parameters since the weather clutter regions might move and change from scan to scan. Instead, the data from a single scan and local reference window enter into an algorithm for the description of statistical parameters used for target detection and clutter suppression. The threshold value T3 is calculated in the range CFAR procedure as a product

$$T3 = F3 \cdot Z \qquad (8)$$

where F3 is a scaling factor and Z an estimate of the local mean clutter power.

For each range azimuth Doppler cell, three different thresholds are calculated. The final threshold T is estimated as the *maximum* of the three thresholds above.

3 STATISTICAL ANALYSIS

In *homogeneous* clutter situations, the clutter power C is assumed to be identically distributed in each cell and inside the range-azimuth area of interest. This assumption implies that the average density of scatterers (the mean clutter power $\hat{C}$) in each range-azimuth cell has a constant value. In this case, the clutter can be described by a gaussian white noise process.

If the mean clutter power $\hat{C}$ is different from cell to cell, the situation is called *non-homogeneous*. In the case of ground clutter, for example, there exists a non-homogeneous spatial distribution which is approximated and described in statistical models by a log-normal distribution. The probability density function (pdf) of this distribution has two parameters, a and b.

$$p(\hat{C}|a,b) = \begin{cases} \frac{1}{\sqrt{2\pi}\cdot b\,\hat{C}}\, e^{-\frac{(\ln\hat{C}-a)^2}{2b^2}} & \hat{C}>0 \\ 0 & \text{otherwise} \end{cases} \qquad (9)$$

The expectation $E(\hat{C})$ and the median $MED(\hat{C})$ of this distribution are

$$E(\hat{C}|a,b) = C_o = \exp(a+b^2/2) \qquad (10)$$

$$MED(\hat{C}|a,b) = \exp(a) \qquad (11)$$

In this paper, the detection performance of an MTD processor is considered from a probability-of-detection standpoint and for a log normal distribution with $E(\hat{C}|a,b)/Med(\hat{C}|a,b) = 20$ dB. It is assumed that the estimated average clutter power $\hat{C}$ in each cell is stored in a GCM which has the full range-azimuth resolution of the radar-raster image. The decision threshold T2 is calculated according to eq. (7).

In a first step, a target/clutter situation (without noise) will be analyzed. Figure 3a shows a realization of statistically independent and log normally distributed random numbers figured in range-azimuth coordinates (non-homogeneous clutter situation) in graphic form. The values shown in Figure 3a must be interpreted as the mean clutter power $\hat{C}$ in each range-azimuth cell. In the mathematical formulation of the clutter model it is assumed that ground clutter has a Rayleigh distribution in each cell with log-normally distributed mean clutter power $\hat{C}$ from cell to cell.

Figure 3b gives a comparison between detection situations for a Swerling 1 target with MTD processor in spatially homogeneous environment and log normally distributed ground clutter environment, respectively, ($Pfa = 10^{-6}$).
Figure 3b shows the probability of detection (Pd) curves as a function of S/C_o (signal-to-average-clutter power C_o inside the range-azimuth area of interest, $E(\hat{C})=C_o$, eq. (10)).

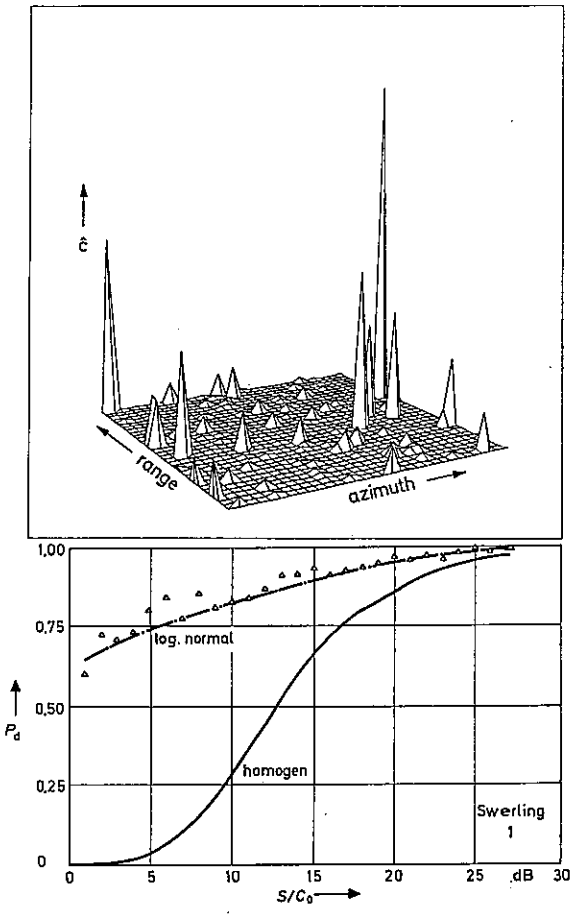


Figure 3:
a) Spatial clutter behavior described by log normally distributed mean clutter power $\hat{C}$.
b) Probability of detection (Pd) for homogeneously and log normally distributed mean clutter power as a function of S/C_o (signal-to-average-clutter power inside the area of interest). Swerling 1 case.

This analysis describes the target detection situation in ground clutter Doppler filters 1 and 8, the so-called superclutter visibility which means target detection in the Doppler region of $-0.2/T < fD < 0.2/T$, where 1/T indicates the pulse repetition frequency (PRF). The target detection rate in a log-normally distributed ground clutter region is much higher than in a homogeneous clutter situation. This fact is important, especially for tracking tangentially flying targets.

Detection behavior in ground clutter situations is quite different in the filters 2 - 7 since the remaining clutter residue $(C/N)_{out}$ is frequently very small (see eq. (4) and Table 1). In this case the noise influence can no longer be neglected in the statistical analysis. In the case of a superimposed target echo, log-normally distributed clutter echoes, and

gaussian white noise, Figure 4 shows the detection behavior of these Doppler filters in a log normally distributed ground clutter situation as a function of the $(S/N)_{out}$ and the ground clutter residue $(C/N)_{out}$. Figure 4 contains as a reference the same curves (dashed lines) for a homogeneous ground clutter situation.

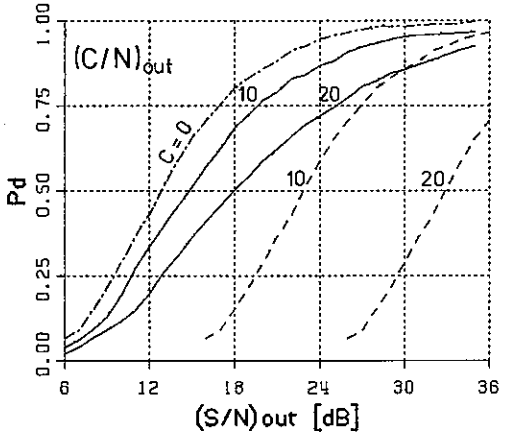


Figure 4:
Detection behavior of filters 2 - 7 in heavy ground clutter. The parameter on the Pd curves describes the clutter residue $(C/N)_{out}$

In spatial exponentially distributed clutter situations (as opposed to log normally distributed clutter) the deviation from homogeneous clutter is smaller. An example of non-homogeneous clutter with spatial exponentially distributed mean clutter power C is shown in Figure 5a, and the comparison between homogeneous and exponentially distributed clutter is shown in Figure 5b in the Pd-S/C_o diagram. The comparisons show that the detection behavior depends strongly on the assumptions made on the clutter background.

4 CONCLUSION

The target detection process using an MTD processor has been improved in the whole Doppler frequency interval when compared with conventional MTI.

- The so-called superclutter visibility, which means detection of radar signals with low Doppler frequencies (e.g. tangentially flying targets), has been increased because of the improved adaptation to the ground clutter environment attainable with the GCM.

- At the same time, the subclutter visibility due to the Doppler frequency interval $0.2/T < fD < 0.8/T$ (target signal filters 2-7) has been increased as a consequence of the improved Doppler filter process.

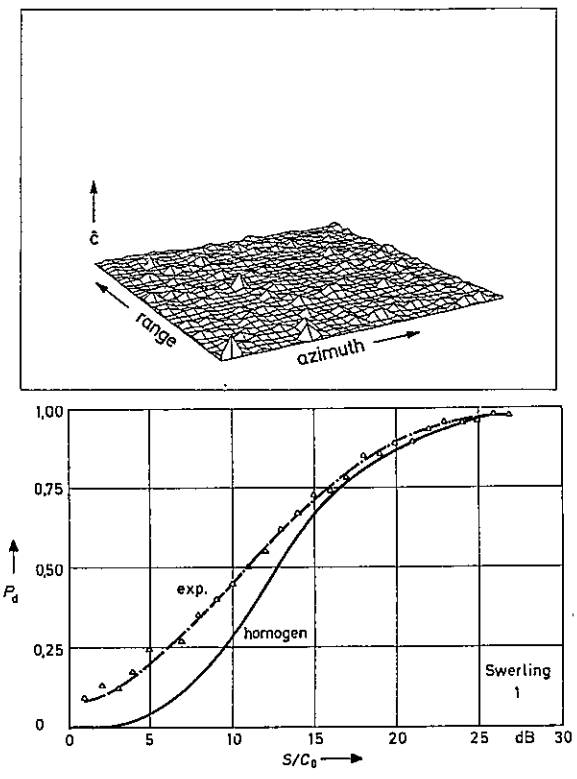


Figure 5:
a) Spatial clutter behavior described by exponentially distributed mean clutter power C.
b) Probability of detection (Pd) for homogeneously and exponentially distributed mean clutter power as a function of S/C_o (signal-to-average-clutter power inside the area of interest). Swerling 1 case.

5 REFERENCES

/1/ Swerling,P.; Probability of Detection for Fluctuating Targets, IRE Trans. Inf. Theory, Band IT-6, No. 2 (1960), pp. 269-308

/2/ DiFranco, J.V., Rubin. W.L.; Radar Detection, Prentice-Hall, London 1968

/3/ Goldstein, G.B. False-Alarm Regulation in Log-Normal and Weibull Clutter, IEEE Trans., AES-9, No.1, Jan. 73, pp. 84-92

/4/ Schleher, D.C., MTI Detection Performance in Rayleigh and Log-Normal Clutter, IEEE Radar Conference, Washington 1980, pp 299-304

/5/ Szajnowski, W.J., Discrimination between Log-Normal and Weibull Clutter, IEEE Trans., AES-13, No. 5, Sept. 77, pp 480-485

/6/ Sekine, M., et.al. Suppression of Weibull distributed Weather Clutter, IEEE Radar Conference, Washington 1980, pp 294-298

/7/ Ludloff, A., et. al., Doppler processing, waveform design and performance measures for some pulsed Doppler and MTD radars. Ortung und Navigation, 3 (1981), pp. 417-457, 1 (1984), pp. 5-54

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

COHERENT TRACKING USING PULSE DOPPLER SODAR ("ULTRASONIC RADAR") - SOME ROBOTICS APPLICATIONS

Tommy Jonsson, Per Klöör, Börje Salomonsson, Åke Wernersson
Swedish Defence Research Institute
Box 1165, 581 11 Linköping, Sweden

ABSTRACT: In a context of coherent ultrasonics, it is studied how accurately measurements of the motion of non-overlapping objects can be made. The applications are in the field of robotics for e.g. tracking a linear structure, following a surface etc. Emphasis is laid on cases when the resolution is better than a wavelength.

1. INTRODUCTION - THE PROBLEM

In robotics, there is a need for short range measurements of geometrical quantities. Coherent pulsed sodar (SOnic Detection And Ranging) is complementary to vision, it has a fairly good range resolution, is very useful for measuring changes in range and has poor angular resolution. Attractive and competitive applications are of the type

- Detecting and counting the number of moving parts on, say, a conveyor belt.
- Measuring displacement - also vectorial
- Measuring the (perpendicular) angle to a surface (and curvature).

There are many other applications. Some introductory results are given in this paper. We do not elaborate on application details.

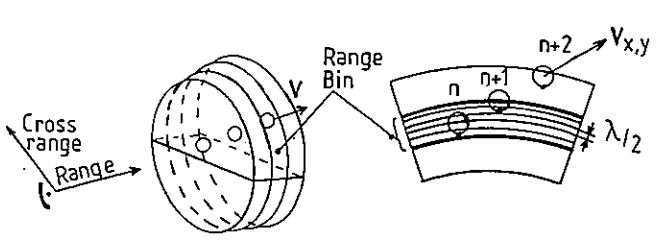
As in radar, the phrase "pulse Doppler" sodar is used. The change of frequency due to Doppler shift corresponds to a phase shift between the different pulses.

2. A SIMPLIFIED SIGNAL MODEL

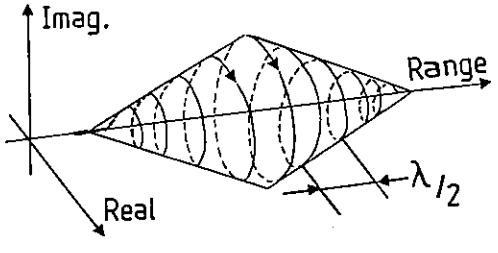
Fig 1 shows a small object moving through the range bins of a coherent sodar. Each bin is a thin spherical slice. The radial thickness is determined by the pulse length while the angular cross range is, essentially, equal to the size of the main lobe. Further, using coherent pulses, each slice is subdivided into a number of half-wavelength ($\lambda/2$)- "shells".
A small object moving within the range bin of a coherent sodar has an echo sequence

$$S(n)=\exp\left[-j\ (4\pi r_n/\lambda + \varphi)\right] \qquad (1)$$

Fig. 1 Principles of pulsed coherent sodar. Doppler shift is equivalent to phase shift.



Range bins in 3-D as determined by pulse length and antenna pattern.



Projection into a plane. The coherence gives the motion in the fixed "$\lambda/2$-shells".

The echo signal in one range gate as the object passes through.

Partially supported by the Swedish Board for Technical Development 85-3944.

Nomenclature:
r_n = true distance at pulse n
r(n) = $2\pi \cdot$ true distance $/\lambda$
R(n) as r(n) but two ways (also bistatic)

where r_n= the range at pulse n and φ is an unknown phase angle. The geometrical picture of S(n) is a phasor that rotates one turn for each "$\lambda/2$-shell" The rate of rotation is proportional to the radial velocity but only measured mod (2π). The observed sequence Y(n) is modelled as

$$Y(n)=A\cdot S(n)+W(n) \qquad (2)$$

where W is a white complex Gaussian sequence and A is an (almost) constant amplitude.

The absolute position is discretized by the size of the range bin. As is apparent from Fig 1, the radial displacement can (except for ambiguity and noise) be measured within a fraction of $\lambda/2$.

2.1. Models of the Objects

To simplify the signal processing the model assumed is a locally convex object such that there is at most one echo in each range gate. Except at sharp corners, geometrical optics can be used to describe the scattering - compare /1/. As the object moves, Fig 2 illustrates how the vectorial displacement of the reflection points differs from the displacement of the object. The goal is to have displacement errors that are less than a quarter of a wavelength. The figure shows, exaggerated, that the shape of the object has to be considered.

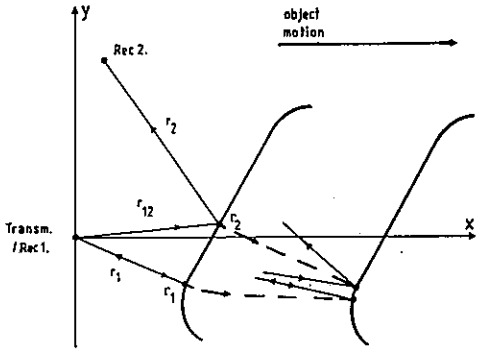


Fig 2. Given a moving locally convex object. The figure illustrates how the monostatic and one bistatic reflection point moves on the surface. The displacement of the reflection points differes from the displacement of the object.

3. TRACKING MOVING OBJECTS - METHODS

Depending on the equipment and on the signal processing, there are essentially three different types of tracking methods. They can be described as:

1. Estimating the velocity from position measurements.
2. Using both position- and Doppler measurements to estimate the trajectory.
3. Coherent tracking i.e. following the object without "phase slip".

In conventional radar tracking a sequence of positions are measured. From these measurements the velocity is estimated. Coherence, if used, only separates the moving objects from an almost stationary background - MTI filtering.

The natural improvement is to use a larger number of coherent pulses in order to estimate the velocity of the moving object. Using both position and Doppler the trajectory can be estimated more accurately. Especially useful when predicting trajectories.

The third tracking method is to try to follow the phase of the echoes and "to count the number of phase rotations". Coherent tracking is an appropriate name. If the motion is smooth and the error in the phase measurements are small the estimation of the displacement can be done more accurately than is possible with the previous methods. Complications like "range glint" occurs if there is more than one scatterer in each range gate. If the FFT is used as a "prefilter" (and to sort out the object from the background) coherent tracking does not have to be more sensitive to noise than the previous methods. This paper is just an example. Applications are towards robotics but without elaboration on application details.

4. THE PHASE RESIDUAL

For a moving object the phase in Y(n) will rotate rapidly. To "slow down the phase variations" we introduce

$$Z(n)=\overline{S'(n)}Y(n)=\exp\left[j(\varphi'+R'(n))\right]\cdot Y(n) \qquad (3)$$

as the phase unwrapping of Y(n) around the trajectory R'(n). If without noise, the true and the assumed trajectories are equal then Z(n)=A ; Y(n) have been backrotated to stand-still. More over, if $\varphi = \varphi'$ the stationary value of Z(n) is a point on the positive real axis. Hence, the problem is to find a Z(n) sequence that is as close as possible to a point on the positive real axis. The phase residual e(n) is defined as the angle of Z(n), i.e.

$$e(n)=\underline{/Z(n)}=\arctan(\mathrm{Im}Z(n)/\mathrm{Re}Z(n)). \qquad (4)$$

e(n) is the "correct measure" of the tracking error.

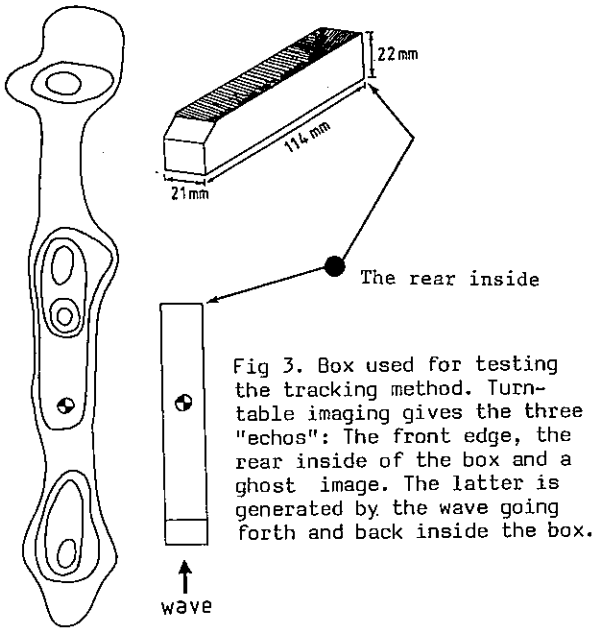
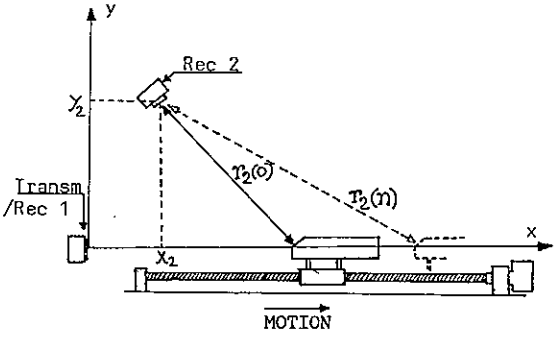


Fig 3. Box used for testing the tracking method. Turntable imaging gives the three "echos": The front edge, the rear inside of the box and a ghost image. The latter is generated by the wave going forth and back inside the box.



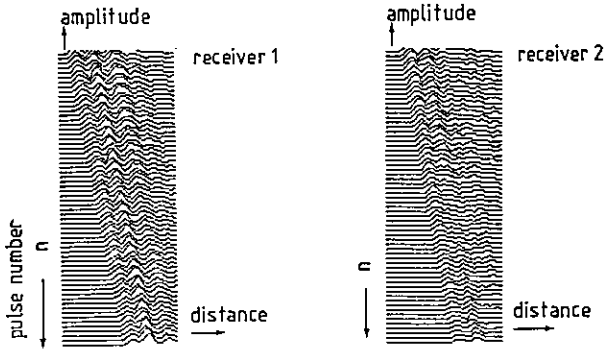


Fig 4. The cross section through the transmitter the two receivers and the screw. The amplitude is plotted versus distance. This is done for each transmitted pulse. Following the amplitude peaks corresponds to tracking method No 1. The motion is 320 mm and the bistatic angle goes from 27° to 16°.

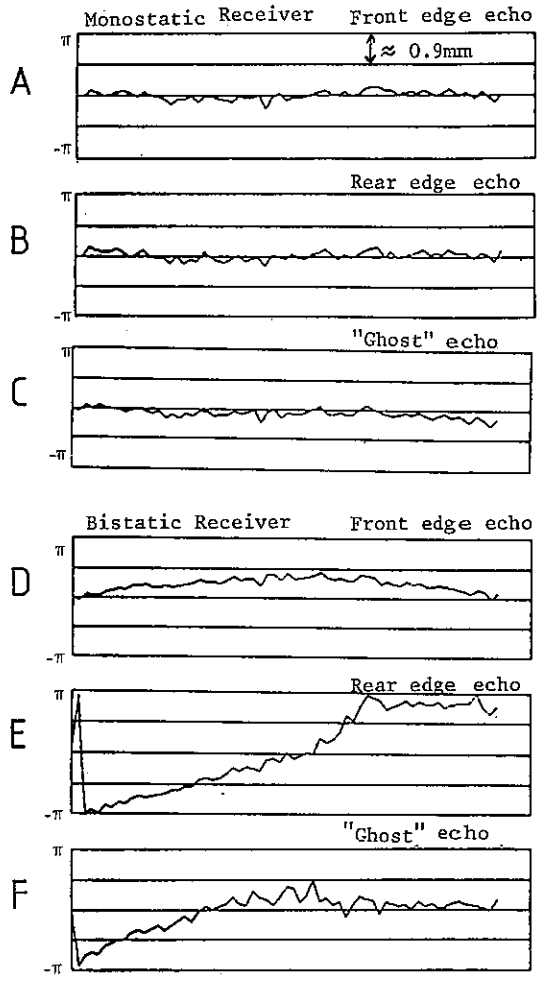


Fig 5. The phase residuals e(n) when tracking the box in Fig 3. Number of pulses = 64. Time gating is used to separate the three echoes in Fig 4. The motion parameter V (Eq. 5) was estimated for the echo from the front edge. Iterations using trial and error. Curve A shows the smallest e(n). Using the same V gives the residuals B and C. The monostatic ghost' has also the same displacement as the object.

If the three peaks in Fig 3 are modelled as point scatterers their bistatic phase trajectories obeys Eq 6. Curves D-F are the resulting phase residuals. The systematic errors are discussed in the text and in /5/. Three balls in a row can give rize to the result in Fig 3 but hardly to the phase trajectory in E. The local variations in A-D are fairly correlated indicating a common turbulence. The amplitudes in E and F was small - hence the noise. Notice that one transmitter and two receivers are not equivalent to two independent transmitter/ receiver pairs.

If all the R(n)'s are linear in n it can be shown that the optimal R'(n) is given by the discrete Fourier transform (DFT). In contrast with the DFT a constant velocity is not required for the phase unwrapping. However, without details, the DFT is most useful as a prefilter in Eq. 3 to reduce noise and to separate objects with different velocities.

5. TESTING THE TRACKING IN 2-D

The box depicted in Fig 3 is used to test the tracking method. The level curves shows an acoustical image of the box - duplicated from /2/. The image is obtained using turntable imaging (rotated 23^0 in steps of 1^0) and short pulses - compare /3/. The three "echos" are the front edge, the rear inside of the box and a ghost image generated by the wave going forth and back inside the box.

The test equipment in Fig 4 consist of one transmitter and two separate receivers used to record, simultaneously, the echo in the two different directions. The sensors and the motion are confined to a plane. The purpose of the tests is to track and compare the two-dimensional motion of the three above mentioned echos.

A model for the increments in the two propagation distances follows from Fig 4 as

$$R_1(n)-R_1(o)= 2nV \qquad (5)$$

$$R_2(n)-R_2(o)= nV- r_2(o) \qquad (6)$$
$$+SQR\left[(nV)^2 +2nV(r_1(o)-x_2)+r_2(o)^2\right]$$

where

$$r_2(o)= SQR\left[(r_1(o)- x_2)^2+(y_2)^2\right]$$

The vectorial motion of each echo (scatterer) is described by these equations. By using a sliding range interval, the echoes from different parts of the object are easy to separate. If the range interval consists of more than one range gate there is a coherent echo summation over the interval.

The test is made in two steps: First V is estimated using the echo from the front edge. This is done by iterating the parameter V in Eq 5 until the phase residual in curve A, Fig 5, is as small as possible. Phase residuals B and C shows that the echoes from the rear edge and from the "ghost" has moved the same distance.

The bistatic motion is modelled by Eq 6. All parameters are given above. The residuals were obtained according to curves D-F. Curve D shows a systematic error of approximately 0.7 mm. Some error sources are: walking reflection points as in Fig 2, the motion goes through the lobe starting just outside the far-zone, the speed of sound is necessary to relate V and x_2,y_2 in Eq 6. The phase center at x_2,y_2 is also difficult to measure. The phase residuals in E and F are typical when tracking multiple scatterers using a model that is too simple - Eq 6 "does not include ghosts". Other experimental tests and a comparison of the tracking methods listed in section 3 can be found in /5/.

6. SUMMARY - APPLICATIONS

Generically, the 'optimal' signal processing is simpel if there is only one dominant scatterer in each range gate. The estimated parameters has 'always' predictable properties.

The models and the tests described above are the introductury examples of manuscripts in preparation. Below we list some robotics applications where pulsed coherent ultrasonics can be a competitive sensor. Recall that this method is complementary to vision.

- Measuring velocity and displacement in 1,2 or 3 dimensions, say, when tracking parts on a tooled conveyor belt. Compensation for object shape is usually necessary.
- Measuring distance variations and angle to a "flat" surface.
- Measuring the shape of "locally convex" surfaces.
- Tracking linear structures like seams under water tubes etc.
- Monitoring a gripper during assembly using learned signatures.
- Object signatures using the distance (modulo $\lambda/2$) between different (convex) scatterers.

REFERENCES

1. Deschamps G A; Ray Techniques in Electromagnetics, Proc. IEEE, Vol 60 No 9, sep 1972
2. Jonsson T; Pulsed Acoustical Image Generation:Multiple Scattering - Imaging Defects, FOA 1985 Rep. D30385
3. Ausherman D A et al; Developments in Radar Imaging, IEEE Trans. AES-20, No 4 July 1984
4. Salomonsson B; Coherent Tracking Using Pulse Doppler Sodar- A Prestudy, FOA 1985 Rep. D30386
5. Coherent Tracking Using Pulse Doppler Sodar, FOA 1986 Rep. D30418-E
6. Viterbi; Principles of Coherent Communication, McGraw Hill, 1966.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

BEAMFORMING A PLANAR NON-LINEAR ARRAY BY ESTIMATING THE SENSOR POSITIONS

Michel BOUVET

Laboratoire des Signaux et Systèmes
CNRS-ESE-UPS *, Plateau du Moulon , 91190 GIF SUR YVETTE (FRANCE)
and
Groupe d'Etude et de Recherche en Détection Sous-Marine
LE BRUSC, 83140 SIX FOURS LES PLAGES, (FRANCE)

ABSTRACT

Array processing based on a linear assumption implies degradations in the source bearing estimation when the array is not linear, which is the usual case in real-world problems. Assuming only that the array is planar (but not necessarily quasi-linear), we describe a method to include in an adaptive way *information* about the sensor locations into the array processing in order to take into account its geometrical shape. This *information* is the relative positions of the sensors. Simulation results in a real underwater noise sample show that it leads to improvement with respect to the classical array processing.

1. INTRODUCTION

Beamforming an array of sensors is a classical and well-known mean for range and bearing estimation of targets in fields such as underwater acoustical signal processing [1-5]. Unfortunately, the array does not stay linear during its motion because of all the mechanical forces on this array. In recent years, some work has been done on calculating error bounds, Cramer-Rao bounds for example, on the parameters estimates, range and/or bearing, when there are uncertainties on the sensor positions [6,7].

Most of the existing works made the assumption of small displacement, or more precisely of a linear shape slightly perturbed. Unfortunately, this assumption is not true in real situations. So, it seems important to derive a method, if possible adaptive, able to deal with a "significantly nonlinear array". This paper addresses this problem: beamforming a towed array with unknown sensors locations, the array shape being possibly far from linear, in a sense that we will defined later. Roughly speaking, there are two different "subproblems": beamforming with a non-linear array; estimating in an adaptive way the sensor locations from the observation. No assumption will be made on the array shape model (polynomial, sinusoidal, travelling wave model ...). The only restrictive assumption we will make is that the array stays planar.

This work has been supported in part by the Direction des Constructions Navales under contract # 844882626100.

* Centre National de la Recherche Scientifique - Ecole Supérieure d'Electricité - Université de Paris-Sud

2. BEAMFORMING A PLANAR TOWED ARRAY

Figure I shows a particular array shape. It is clear that Δx can be rather important, say $0.2\,L$ for example, with L being the array length. Δz is usually small. In [8], Δz is roughly 1% of L, $\Delta z = -1.5$ feet for $L = 150$ feet. Another important remark is that for bearing estimation, the influence of Δz is much less important than the one of Δx. Hence, we will assume that we will work on the plane (Oxy). The array will be assumed planar, i.e., the sensors are coplanar. The problem will be a two-dimensional one.

2.1. Local Geometry: Notations

Figure II represents a portion of the array (5 sensors) and defines the notations that we will use. (Oxy) is a reference frame. θ is the bearing of a far-field source in this reference frame. The η_i correspond to the angles between the array and the reference frame (Oxy) whereas the ξ_i correspond to the angles between the array and the wave front coming from the source at a bearing θ. It is easy to see that we have

$$\xi_1 = \theta - \eta_1 \,, \tag{2.1}$$

$$\xi_i = \xi_{i-1} - \eta_i \,, \quad i = 2, \cdots, N \,, \tag{2.2}$$

where N is the number of sensors. The τ_i are the time delays for the wave front to go from sensor $\#i$ to sensor $\#i+1$,

$$\tau_i = \frac{l}{c} \sin(\xi_i) \,, \tag{2.3}$$

where l is the intersensor length and c the sound velocity. All the parameters indexed i correspond to the values between the i-th sensor and the $i+1$-th one. All the previous parameters can, of course, be positive or negative. For example, in Figure I, we have

$$\begin{array}{ccc} & \theta < 0 & \\ \eta_1 < 0 & \xi_1 > 0 & \tau_1 > 0\,, \\ \eta_2 < 0 & \xi_2 > 0 & \tau_2 > 0\,, \\ \eta_3 > 0 & \xi_3 < 0 & \tau_3 < 0\,, \\ \eta_4 < 0 & \xi_4 > 0 & \tau_4 > 0\,. \end{array}$$

We have chosen these angles as parameters because one can easily imagine a mechanical device providing knowledge of the angles ξ_i.

At time t, the observation is taken as a vector

$$\mathbf{X}(t) = \left[\, x_1(t),\, x_2(t),\, \cdots,\, x_N(t) \,\right]^t , \qquad (2.4)$$

$x_i(t)$ being the output at t of the i-th sensor.

2.2. Beamforming

The beamforming methods are all based on the cross-spectral matrix $G(f)$ [5],

$$G(f) \triangleq E\left\{ \mathbf{X}(f)\,\mathbf{X}^{\dagger}(f) \right\}, \qquad (2.5)$$

where $\mathbf{X}(f)$ is obtained from $\mathbf{X}(t)$ by discrete Fourier transform (DFT), and on $\mathbf{D}(\theta)$, the steering vector,

$$\mathbf{D}(\theta) \triangleq \left[1,\, e^{2\pi j T_2(\theta,\mathbf{P})},\, \cdots,\, e^{2\pi j T_N(\theta,\mathbf{P})}\right]^t , \qquad (2.6)$$

where $T_n(\theta,\mathbf{P})$ is the total delay between sensor #1 and sensor #n. In this expression, $\mathbf{P}$ is the set of parameters defining the array shape, for example,

$$\mathbf{P} \triangleq \left[\, \xi_1,\, \xi_2,\, \cdots,\, \xi_{N-1} \,\right]^t . \qquad (2.7)$$

It is easy to see that we have

$$x_n(f) = e^{-2\pi j f T_n}\, s(f) + v_n(f)\,, \qquad (2.8)$$

$s(f)$ being the signal received on sensor #1 and $v_n(f)$ being the noise received on sensor #n. From (2.3) and Figure II, we have the delays with respect to the sensor #1,

$$T_n(\theta,\mathbf{P}) = \sum_{i=1}^{n-1} \tau_i = \frac{l}{c} \sum_{i=1}^{n-1} \sin(\xi_i)\,. \qquad (2.9)$$

By putting

$$\phi_i \triangleq \theta - \sum_{j=1}^{i-1} \eta_j\,, \qquad (2.10)$$

we have

$$T_n(\theta,\mathbf{P}) = \frac{l}{c} \sum_{i=1}^{n-1} \sin(\theta - \phi_i)\,. \qquad (2.11)$$

Then, if one knows the angles, say ϕ_i, for example by using some mechanical device, beamforming is done by using ϕ_i in (2.11) and then including $T_n(\theta,\mathbf{P})$ in the steering vector (2.6). If it is not possible to include such a device in the array, perhaps because one wants to use an existing array, an alternative is to get an estimate of the parameter vector $\mathbf{P}$ with the help of a cooperative source at a known bearing.

3. ESTIMATION OF THE SENSORS POSITIONS

Let α be the *known* bearing of the cooperative source. We can rewrite (2.8) with a known signal, say $S(f)$ coming from the direction α,

$$x_i(f) = e^{-2\pi j f \frac{l}{c}\beta_i}\, S(f) + v_i(f)\,, \qquad (3.1)$$

where

$$\beta_i \triangleq \sum_{n=1}^{i-1} \sin(\alpha - \sum_{k=1}^{n-1} \eta_k)\,. \qquad (3.2)$$

3.1. Estimation of the relative angles

If there is no noise, $v_i(f) = 0$, it is easy to see that

$$-\frac{1}{a} Arg\,(x_i(f)S^*(f)) = \beta_i = \sum_{j=1}^{i} \sin(\alpha - \sum_{k=1}^{j-1} \eta_k), \qquad (3.3)$$

where we have defined

$$a \triangleq 2\pi f \frac{l}{c}\,. \qquad (3.4)$$

So, we can obtain η_i,

$$\eta_i = \alpha - \sum_{k=1}^{i-1} \eta_k - \sin^{-1}(\beta_{i+1} - \beta_i)\,. \qquad (3.5)$$

Finally, we get

$$\eta_i = \alpha - \sum_{k=1}^{i-1} \eta_k - \qquad (3.6)$$

$$\sin^{-1}\left\{ \frac{1}{a} \left[Arg\,(x_{i+1}(f)S^*(f)) - Arg\,(x_i(f)S^*(f)) \right] \right\},$$

which can be rewritten in term of ϕ_i,

$$\phi_i = \alpha - \qquad (3.7)$$

$$\sin^{-1}\left\{ \frac{1}{a} \left[Arg\,(x_{i+1}(f)S^*(f)) - Arg\,(x_i(f)S^*(f)) \right] \right\},$$

If there is noise, and if this noise is assumed stationary, white and Gaussian, the previous relation gives the maximum likelihood estimate of β_i. So, we can use (3.6) to estimate η_i while replacing in the right hand side of (3.6), η_k by its estimate $\hat{\eta}_k$, $1 \leq k \leq i-1$, previously computed.

3.2. Statistical characteristics of $\hat{\phi}_n$

As an indicator of the performance of this estimate, we are going to calculate the mean and variance of the estimate of ϕ_i, given by (3.7). We put

$$\frac{v_i(f)}{S(f)} \triangleq \rho_i \; e^{j\gamma_i} \; . \qquad (3.8)$$

s the inverse of the SNR at the sensor #i. Assuming ; ρ_i is constant, $\rho_i = \rho$, and small with respect to , it can be shown that

$$\hat{\phi}_i \cong \phi_i + \frac{\rho}{a\cos(\phi_i - \alpha)}(\sin(\varsigma_{i+1}) - \sin(\varsigma_i)) \triangleq \phi_i + \mu_i \; , \qquad (3.9)$$

ere

$$\varsigma_i \triangleq \gamma_i + a\;\beta_i \qquad (3.10)$$

a random variable, white, uniformly distributed on 2π [because we have made the assumption that f) was white stationary Gaussian. It turns out that is a random variable with zero mean and variance

$$\frac{\rho^2}{a^2\cos^2(\phi_i - \alpha)} = \frac{\rho^2}{\pi^2\cos^2(\phi_i - \alpha)} \; . \qquad (3.11)$$

obtain the right hand side of (3.11), we have assumed t the intersensor length was the half wave length, i.e., $= \frac{\lambda}{2}$. Hence, $\hat{\phi}_i$ is unbiased with the same variance 1). The variance of $\hat{\phi}_i$ is reduced if

ρ decreases, which means that the SNR of the cooperative source increases;

λ decreases, i.e., it is more helpful to use low frequency;

$\phi_i - \alpha$ is close to 0, i.e., the cooperative source is located to the side of the array.

s last remark is rather natural and means, first, that blems can arise if this cooperative source is along the of the array, and, second, that it is impossible to use reference the tow ship.

SIMULATION RESULTS: ARRAY SHAPED AN ARC OF CIRCLE

We have performed some simulations using a dised array. This array was composed of 64 sensors, larly distributed with a intersensor length of $\frac{\lambda}{2}$, λ g 7.5 m. It means that $f = 200\ Hz$ $l = 3.75\ m$ $L = 236.25\ m$ (total array length). As a deformamodel, we have arbitrarily chosen an arc of circle. a small angle η_i was taken equal to 0.01 rad. We used a real underwater noise [9].

The results appear in Figure III, the bearing of the erative source being arbitrarily fixed at 10^o. We plotted the beam pattern,

$$BP(\theta) = |\mathbf{X}^*(\theta)\,\mathbf{D}(\theta)|^2 \; , \qquad (4.1)$$

hree different steering vectors:

the one assuming a linear array shape, indicated by a dashed line on the figures;

(ii) the one using the true form of the steering vector (2.6), but with the estimated angles indicated by a dashed line;

(iii) the one using the perfect steering vector, with the true delays; this last one, indicated by a continuous line, serves as reference for the discussion.

The principal conclusion is that processing based on a linear assumption is poor, as expected. Processing using the estimated angles has very good performance and its beam pattern is very close to the one associated with the processing with perfect knowledge of the angles.

5. CONCLUSION

Since the shape of a towed array of sensors should not be assumed to be linear, the assumption on which classical array processing is based, one must get information on the sensor locations in order not to degrade the performance. This paper has presented an adaptive method to estimate the relative positions of the sensors. This method is based on the help of a cooperative source, emitting at a known bearing. Some simulation results show that the method that we have presented works well as long as the SNR of the cooperative source, the *reference* used to estimate the array shape, is high enough.

REFERENCES

[1] HINICH M.J., RULE W.: "Bearing estimation using a large towed array", *Journal of the Acoustical Society of America*, vol 58, no 5, November 1975, pp. 1023-1029

[2] GROSSI M.D., TACCONI G.: Special issue on Beamforming, *IEEE Journal of Oceanic Engineering*, vol 10, no 3, July 1985

[3] OWSLEY N.L.: "Sonar array processing", in *Array Signal Proceessing*, edited by S. Haykin, Prentice-Hall, 1985

[4] BIENVENU G., MERMOZ H.: "Principles of high-resolution array processing" in *V.L.S.I. and modern signal processing*, Kung S.Y., Whitehouse H.J. and Kailath T. (editors), Prentice Hall, pp. 83-105

[5] LUCAS B.G., LE CADRE J.P.: "Experimentation of spatial processing methods", *Proceedings of NATO ASI on Adaptive Methods in Underwater Acoustics*, H. Urban (editor), Luneburg (Germany), July 30-August 10 1985

[6] SCHULTHEISS P.M., IANNIELLO J.P.: "Optimum range and bearing estimation with randomly perturbated arrays", *Journal of the Acoustical Society of America*, vol 65, no 2, February 1979, pp. 528-531

[7] ASHOK E.: "Source location with arrays of imperfectly known configuration", *O.N.R. Report Yale University - Center for Systems Science*, June 1984

[8] LEE C.: "A modeling study on steady-state and transverse dynamic motion of a towed array system", *IEEE Journal on Oceanic Engineering*, Vol 3, no 1, January 1978, pp. 14-21

[9] BOUVET M., SCHWARTZ S.C.: "Signal detection and normalization in underwater noises modelled as a Gaussian-Gaussian mixture", Information Sciences and Systems Laboratory Report #18, Princeton University, January 1986

FIGURES

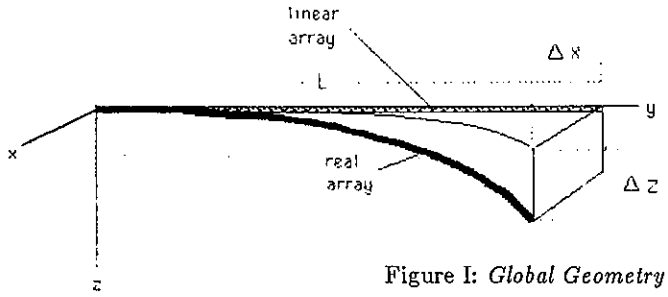


Figure I: *Global Geometry*

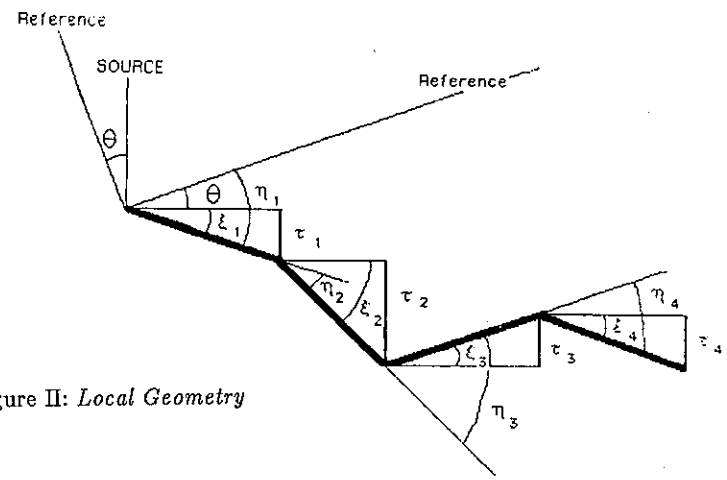


Figure II: *Local Geometry*

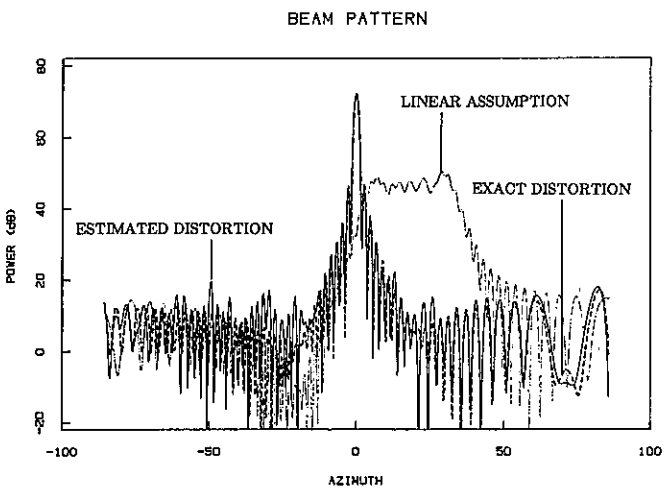


Figure III:
Beam Pattern : $\theta = 0°$, $SNR = 20\ dB$

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

P2.4

L_1-NORM ALGORITHM FOR SUPER-RESOLUTION IN TRACKING RADARS

G. Martinelli, P. Burrascano and G. Orlandi

INFO-COM Dpt.
University of Roma
via Eudossiana, 18
00184, ROMA - ITALY

Electronics and Automatics Dpt.
University of Ancona
via Brecce Bianche
60100, ANCONA - ITALY

A method is proposed for identifying the parameters of the signal received by a tracking radar under multipath environment. The model of the received signal consists of three sinusoidal signals equispaced in frequency and embedded in white noise. The proposed method is based on an L_1-norm minimization and achieves performances very close to Cramer-Rao bounds.

1. INTRODUCTION

A complete model of the signal received by a tracking radar in the case of a multipath situation is as follows

$$(1) \quad x(n) = A_1 e^{j\phi_1} e^{j(\omega_o+\Delta\omega)n} + A_2 e^{j\phi_2} e^{j\omega_o n} + A_3 e^{j\phi_3} e^{j(\omega_o-\Delta\omega)n} + b(n)$$

where the terms on the rigth-hand side of (1) are respectively the double-reflected echo, the reflected echo, the direct echo and the complex white noise with variance σ^2 (SNR=20logA_3/σ). The pulse repetition period (inverse of the PRF), is considered to be normalized to 1.
Moreover, if ρ is the magnitude of the surface reflection coefficient, we have

$$(2) \quad A_1 = \rho^2 A_3 \quad ; \quad A_2 = \rho A_3$$

The independent parameters of the model reduce consequently to ω_o, $\Delta\omega$, A_3, ρ and two phase differences, as for instance $\phi_3-\phi_1$, $\phi_2-\phi_1$.
The identification of these parameters is difficult, since the frequencies of the three tones present in (1) are very close. Namely, typical values of $\Delta f=\Delta\omega/2\pi$ are in the range: $10^{-5} \div 10^{-3}$. These very low values of Δf and the typical SNR ratios impose a large number of samples, for estimating the parameters of the model, as dictated by the Cramer-Rao bounds [1].
It is interesting is note that the model (1) is very general; important simplifications can be obtained if some of its parameters are known. An example of this remark is the method proposed in [2], where the phases ϕ_1, ϕ_2, ϕ_3 of the three tones are assumed to be zero.
In the present work we propose a method for estimating the model, based on the minimization of the L_1-norm [3].

The method is based on a parameter separation criterion which enables us to evaluate separately the different parameters of interest. This approach requires to operate on sequences derived from the original one by proper manipulations; model (1) is obviously changed as well, and the L_1-norm minimization operates in connection with sequences of the form

$$(3) \quad \zeta(n) = H_1 \sin\Delta\omega n + H_2 \sin 2\Delta\omega n + \varepsilon(n)$$

where H_1 and H_2 are related to the parameters of the model and $\varepsilon(n)$ to the noise. The sequences (3) are derived in two steps. In the first step we evaluate the square magnitude of the signal, i.e. $y(n)=|x(n)|^2$, which takes on the form

$$(4) \quad y_m(n) = B_o + B_1\cos\Delta\omega n + C_1\sin\Delta\omega n + B_2\cos 2\Delta\omega n + C_2\sin 2\Delta\omega n + \varepsilon_1(n)$$

where $\varepsilon_1(n)$ is related to the noise and B_o, B_1, C_1, B_2, C_2 are given by

$$(5.1) \quad B_o = A_1^2 + A_2^2 + A_3^2$$

$$(5.2) \quad B_2 = 2 A_1 A_3 \cos(\phi_3-\phi_1)$$

$$(5.3) \quad C_2 = -2 A_1 A_3 \sin(\phi_3-\phi_1)$$

$$(5.4) \quad B_1 = 2A \cos\phi$$

$$(5.5) \quad C_1 = -2A \sin\phi$$

with A and given by

This work was partly supported by SELENIA S.p.A. and partly by the Ministero Pubblica Istruzione.

$$(5.6)\quad A\,e^{j\phi} = A_2(A_1\,e^{j(\phi_2-\phi_1)} + A_3\,e^{j(\phi_3-\phi_2)})$$

The second step links (4) and (3) by combining two suitable samples of $y_m(n)$. Namely we consider the following combinations:

$$z(n) = \frac{1}{2}\,[y_m(n) - y_m(-n)]$$

$$n = 0,\ldots,1.5\,N$$

$$(6)\quad z_1(n) = \frac{1}{2}\,[y_m(n - \tfrac{1}{2}N) - y_m(-n - \tfrac{1}{2}N)]$$

$$z_2(n) = \frac{1}{2}\,[y_m(n + \tfrac{1}{2}N) - y_m(-n + \tfrac{1}{2}N)]$$

$$n = 0,\ldots,N$$

where we assume that the samples available for $x(n)$ are in the range $-1.5N \le n \le 1.5N$. The values of H_1 and H_2 of these sequences are listed in tab.1 together with the number M of samples available for them in terms of N. The necessity for considering more than one of these sequences is a consequence of the particular values taken on by the parameters of the received signal. In fact, in some cases one of them could have a very low level of energy and could be consequently unsatisfactory for the use in connection with the successive L_1-norm minimization. We will use at this regard the sequence (6) having the maximum energy.
We applied the L_1-norm minimization in place of the more popular L_2-norm criterion in consequence of very severe numerical problems connected to the latter, when applied to the matching of the model (4) to y(n). Details regarding this point are available in [4].

2. L_1-NORM MINIMIZATION

The basis of our method is the use of the L_1-norm in the spectral analysis of harmonic processes proposed in [3] and further improved in [5]. Namely, in [5] the procedure for estimating the quantities P_i, ω_i appearing in

$$(7)\quad r(m) = \sum_{i=1}^{N} P_i \cos\omega_i m + \delta(m)$$

where r(m) is the m-th sample of the autocorrelation of a process consisting of N tones embedded in white noise with $\delta(m)$ depending on the noise, requires the following steps

a) Model r(m) by

$$(8)\quad \xi(m) = \sum_{k=0}^{H} A_k \cos(\omega_o + k\Delta)m$$

where ω_o, Δ and H are chosen such that the known range of the ω_i in (7) is covered, the value of Δ is sufficiently small for guarantening a fine resolution

b) the estimation is carried out by solving the following linear programming problem

$$(9)\quad \min \sum_{k=0}^{H} A_k$$

$$\xi(m) = r(m) \qquad m = 1,\ldots 2N$$

$$A_k \ge 0$$

Table 1

Sequence	z(n)	$z_1(n)$	$z_2(n)$
H_1	C_1	$B_1\sin\Delta\omega\frac{N}{2} + C_1\cos\Delta\omega\frac{N}{2}$	$-B_1\sin\Delta\omega\frac{N}{2} + C_1\cos\Delta\omega\frac{N}{2}$
H_2	C_2	$B_2\sin\Delta\omega N + C_2\cos\Delta\omega N$	$C_2\cos\Delta\omega N - B_2\sin\Delta\omega N$
M	1.5 N	N	N

If the solution of the previous problem is A_{k_r}, r=1,2,...2N, then the resulting estimation is

$$(10)\quad P_i = (A_{k_{2i}} + A_{k_{2i-1}})$$

$$\omega_i = \omega_o + \frac{1}{2}(k_{2i} + k_{2i-1})\Delta \qquad i = 1,\ldots N$$

We have modified the procedure proposed in [5] in order to deal with the estimation of the parameters H_1 and H_2 of a sequence like (3). This case is different from that dealt with in [5], since H_1 and H_2 can be negative and the two tones are equispaced. In consequence of this difference, the linear programming problem becomes

a) Model $\zeta(n)$, n=1,...M by

$$(11)\quad \eta(n) = \sum_{k}' (H_{k1}\sin\omega_k n + H_{k2}\sin 2\omega_k n)$$

with the values ω_k chosen such that the known range of $\Delta\omega$ in (3) is covered and sufficient resolution is achieved;

b) the estimation is carried out by minimizing

$$(12)\quad \sum_{k}' (|H_{k_1}| + |H_{k_2}|)$$

under the constraints

$$\sum_k (H_{k_1} \langle \sin \omega_k n \rangle + H_{k_2} \langle \sin 2\omega_k n \rangle) = \langle \zeta(n) \rangle$$

$$(13) \quad \sum_k (H_{k_1} \langle\langle \sin \omega_k n \rangle\rangle + H_{k_2} \langle\langle \sin 2\omega_k n \rangle\rangle) = \langle\langle \zeta(n) \rangle\rangle$$

where the symbols $\langle \ldots \rangle$ and $\langle\langle \ldots \rangle\rangle$ stand for mean values computed in two suitable intervals. If M is the number of samples available for the sequence $\zeta(n)$, a convenient choice is the range $1 \div M$ for $\langle \ldots \rangle$ and $(\frac{M}{2} + 1) \div M$ for $\langle\langle \ldots \rangle\rangle$.
The choice of (12) as our objective function is directly related to the L_1 based spectrum analysis method above mentioned.
The solution of our problem could be achieved by using the simplex procedure. In this case, the formulation of the constraints (13) whould have needed some minor adjustment in order to impose that, also in the case of low SNR ratios, the couple $(H_{kr}; H_{jh})$ whose values the simplex choses as different from zero, is relative to two tones in the correct frequency ratio, i.e. $k=j, r \neq h$.
The L_1-norm minimization problem can, in this case, be highly simplified by considering the following remarks:

1) we are in the presence of only two constraints;
2) there are explicit formulas for the evaluation of $\langle \sin(h\omega_k n) \rangle$ and $\langle\langle \sin(h\omega_k n) \rangle\rangle$

On the basis of the above remarks we can solve system (13) for each value of ω_k in the range of interest; the value ω_k leading to the couple (H_{k_1}, H_{k_2}) minimizing (12) is the solution. This approach allows us to implicity impose that the two tones are in the correct frequency ratio.
In summary, the estimation procedure requires to compute, as a function of the variable x

$$H_1(x) = \frac{1}{\Delta}[\zeta_1 A(\tfrac{M}{2}, 2x) - \zeta_2 A(M, 2x)] \; ;$$

$$H_2(x) = \frac{1}{\Delta}[\zeta_2 A(M, x) - \zeta_1 A(\tfrac{M}{2}, x)]$$

$$\Delta = [A(M,x)A(\tfrac{M}{2}, 2x) - A(\tfrac{M}{2}, x)A(M, 2x)]$$

$$(14) \quad A(M,x) = \frac{\sin Mx/2}{M \sin x/2} \sin(M+1)x/2$$

$$\zeta_1 = \langle \zeta(n) \rangle \quad ; \quad \zeta_2 = \langle\langle \zeta(n) \rangle\rangle$$

The value x_{min} which minimizes

$$(15) \quad E(x) = |H_1(x)| + |H_2(x)|$$

is the solution. The estimated parameters are

$$(16) \quad \begin{aligned} \Delta\omega &= x_{min} \\ H_1 &= H_1(\Delta\omega) \\ H_2 &= H_2(\Delta\omega) \end{aligned}$$

Remark. It is important to note that the objective function (15) actually presents a behaviour versus x as shown in Fig.1. The value x_{min} is tipically equal to $0.85\,\Delta\omega$; moreover $E(x)$ tends to be less than $E(x_{min})$ for large values of x beyond the range of interest. Our method incorporates this remark by replacing $\Delta\omega = x_{min}$ for

$$(17) \quad \Delta\omega = x_{min}/0.85$$

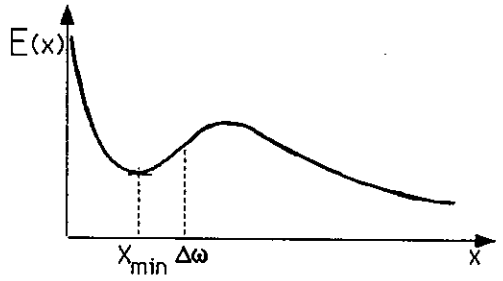


Fig.1. Behaviour of $E(x)$ versus x.

3. THE PROPOSED ALGORITHM

The previous L_1-norm minimization is applied to the sequence in (6) having the maximum energy. We obtain in this manner an estimation of $\Delta\omega$. The values of C_1, C_2 appearing in (5) are obtained as H_1 and H_2 from formula (14) by applying it in the case

$$(18) \quad \zeta(n) = z(n) \quad ; \quad M = 1.5\,N$$

and using in place of x the estimated $\Delta\omega$.
The values of B_1, B_2 are obtained from the same formula with

$$(19) \quad \zeta(n) = \frac{1}{2}[z_1(n) - z_2(n)] \quad ; \quad M = N$$

and using in place of x the estimated $\Delta\omega$. We obtain from the values H_1 and H_2 estimated in this manner

$$(20) \quad B_1 = H_1/\sin 0.5\Delta\omega\, N \quad ; \; B_2 = H_2/\sin \Delta\omega N$$

Finally, we obtain B_o as

$$(21) \quad B_o = \frac{1}{1.5N+1}\{\sum_{n=o}^{1.5\,N} \frac{y(n)+y(-n)}{2} +$$

$$- (B_1+B_2) - B_1 \frac{\sin\Delta\omega 3N/4}{\sin\Delta\omega/2} \cos(1.5N+1)\Delta\omega/2 +$$

$$- B_2 \frac{\sin\Delta\omega 3N/2}{\sin\Delta\omega} \cos(1.5N+1)\Delta\omega\}$$

From the values of B_o, B_1, B_2, C_1, C_2 we derive the parameters of the model as follows

1) the quantities A_1A_3 and $(\phi_3-\phi_1)$ are obtained from (5.2) and (5.3);

2) the quantity ρ is given by

$$\rho^2 = \sqrt{\frac{1}{4}\left(1-\frac{B_o}{A_1A_3}\right)^2-1}+\frac{1}{2}\left(1-\frac{B_o}{A_1A_3}\right) \tag{22}$$

3) we have

$$A_3 = \sqrt{\frac{A_1A_3}{\rho^2}} \quad ; \quad A_1 = \rho^2 A_3 \quad ; \quad A_2 = \rho A_3 \tag{23}$$

4) from B_1 and C_1 we obtain A and ϕ of (5.6). Then, we yield $(\phi_2-\phi_1)$ from

$$\cos[(\phi_3-\phi_1)-2(\phi_2-\phi_1)] = \tag{24}$$

$$= \frac{1}{2A_1A_3}\left\{\frac{A^2}{B_o-(A_1^2+A_3^2)}-(A_1^2+A_3^2)\right\}$$

by choosing the solution which satisfies also the equation

$$\rho^2 = \left[\frac{\sin[(\phi_3-\phi_1)-2(\phi_2-\phi_1)]}{\tan[\phi-(\phi_2-\phi_1)]}+\right. \tag{25}$$

$$\left.-\cos[(\phi_3-\phi_1)-2(\phi_2-\phi_1)]\right.$$

5) the last parameter ω_o is obtained by

$$\omega_o = \arg r_1 - \gamma\Delta\omega \tag{26}$$

where r_1 is the mean computed in the interval $-1.5N\div(1.5N-1)$ of $x(n+1)x^*(n)$, where the star denotes the complex conjugate; γ is given by

$$\gamma = \frac{A}{D} \tag{27}$$

where

$$A = \rho^4-1-\rho\{\rho^2\cos\alpha-\cos(\alpha-\beta)\}*$$

$$*\frac{\sin 1.5N\Delta\omega/2}{1.5N\sin\Delta\omega/2}\cos(1.5N+1)\Delta\omega/2$$

$$D = 1+\rho^2+\rho^4+2\rho\{\rho^2\cos\alpha-\cos(\alpha-\beta)\}*$$

$$*\frac{\sin 1.5N\Delta\omega/2}{1.5N\sin\Delta\omega/2}\cos(1.5N+1)\Delta\omega/2+$$

$$+2\rho^2\cos\beta\,\frac{\sin 1.5N\Delta\omega}{1.5N\sin\Delta\omega}\cos(1.5N+1)\Delta\omega$$

with

$$\alpha = \phi_2-\phi_1 \qquad \beta = \phi_3-\phi_1 \tag{28}$$

Remark. In computing ρ^2 we have disregarded in (22) the square root term when $(1-B_o/A_1A_3)^2<4$. Moreover, we have used the criterion $|\gamma|<1$ as a criterion on the validity of the estimate.

4. NUMERICAL EXAMPLE

The previous algorithm has been numerically tested in order to check its effectiveness and its performance with respect to Cramer-Rao bounds. In the following we describe one of the examples worked out.

The parameters of the model considered in the example are

$$A_3 = 1 \quad , \quad \rho = 0.8 \quad , \phi_1 = \phi_2 = \phi_3$$

$$f_o = \omega_o/2\pi = 0.1, \text{ SNR} = 30 \text{ dB} , \Delta f = 10^{-3}$$

In relation to this example we have experimented a thereshold, as a function of the number of samples, in correspondence to 3N+1 400.

In the following we report the results of the experiments concerning different noise sequences and with 3N+1=421. The details of the results are summarized in tab.2. In all the cases we found $|\gamma|<1$. The statistic of the estimate of f_o is

$$\text{mean} = 0.09996$$
$$\text{stand. dev.} = 1.3\ 10^{-4}$$

This result is very satisfactory, taking account that the Cramer-Rao bound on f_o gives a standard deviation equal to 10^{-4}.

Table.2. Results of the estimate of the parameters

Parameter	Estimate 1	2	3	4	5	6	7	8	9	10
f_o (10^{-4})	8.5	8.8	8.3	8.8	8.8	8.3	8.5	8.8	8.5	8.8
ρ	.82	.70	.97	.68	.69	.97	.81	.68	.81	.68
A_3	.96	1.1	.83	1.1	1.1	.82	.94	1.1	.96	1.1
$\phi_2-\phi_1$ (10^{-2})	-0.10	-0.45	-0.21	0.45	0.097	-1.3	0.19	1.3	-0.06	-0.76
$\phi_3-\phi_1$ (10^{-2})	-0.21	-0.90	2.8	-0.90	-0.19	-2.6	0.39	1.3	-0.13	-1.5
f_o	0.0998	0.100006	0.09974	0.100092	0.100089	0.09975	0.10008	0.09988	0.099895	0.100084

REFERENCES

[1] Carletti, U., G. Picardi and G. Spina, Elaborazione adattativa coerente di segnali radar, in: INFO-COM Dpt. Tech. Rpt n.003-5-84, Eng.ing Faculty of Roma, Italy.

[2] Orlandi, G., G. Martinelli and P.Burrascano, Explicit formulas for super-resolution, in: Proc. Int. Conf. on ASSP, Tampa (Fl., USA), 1985, pp.1356-1359.

[3] Levy, S., C.Walker, T.J.Ulrych and P.K. Fullagar, A linear programming approach to the estimation of the power of harmonic processes, in: IEEE Trans. on ASSP, vol.30, Aug. 1982, pp.675-679.

[4] De Felicis, G., Metodi di super-risoluzione nei radar ad inseguimento, Eng.ing thesis, Eng.ing Faculty of Roma, Italy, 1985.

[5] Figueiras-Vidal, A.R. and R.Garcia-Gomez, An improved approach to harmonic spectra estimation by linear programming, 1983 L´Aquila Workshop on Digital Signal Processing, 1983, L´Aquila (Italy).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DIRECT MLE APPROACH TO SOLVE MULTIPATH PROBLEMS IN TRACKING RADAR*

G. Picardi, R. Seu

INFO-COM Dpt., University of Rome
via Eudossiana, 18
00184, ROMA - ITALY

In this paper the sinewave selection problem in white noise environment for tracking radar applications has been analysed. The frequency shift between the direct and reflected echoes (by using specular reflection model) is very small and superresolution techniques must be applied by modeling the input process. Maximum Likelihood Estimator (MLE) approach is considered to solve the sinewave parameter estimation and an architecture of a linearized MLE is proposed. Therefore acquisition algorithms and sensitivities are analysed. The simulation results are compared with the theoretical Cramer-Rao bounds [1].

1. INTRODUCTION

It is known that the multipath [2] as interaction between direct and reflected echoes from the ground or sea surface produces the target fading and pointing errors. The importance of this problem depends on target and radar elevation angle, the distance target-radar, the transmission frequency, the polarization, the surface roughness, etc. In [1] the importance of specular reflection was evaluated: the diffuse reflection can be easily reduced by a proper frequency agility; the specular reflection (in main lobe region) which is due to smooth reflecting surfaces with low elevation target, causes important pointing problems hard to eliminate. The direct and reflected signals by considering conventional tracking radar could be separated on the basis of doppler difference, but differences are very poor so the superresolution techniques must be applied.

By comparing the frequency displacements due to doppler effect, the needed time with conventional resolution techniques and the target displacement (and consequently its doppler frequency) in the same time, it is easy to conclude that it is necessary to use the superresolution technique and the superresolution gain must be five or ten.

2. LINEARIZED MLE ARCHITECTURE

The analytical expression of the direct and reflected echoes is the following

$$(1)\quad z(t)=\underbrace{A_1 e^{j\omega(t-\frac{2R_1}{c})}}_{\text{direct echo}} + \underbrace{A_2 e^{j\omega(t-\frac{R_1+R_2}{c})}}_{\text{reflected echo}} + n(t)$$

where

$A_2=\rho A_1$ ρ is the reflection coefficient
$n(t)$ is the noise defined by signal to noise ratio
R_1 and R_2 are the range related to the direct and reflected paths
$\omega = 2\pi f_T$ and f_T is the transmitted frequency

The multipath problem can be reduced to the extraction of complex tones in white noise environment (s. eq.(1)).

By sampling eq.(1) (sampling rate is given by Pulse Repetition Frequency = PRF) and by considering discrete time observations (N+1 complex values) we can write:

$$(2)\qquad \underline{Z} = \underline{S} + \underline{W}$$

where

$\underline{Z}$ $(Z_{-N/2}, Z_{-N/2+1} \cdots Z_{N/2})$ is the return echo vector
$\underline{S}$ $(S_{-N/2}, S_{-N/2+1} \cdots S_{N/2})$ is the signal vector (direct and reflected echoes)
$\underline{W}$ $(W_{-N/2}, W_{-N/2+1} \cdots W_{N/2})$ is the noise vector

Each vector components are complex and the probability density function of $\underline{Z}$ given $\underline{S}$ is

*This work was partially supported by SELENIA S.p.A. (C.N. 50900/C), Nov. 1985.

$$(3)\quad p(Z/S) = \frac{1}{(2\sigma^2)^N} \exp\left\{-\frac{1}{2\sigma^2}\underbrace{\sum_{k=-N/2}^{N/2}[(x_k-s_k)^2+(y_k-\check{s}_k)^2]}_{L}\right\}$$

where

$x_k + jy_k = Z_k$

$s_k + j\check{s}_k = S_k$

σ^2 is the gaussian noise variance.

The optimization entails to minimize the L function of eq.(3) where s_k is the estimated signal (in the following we will write $\hat{s}_k$) and the minimization procedure is given by

$$(4)\quad \begin{aligned}
\frac{\partial L}{\partial A_1} &= \frac{1}{N}\sum_k (x_k\cos\hat{\phi}_{1k}+y_k\operatorname{sen}\hat{\phi}_{1k}) + \\
&- \frac{1}{N}\hat{A}_2\sum_k\cos(\hat{\phi}_{1k}-\hat{\phi}_{2k}) - \hat{A}_1 \\
\frac{\partial L}{\partial A_2} &= \frac{1}{N}\sum_k (x_k\cos\hat{\phi}_{2k}+y_k\operatorname{sen}\hat{\phi}_{2k}) + \\
&- \frac{1}{N}\hat{A}_1\sum_k\cos(\hat{\phi}_{1k}-\hat{\phi}_{2k}) - \hat{A}_2 \\
\frac{\partial L}{\partial \omega_1} &= \frac{\hat{A}_1}{N}\sum_k (-kTx_k\operatorname{sen}\hat{\phi}_{1k} + kTy_k\cos\hat{\phi}_{1k}) + \\
&+ \frac{\hat{A}_2}{N}\sum_k (-kTx_k\operatorname{sen}\hat{\phi}_{2k} + kTy_k\cos\hat{\phi}_{2k}) \\
\frac{\partial L}{\partial \Delta\omega} &= -\frac{\hat{A}_1\hat{A}_2}{N}\sum_k (-kT\operatorname{sen}(\hat{\phi}_{1k}-\hat{\phi}_{2k})) + \\
&+ \frac{\hat{A}_2}{N}\sum_k (kTx_k\operatorname{sen}\hat{\phi}_{2k}-kTy_k\cos\hat{\phi}_{2k}) \\
\frac{\partial L}{\partial \theta_1} &= \frac{\hat{A}_1\hat{A}_2}{N}\sum_k \operatorname{sen}(\hat{\phi}_{1k}-\hat{\phi}_{2k}) + \\
&+ \frac{\hat{A}_1}{N}\sum_k (-x_k\operatorname{sen}\hat{\phi}_{1k} + y_k\cos\hat{\phi}_{1k}) \\
\frac{\partial L}{\partial \theta_2} &= -\frac{\hat{A}_1\hat{A}_2}{N}\sum_k \operatorname{sen}(\hat{\phi}_{1k}-\hat{\phi}_{2k}) + \\
&+ \frac{\hat{A}_2}{N}\sum_k (-x_k\operatorname{sen}\hat{\phi}_{2k}+y_k\cos\hat{\phi}_{2k})
\end{aligned}$$

where (by writing $\phi_{ik}=\omega_i kT+\theta_i$, $\hat{\phi}_{ik}=\hat{\omega}_i kT+\hat{\theta}_i$):

$$(5)\quad \begin{aligned} x_k &= A_1\cos\phi_{1k} + A_2\cos\phi_{2k} + W'_k \\ y_k &= A_1\operatorname{sen}\phi_{1k} + A_2\operatorname{sen}\phi_{2k} + W''_k \end{aligned}$$

are the samples of the input signal to be estimated and $\hat{A}_i$, $\hat{\phi}_{ik}$ are the corresponding estimated values.

The linearization of the trigonometric functions of eq.(4) by considering noise absent allows a linear system:

$$\begin{aligned}
\frac{\partial L}{\partial A_1} &= \Delta A_1 + \Delta A_2 H + A_2(\delta_1-\delta_2)\tilde{H}_1 + A_2\gamma_2\tilde{H} \\
\frac{\partial L}{\partial A_2} &= \Delta A_2 + \Delta A_1 H - A_1\delta_1\tilde{H}_1 - A_1\gamma_1\tilde{H} \\
\frac{\partial L}{\partial \omega_1} + \frac{\partial L}{\partial \Delta\omega} &= \hat{A}_1A_1\delta_1 Q + \hat{A}_1A_2(\delta_1-\delta_2)H_2 + \\
&+ \hat{A}_1A_2\gamma_2H_1 - \hat{A}_1\Delta A_2\tilde{H}_1
\end{aligned}$$

$$(6)\quad \begin{aligned}
\frac{\partial L}{\partial \Delta\omega} &= -\hat{A}_2A_1\delta_1H_2 - \hat{A}_2A_1\gamma_1H_1 + \\
&- \hat{A}_2\Delta A_1\tilde{H}_1 - \hat{A}_2A_2(\delta_1-\delta_2)Q \\
\frac{\partial L}{\partial \theta_1} &= \hat{A}_1A_1\gamma_1 + \hat{A}_1A_2(\delta_1-\delta_2)H_1 + \\
&+ \hat{A}_1A_2\gamma_2H - \hat{A}_1\Delta A_2\tilde{H} \\
\frac{\partial L}{\partial \theta_2} &= \hat{A}_2A_1\delta_1H_1 + \hat{A}_2A_1\gamma_1H + \hat{A}_2\Delta A_1\tilde{H} + \\
&+ \hat{A}_2A_2\gamma_2
\end{aligned}$$

where

$$H = \frac{1}{N}\sum_k\cos(\hat{\phi}_{1k}-\hat{\phi}_{2k});\quad \tilde{H} = \frac{1}{N}\sum_k\operatorname{sen}(\hat{\phi}_{1k}-\hat{\phi}_{2k})$$

$$(7)\quad H_1 = \frac{1}{N}\sum_k kT\cos(\hat{\phi}_{1k}-\hat{\phi}_{2k});\quad \tilde{H}_1 = \frac{1}{N}\sum_k kT\operatorname{sen}(\hat{\phi}_{1k}-\hat{\phi}_{2k})$$

$$H_2 = \frac{1}{N}\sum_k k^2T^2\cos(\hat{\phi}_{1k}-\hat{\phi}_{2k})$$

and the unknown terms are the following

$$(8)\quad \begin{aligned}
X_1 &= \Delta A_1 = A_1-\hat{A}_1;\quad X_2 = \Delta A_2 = A_2-\hat{A}_2 \\
X_3 &= A_1\delta_1 = (\omega_1-\hat{\omega}_1)/(\hat{A}_1+\Delta A_1) \\
X_4 &= A_2(\delta_1-\delta_2) = (\omega_2-\hat{\omega}_2)/(\hat{A}_2+\Delta A_2) \\
X_5 &= A_1\gamma_1 = (\theta_1-\hat{\theta}_1)/(\hat{A}_1+\Delta A_1) \\
X_6 &= A_2\gamma_2 = (\theta_2-\hat{\theta}_2)/(\hat{A}_2+\Delta A_2)
\end{aligned}$$

In conclusion we can evaluate directly the $\partial L/\partial$. [eq.(4)] by using the discrete time observations Z_k and the previous estimated values $\hat{A}_i$, $\hat{\phi}_{ik}$ (for i=1,2, k= - N/2,...N/2). The linear system entries [eq.(6)] can be also evaluated by the previous estimated parameters, in this way by solving the linear system with classical algo-

rithms (the Gauss-Jordan algorithm was used), we can evaluate the estimation errors and by feedback we can adjust the previous estimated parameters.

The evaluation can be repeated on the same input samples in order to obtain steady condition and consequently optimum estimation will be obtained.

Figure 1 shows the analytical evaluation procedure of the entries and the $\partial L/\partial$.

3. DESIGNATION, ACQUISITION AND TRACKING ALGORITHMS

In order to allow the tracking of the input parameters it is necessary to have a frequency or doppler designation. The input discrete-time observations must be analysed with a FFT algorithm to obtain this designation. The signal to noise ratio in practical applications can be 10÷30 dB so the sensitivity problems are absent.

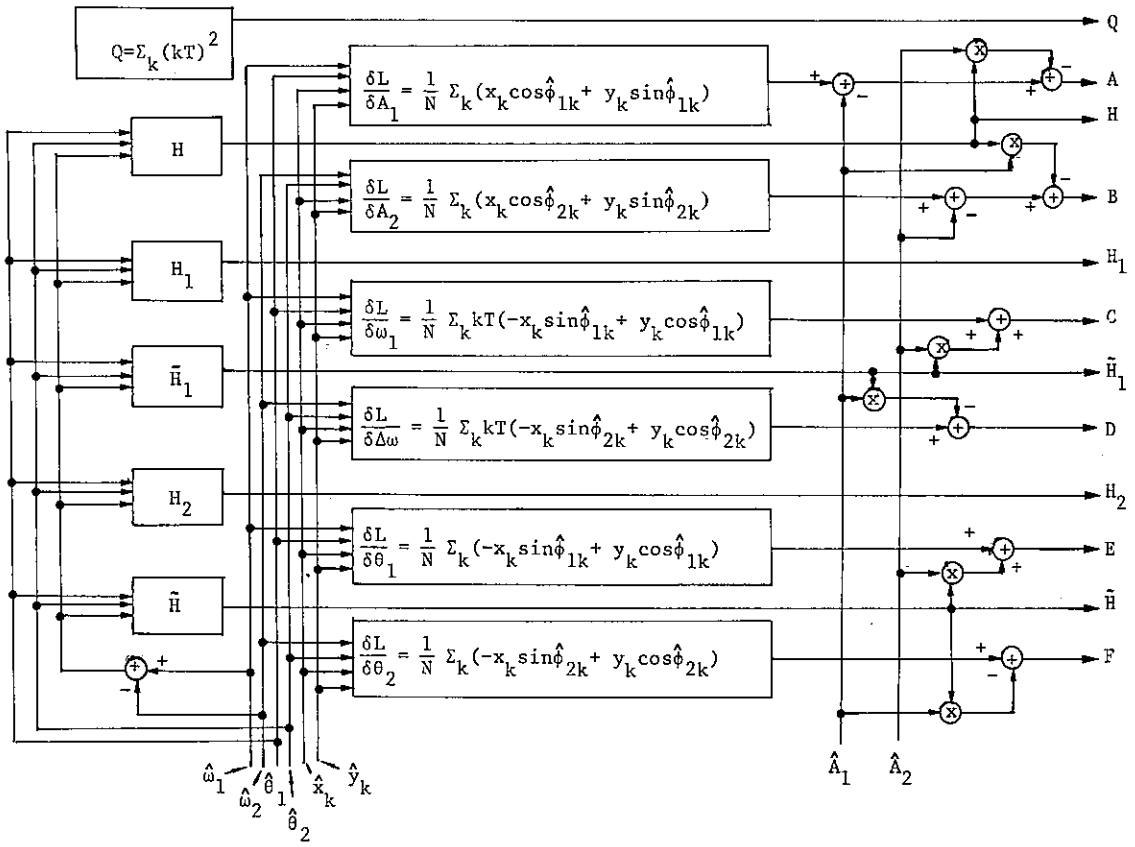


Figure 1

Matrix like form of the linearized system

$$\begin{pmatrix} 1 & H & 0 & \tilde{H}_1 & 0 & \tilde{H} \\ H & 1 & -\tilde{H}_1 & 0 & -\tilde{H} & 0 \\ 0 & -\tilde{H}_1 & Q & H_2 & 0 & H_1 \\ \tilde{H}_1 & 0 & H_2 & Q & H_1 & 0 \\ 0 & -\tilde{H} & 0 & H_1 & 1 & H \\ \tilde{H} & 0 & H_1 & 0 & H & 1 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \end{pmatrix} = (A \quad B \quad C \quad D \quad E \quad F)$$

The acquisition of the MLE proposed system, [eq.s (4), (6)], can be obtained by studying a proper algorithm which starts by a raw amplitude estimation and so the frequency estimation arises by the higher frequency of selected FFT range. Therefore the algorithm is given by:

- f_1 and f_2 selection (the f_1 and f_2 loops are closed). The frequency tracking loop is accomplished by a proper phases (θ_1 and θ_2) initial conditions. Figure 2 shows open loops f_1 discriminator characteristic
- θ_1 and θ_2 proper acquisition
- amplitude, acquisition
- finally together the overall loops are closed contemporary

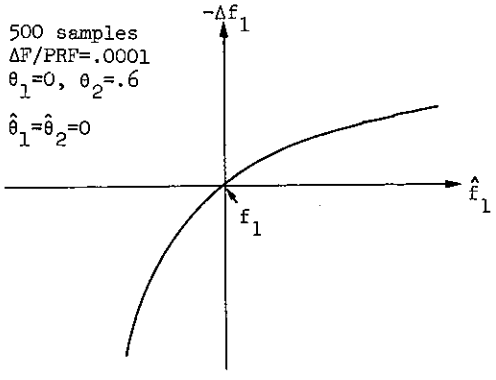
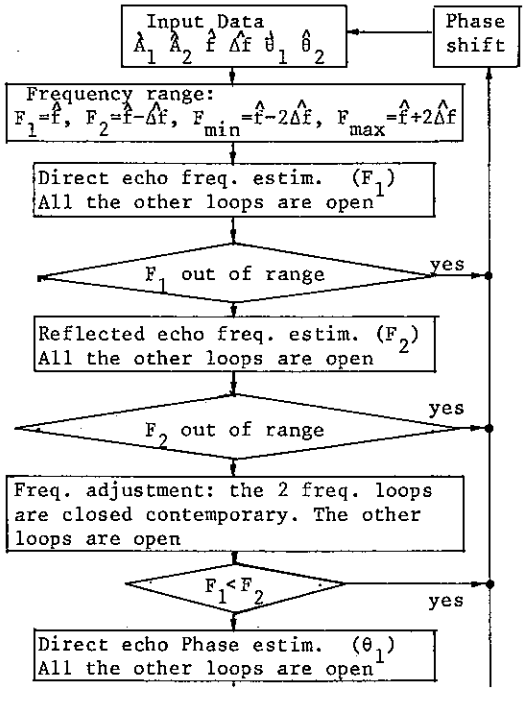


Figure 2

Therefore the Figure 3 shows a flow diagram of the acquisition algorithm.



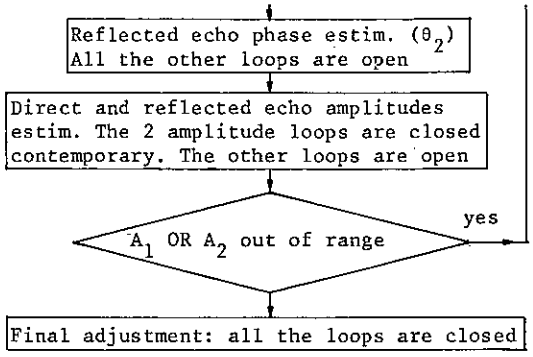


Figure 3

In tracking phase by designation of previous estimate the loops are only closed contemporary. The sensitivity results in comparison to the Cramer-Rao bounds are sketched in Figure 4.

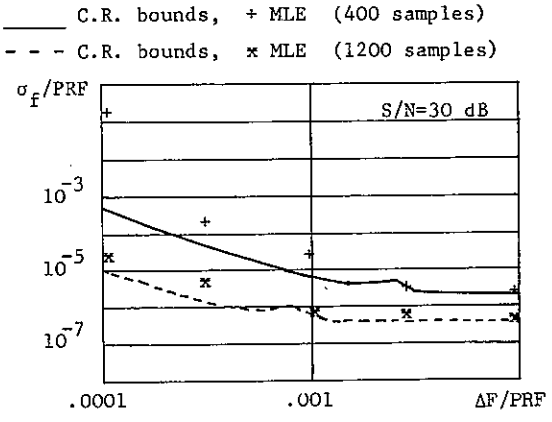


Figure 4

CONCLUSIONS

The MLE such as superresolution technique allows to distinguish two or more complex tones very close, with a limited discrete-time observations. In tracking radar systems in the presence of multipath, the reflected echo creates an error signal and, consequently, a pointing error. The frequency difference between the direct and reflected echo is very poor in this case. The MLE approach leads to the possibility of separating the two reflected and direct echoes. Now it is under investigation the angle tracking behaviour in multipath environment.

REFERENCES

[1] Picardi, G. and Spina, G., The ruperresolution applications in tracking radar, in IEEE Int. Radar Conf. Wash. 1985, pp.95-100.

[2] Barton, D.K., Low Angle Radar Tracking, in Proc. IEEE, Vol.62, No.6, June 1974, pp.687-704.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

RADAR CLUTTER SUPRESSION USING THE ADAPTIVE LEAST-SQUARES LATTICE ALGORITHM

K.K. PALIWAL and R. RAGHURAM

Computer Systems and Communications Group
Tata Institute of Fundamental Research
Homi Bhabha Road, Bombay 400005, India

The least-squares lattice (LSL) algorithm has been recently proposed for the design of an adaptive filter for the complex-valued data. In the present paper, this algorithm is compared as to its spectral estimation performance on complex-valued data with the gradient lattice (GL) and the least-mean-square (LMS) algorithms. It is found that the LSL algorithm performs better than the GL and LMS algorithms. Application of these algorithms for adaptive radar clutter supression is also studied here. Three different clutter conditions (ground clutter, average storm clutter and severe storm clutter) are investigated in the present study. It is found that the LSL algorithm is more effective in supressing radar clutter than the other two algorithms.

1. INTRODUCTION

Performance of the pulse Doppler radar systems in detecting targets (such as the moving aircraft) is seriously affected due to the presence of clutter. In an air-traffic control environment, this clutter arises due to the reflection from objects such as ground, trees, buildings, weather disturbances and moving flocks of birds. It is known that the clutter consists of a slowly varying stochastic process of unknown statistics. The conventional radar systems use an MTI (moving target indicator) filter to supress clutter. The MTI filter is a fixed high-pass filter and, hence, can not suppress effectively the clutter of varying statistics.

Recently, Haykin and his coworkers [1,2] have proposed an adaptive filter to supress the time-varying clutter. They have assumed the clutter to follow an autoregressive (AR) process and designed the adaptive filter using the gradient lattice (GL) algorithm which offers faster convergence than the least-mean-squares (LMS) algorithm. Recently, Hodgkiss and Presley [3] have developed a least-squares lattice (LSL) algorithm for the complex-valued signals. They have shown that this algorithm converges faster than the GL and LMS algorithms. In the present paper, we compare the spectral estimation performance of the LSL algorithm with that of the GL and LMS algorithms for the complex-valued data. We also study the application of the LSL algorithm for adaptive radar clutter supression and compare its performance with that of the GL and LMS algorithms. A description of these algorithms for the complex-valued data may be found in [3].

2. SPECTRUM ESTIMATION RESULTS

The LSL, GL and LMS algorithms have been compared earlier by Hodgkiss and Presley [3] using the output mean square error criterion. However, it has been shown by Honig and Messerschmitt [4] that a performance criterion based on the power spectral estimates is more relevant for comparison than the output mean square error criterion, since the former depends upon the accuracy of the estimated filter coefficients. We use here both of these criteria to measure the performance of these algorithms.

We have studied a number of AR processes for evaluating the performance of these algorithms, but for illustrating our results we select here a typical example of the AR process. In this example, the AR process is generated by passing a complex, zero-mean, white Gaussian noise sequence through a 4th order AR system. The parameters a_1, a_2, a_3 and a_4 of the AR system are (-1.040,-1.251), (-0.396,1.067), (0.364,0.185) and (0.091,-0.077), respectively. This AR system has been chosen so as to approximate the experimentally observed power spectral density of severe storm clutter. The adaptation constants for the three algorithms are chosen here so as to make the misadjustment errors equal.

Figure 1 shows the output mean square error (ensemble averaged over 200 different realizations) as a function of time (sample number) for the three algorithms. It can be seen from this figure that the LSL algorithm converges much faster than the GL and LMS algorithms.

Power spectral estimates at time instants t=10, t=45, t=100 and t=300 are shown in Fig. 2 for the three algorithms. It can be seen that the LSL algorithm outperforms the GL and LMS algorithms, especially at earlier instants. However, after t=100, there is little to choose between the LSL and GL algorithms.

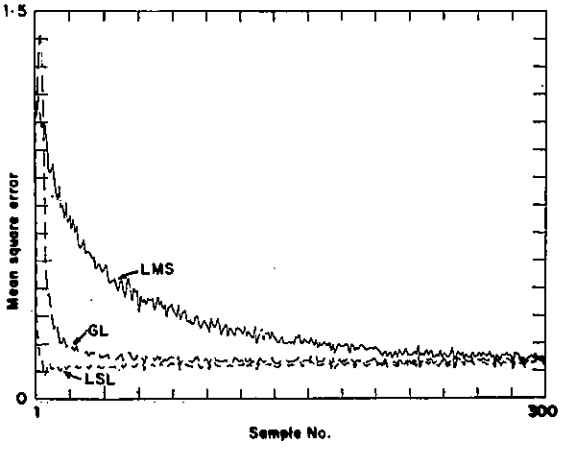


Fig. 1

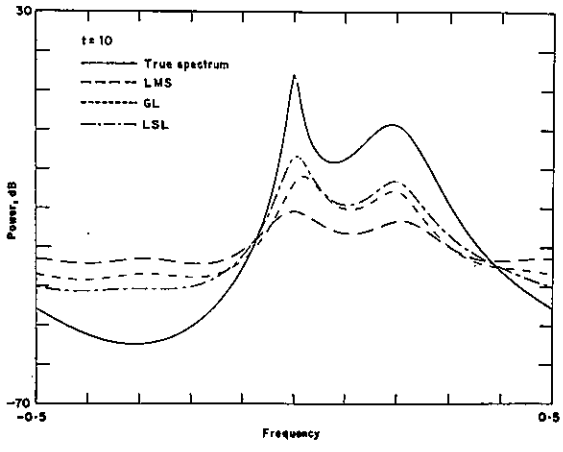


Fig. 2(a)

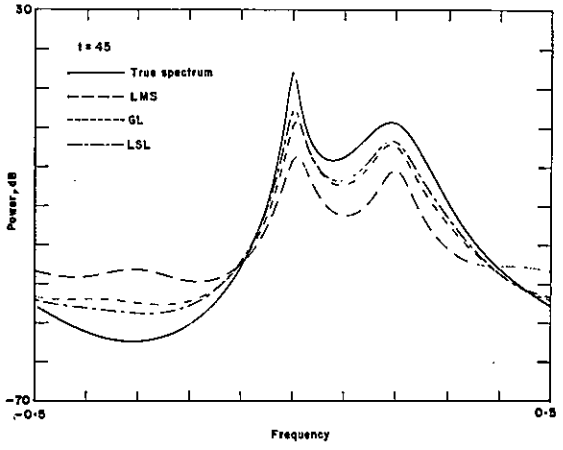


Fig. 2(b)

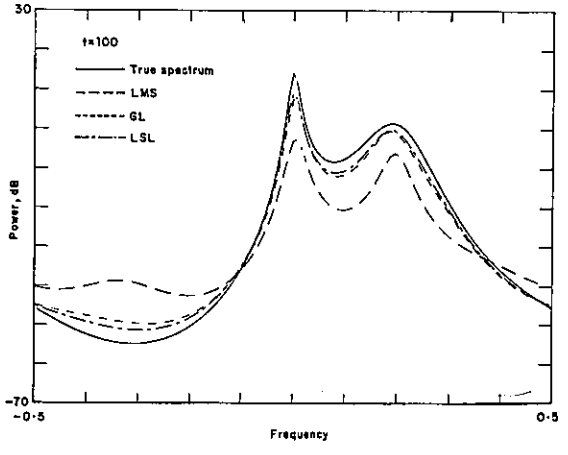


Fig. 2(c)

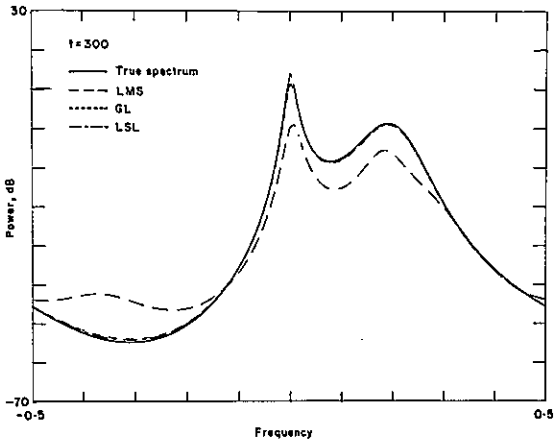


Fig. 2 (d)

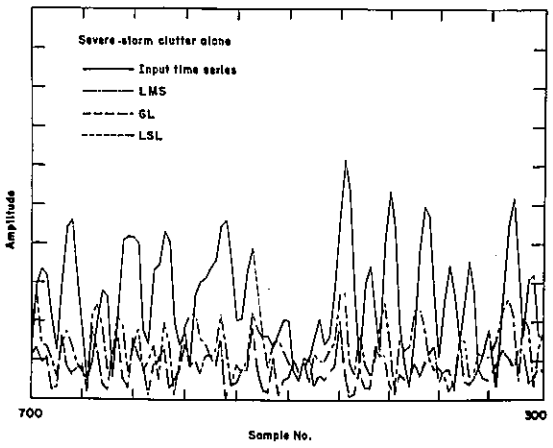


Fig. 3

3. RESULTS FOR CLUTTER SUPPRESSION

We study here the performance of the three algorithms for adaptive radar clutter supression under three different clutter conditions: 1) ground clutter, 2) average storm clutter and 3) severe storm clutter. Signal improvement factor (SIF) is used here as a criterion for performance evaluation. It is defined as the ratio of the target gain in the filter to the clutter gain in the filter, averaged over all frequencies of interest.

It is known [1] that the clutter signal can be modelled adequately as a complex-valued AR time-series. Accordingly, three such series were chosen to simulate the three clutter conditions. For the purpose of illustration, however, we show here the results for severe storm clutter only. The AR parameters for this clutter are already listed earlier. The target is represented here by a complex sine wave. The target duration is taken to be 10 hits embedded in a clutter record of 300 samples. The input signal to clutter ratio is 0 dB.

Initially, the clutter signal alone was used as input to the adaptive filter. Figure 3 shows the filter output along with the input for the case of severe storm clutter for the three algorithms. It can be seen that the LSL algorithm supresses clutter better than the GL and LMS algorithms. In Fig. 4, we show the results for the situation when the target signal has been added to the clutter. This figure shows that in enhancing target over clutter the LSL

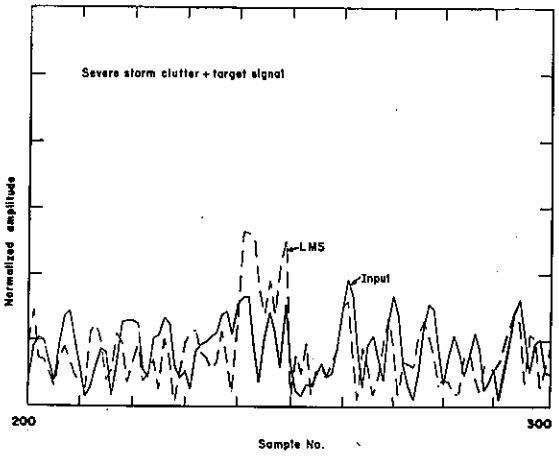


Fig. 4 (a)

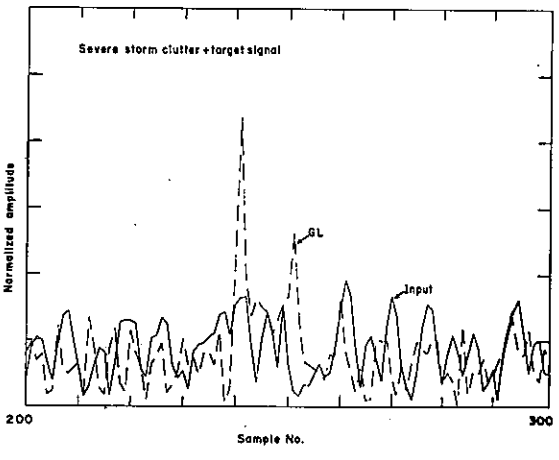


Fig. 4 (b)

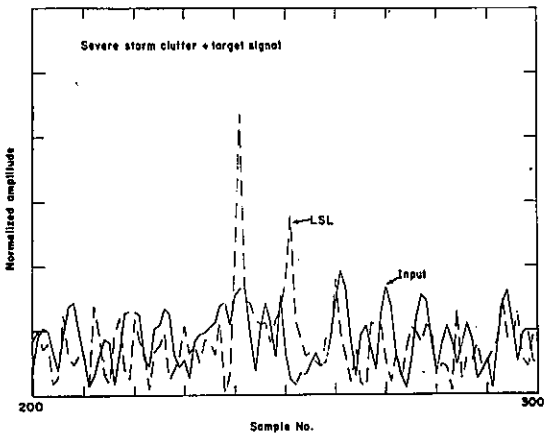


Fig. 4(c)

algorithm is more effective than the GL and LMS algorithms. This is quantitatively borne out by the SIF values which are listed in Table I for the three algorithms under the three different clutter conditions.

Table I. Signal improvement factor in dB for the LMS, GL and LSL algorithms under three different clutter conditions.

Clutter	Algorithm		
	LMS	GL	LSL
Ground	11.99	18.85	19.38
Average storm	9.16	14.10	18.69
Severe storm	7.48	8.19	13.60

4. CONCLUSIONS

In this paper, the LSL, GL and LMS algorithms are compared as to their spectral estimation performance. The LSL algorithm is shown to perform better than the other two algorithms. These algorithms are then applied to supress radar clutter under three different clutter conditions: ground clutter, average storm clutter and severe storm clutter. Signal improvement factor for the LSL algorithm is found to be higher than that for the GL and LMS algorithms.

REFERENCES

[1] S. Haykin, "Radar signal processing", IEEE ASSP Magazine, Apr. 1985, pp. 2-18.

[2] C. Gibson and S. Haykin, "Radar performance studies of adaptive lattice clutter-suppression filters", Proc. IEE, Vol. 130, Pt. F, 1983, pp. 357-367.

[3] W.S. Hodgkiss and J.A. Presley, Jr., "The complex adaptive least-squares lattice", IEEE Trans. ASSP-30, Apr. 1982, pp. 330-333.

[4] M.L. Honig and D.G. Messerschmitt, "Comparison of least-squares and stochastic gradient lattice predictor algorithms using two performance criteria", IEEE Trans. ASSP-32, Apr. 1984, pp. 441-444.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SEISMIC EXPLORATION IN THE SEARCH FOR OIL AND GAS, A REVIEW

A.J. Berkhout

Delft University of Technology, Dept. of Applied Physics,
P.O. Box 5046, 2600 GA Delft, The Netherlands

INTRODUCTION

In echo-acoustical applications the medium under investigation is 'illuminated' from the surface with acoustic waves. The incident wave field is reflected at inhomogeneities in the medium. The reflected wave field is registered at the surface and yields detailed information on the medium. Echo-acoustics is a fast growing field. Applications can be found in three main areas (fig.1):

1. Investigation of the earth's subsurface on land and at sea (seismic exploration)
2. Medical echo-diagnostics (ultra-sonic imaging)
3. Nondestructive inspection of the surface and the interior of materials ('NDT').

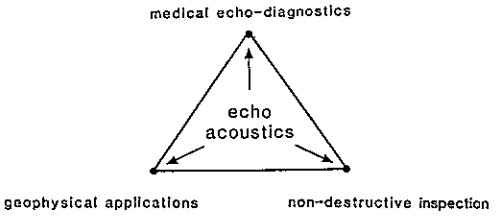


Fig. 1: Subdivision of echo-acoustical applications.

Although the required instrumentation in the above application is significantly different, a unified theory can be designed so that a good communication between experts from the different areas is very well possible.
In modern echo-techniques advanced data processing and information extraction play an essential role. Therefore, the most advanced digital technology can be found in this field. This is particularly true for seismic applications. In underwater inspection applications, acoustic sensors connected with intelligent digital hardware are state-of-the-art solutions. In the near future acoustic microscopes may play a very important role in the inspection of subsurface properties of micro-electronic devices.

In acoustical echo-techniques information about the mechanical properties of the interior is derived from the registration of the reflected wave field. For most practical problems it is necessary to illuminate the medium from different positions of the surface; registered wave fields are optimally combined during processing. The properties of the reflected waves are determined by two properties of the medium:

1. The **propagation** properties, which depend on the macro acoustic parameters of the medium such as 'average' velocity and 'average' absorption.

2. The **reflectivity** properties, which depend on the local acoustic parameters of the medium such as local variations in the modulus of elasticity and local variations in the density.

In many applications the macro parameters of the medium are well known but local variations are unknown and the objective of echo-techniques is to image in particular these local variations with the aid of reflected wave fields. The result should not only **localize** but also **characterize** the medium inhomogeneities.

Typically, the objective of seismic exploration in unknown areas is to obtain largely an estimate of the **macro** subsurface model by delineating the main reflecting boundaries and determining the **average** elastic parameters (preferably with their gradients) within each related geological layer. However, in field development projects the macro model is well known and the objective of the seismic method is to obtain reliable information on the detail of the elastic parameters within each main layer.

In fluids we have one wave type only, the longitudinal wave. But in media where significant shear forces can exist, also transverse waves occur. In addition surface waves can occur at boundaries. In medical echo-diagnostics only longitudinal waves play a role. In seismic echo-techniques also transverse and surface waves should be considered but, sofar, in almost all practical applications longitudinal waves are used for the actual extraction of information; the other wave types are suppressed by special measurement and processing techniques.
For the acoustic investigation of materials all wave types are used. Unlike in seismics, transverse waves can be easily generated by using special devices between source and material. Surface waves play an important role in the detection of surface cracks in construction materials.

In echo-acoustic techniques one has to deal with three basic problems:

1. Maximization of the signal-to-noise ratio
2. Optimization of the resolution along the vertical and lateral coordinates
3. Information extraction and presentation.

Noise is not only determined by statistical background signals but also by undesired wave types and multiple scattering. The axial resolution is given by the pulse length of each echo; the lateral resolution is determined by the number of different angles each inhomogeneity is illuminated with. In modern echo-systems pulse length and illumination properties are optimized for each point in the medium during data processing ('deconvolution' and 'synthetic aperture focussing'). Multichannel methods are essential for synthetic aperture techniques.

Finally, during the phase of information-extraction and presentation the acoustic results should be translated in terms of **geologic** information (seismics), **tissue** information (medical echo- diagnostics) or **mechanical properties and defects** (inspection). An important part of the informationextraction procedure may occur during visual evaluation of the processed

data ('interpretation'). In seismics, interpretive processes are now being developed which aim at an optimum interaction between multi data files, information-extraction algorithms and the seismic interpreter. Nowadays, these type of interactive procedures can be elegantly realized by means of workstations with excellent image-processing and display facilities.

Figure 2 gives a view on the applied frequencies in the different applications. Every frequency range is a **compromise** between resolution and penetration: by increasing frequencies the resolution increases but the penetration decreases due to increased absorption.

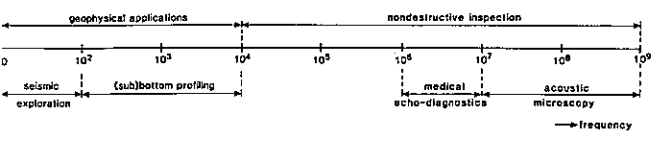


Fig. 2: Approximate frequency range as used in the different echo-acoustical techniques.

SEISMIC IMAGING

In seismic applications for oil and gas exploration, the subsurface is 'illuminated' from the surface with acoustic waves. The incident wave field is reflected at the geologic layer boundaries. The reflected wave field is registered at the surface and yields detailed information on the subsurface.

Over the years well over 90% of exploration geophysical expenditures in the oil industry have been committed to reflection seismic methods to provide information for drilling decisions. An important aspect of the seismic method is that an unprocessed seismic image does not represent an actual picture of the subsurface. Each reflection has been distorted during its propagation through the subsurface. In many situations these distortions are significant and have to be eliminated before an accurate picture of the subsurface can be developed. This is particularly accomplished by 'seismic inversion'.

Current and future seismic developments will be based on

- increasing role of wave theory in seismic processing
- use of available geologic information in seismic inversion ('model-driven inversion')
- improved computational systems for modeling and inversion, workstations for interpretive processing and presentations, microprocessor-based hardware for data acquisition.

In this paper a review of current seismic techniques and future developments is given with an emphasis on the to-day's key issue: **seismic inversion.** It is shown that important new developments in theory, software and hardware have made properly formulated wave theory solutions available which were unthinkable in the past.

Most conclusions are general and apply equally well to other echo techniques such as ultrasonic medical imaging, non-destructive testing, acoustic microscopy, sonar, ground radar.

REFERENCES

Berkhout, A.J., 1984a, Seismic Resolution -Resolving Power of Acoustical Echo Techniques-, Geophysical Press, London and Amsterdam

Berkhout, A.J., 1984b, Seismic Migration -Imaging of Acoustic energy by Wave Field Extrapolation-, vol. 14B, Practical Aspects, Elsevier, Amsterdam and New York.

Berkhout, A.J., 1985, Seismic Migration -Imaging of Acoustic Energy by Wave Field Extrapolation-, vol. 14A, Theoretical Aspects, third edition, Elsevier, Amsterdam and New York.

Claerbout, J.F., 1976, Fundamentals of Geophysical Data Processing, New York, McGraw Hill.

Robinson, E.A., 1982, Migration of geophysical data: IHRDC publishers.

Robinson, E.A., 1984, Seismic Inversion and Deconvolution, Geophysical Press, Amsterdam.

Schneider, W.A., 1978, Integral formulation for migration in two and three dimensions, Geophysics, vol. 23, p. 49 - 76.

Stolt, R.H., 1978, Migration by Fourier transform, Geophysics, vol. 23, p. 23 - 48.

Tarantola, A., Linearized Inversion of Seismic Reflection data, Geophysical Prospecting, vol. 32, no. 6, p. 998 - 1016.

Treitel, S. and Lines, L.R., 1982, Linear inverse theory and deconvolution, Geophysics, v. 47, p. 1153 - 1159.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ATTACKING THE ONE-DIMENSIONAL, TWO-CHANNEL INVERSE PROBLEM OF REFLECTION SEISMOLOGY

Ralf-Günter Ferber

Institut für Geophysik - Ruhr Universität Bochum
Bochum, Federal Republic of Germany

The solution of the one-dimensional, one-channel communication-theoretical inverse problem of reflection seismology is generalized in this paper to the solution of the one-dimensional but two-channel inverse problem

1. INTRODUCTION

To obtain the reflection coefficients of a horizontally plane layered earth from its normal-incidence plane wave response is the one-dimensional, one-channel inverse problem of reflection seismology. The petrological parameters of the earth model vary with depth only, therefore it is called an one-dimensional problem. The normal-incidence plane wave response has only a vertical compressional wave component, that is the reason why it is called an one-channel problem. In this case the earth model acts as an one-dimensional linear system with a rational transfer function. For earth models with equal traveltime layering the numerator polynomial of the transfer function is a good approximation to the z-transform of the reflection coefficients series, as long as no strong multiples occur.
This results in a transformation of the seismic inverse problem into a communication-theoretical one (Robinson and Treitel, 1977).
In a first step, the spectral function of the earth model, which represents the net downgoing energy in the first layer due to the normally incident plane wave, and which is known to be an autocovariance function, is computed from the impulse response (the reflection seismogram). The spectral function is algebraically inverse to the autocovariance function of the minimum delay denominator polynomial of the earth filter.

In the second step, the denominator polynomial is computed from the spectral function using the Levinson recursion.
As a third step, the denominator polynomial is used as an "inversion filter", because multiplication of the denominator polynomial with the z-transform of the impulse response equals the numerator polynomial (Ferber, 1985).
In conclusion the inversion of a normal-incidence seismogram only yields the acoustic impedance, given by multiplication of compressional wave velocity and density.
In reflection seismic surveys, the earth is probed by a compressional wave source which can be decomposed into a spectrum of plane waves with varying angles of incidence (Treitel et al., 1982). In case of non normal-incident plane waves the inverse problem is far more complicated due to mode conversion of compressional into shear waves (and vice versa) at the layer boundaries. On the other hand, one can extract far more information from the non normal-incidence reflection seismograms, because in this case one can, at least theoretically, resolve compressional and shear velocities as well as the density of the layers.
The result is an one-dimensional (the earth model is still varying with depth only), but two-channel inverse seismic problem. One channel represents the compressional wave response and the other the shear wave response of the earth model.

In this case the earth model can again be described as a linear system with a rational transfer function, but now with numerator and denominator polynomials containing (2x2)-matrix coefficients (Frasier, 197o).
In order to extent the normal-incidence solution of the one-dimensional seismic inverse problem to the two-channel (non normal-incidence) setting, at first the spectral function approach is generalized in this paper to solve the related communication-theoretical inverse problem. It is shown, that the generalized spectral function exists, that it is in analogy a crosscovariance function, and that it still can be factorized to compute the denominator polynomial. Again the numerator polynomial is computed by multiplication of the z-transform of the matrix reflection seismogram (the impulse response) and the denominator polynomial. This (2x2)-matrix seismogram is built up by the (1x2)-columns of the compressional (P) and shear (SV) wave components of the compressional and shear pulse source reflection seismograms.
In conclusion, using the generalized spectral function, the communication-theoretical one-dimensional but two-channel inverse problem is solved.
The relation to the non normal-incidence seismic inverse problem appears to be difficult due primarily to the unequal transit times for compressional and shear waves, and needs further research.

2. THE ONE-DIMENSIONAL, ONE-CHANNEL INVERSE PROBLEM

This section gives a brief introduction to the theoretical work done to solve the one-dimensional and one-channel inverse problem.
If a unit spike is used as a downgoing plane wave source located just below the top of the first layer, the normal-incidence spectral function, which is shown to be an autocovariance function, is given by (Robinson and Treitel, 1977)

$$\Phi(z)=1-r_o(X(z)-X(z^{-1}))+(r_o^2-1)X(z)X(z^{-1}) \qquad (1)$$

where X(z) is the z-transform of the reflected wave in the source plane.
Suppose that the reflection seismograms are processed in a way, that the influence of the surface is removed from the data. In this case we have for the surface reflection coefficient $r_o=o$ and the spectral function is reduced to

$$\Phi(z)=1-X(z)X(z^{-1}). \qquad (2)$$

The reflection seismogram is equal to the impulse response of a pole-zero filter

$$X(z)=B(z)A(z)^{-1},\ a_o=1 \qquad (3)$$

where A(z) holds the minimum delay property.
As a result we derive the following equation for the spectral function

$$\Phi(z)=A(z)^{-1}vA(z^{-1})^{-1} \qquad (4)$$

from which A(z) can be computed (due to its minimum delay property) via the Levinson recursion. By multiplication of A(z) and X(z) the numerator polynomial is computed by

$$B(z)=A(z)X(z). \qquad (5)$$

In other words, the related communication-theoretical inverse problem, i.e. the computation of A(z) and B(z) from X(z), is solved.
If strong multiples can be neglected, the numerator polynomial is a good approximation to the z-transform of the two-way traveltime reflection coefficients series and the reflection seismic inverse problem is solved as well.

3. THE ONE-DIMENSIONAL, TWO-CHANNEL INVERSE PROBLEM

The following conclusions can be drawn from a strict theoretical generalization of the solution to the inverse problem outlined in the preceding section. First, in addition to the vertical component of the ground motion, the in-line horizontal component has to be measured. Second, in addition to the measurements with a source radiating compressional waves, a second experiment with a source radiating vertically

polarized shear waves has to be done (Clarke, 1984).

In conclusion, instead of one seismogram, four seismograms have to be measured.

The following additional processing is necessary to produce the input data for one-dimensional but two-channel inversion algorithms.

First, each common shotpoint gather has to be decomposed into its spectrum of plane wave reflection seismograms, for example by using a slant stack procedure (Treitel et al., 1982).

Second, the vertical and horizontal components of the plane wave seismograms have to be rotated to give the compressional and shear wave components.

Let $\underline{W}_P(z)=(1,o)^T$ denote the compressional pulse source, $\underline{W}_S(z)=(o,1)^T$ the vertically polarized shear pulse source and $\underline{X}_P(z)$, $\underline{X}_S(z)$ the P-SV component vectors of the related plane wave reflection seismograms.

Let $\underline{\underline{X}}(z)=(\underline{X}_P(z),\underline{X}_S(z))$ denote the (2x2)-matrix reflection seismogram. The generalized (non normal incidence) spectral function, which is shown to be a crosscovariance function (Ferber, 1986) is given by

$$\underline{\underline{\Phi}}(z)=\underline{\underline{I}}_2+\underline{\underline{R}}_o\underline{\underline{X}}(z^{-1})+\underline{\underline{X}}(z)\underline{\underline{R}}_o+\underline{\underline{X}}(z)\underline{\underline{R}}_o^2\underline{\underline{X}}^T(z^{-1}) -\underline{\underline{X}}(z)\underline{\underline{X}}^T(z^{-1}). \quad (6)$$

Suppose again that the reflection seismograms are processed in a way, that the influence of the surface is removed from the data. In this case we have for the surface reflection matrix $\underline{\underline{R}}_o=\underline{\underline{o}}$ and the generalized spectral function is reduced to

$$\underline{\underline{\Phi}}(z)=\underline{\underline{I}}_2-\underline{\underline{X}}(z)\underline{\underline{X}}^T(z^{-1}) \quad (7)$$

Again, the (2x2)-matrix reflection seismogram is equal to the impulse response of a "pole-zero" filter (Frasier, 197o)

$$\underline{\underline{X}}(z)=\underline{\underline{B}}(z)\underline{\underline{A}}(z)^{-1}, \quad \underline{\underline{a}}_o=\underline{\underline{I}}_2 . \quad (8)$$

In analogy, we have for the generalized spectral function the identity

$$\underline{\underline{\Phi}}(z)=\underline{\underline{A}}(z)^{-1}\underline{\underline{V}}\underline{\underline{A}}^T(z^{-1})^{-1} \quad (9)$$

from which $\underline{\underline{A}}(z)$ can be computed using the the two-channel Levinson recursion.

Multiplication of $\underline{\underline{X}}(z)$ by $\underline{\underline{A}}(z)$ from the right

$$\underline{\underline{B}}(z)=\underline{\underline{X}}(z)\underline{\underline{A}}(z) \quad (1o)$$

finally solves the communication-theoretical inverse problem.

As it was pointed out in the introduction the relation of the solution to the communication-theoretical inverse problem to the seismic inverse problem in the two-channel case is still an open question.

4. CONCLUSIONS

The solution of the one-dimensional, one-channel communication-theoretical inverse problem of reflection seismology is generalized to the solution of the one-dimensionla but two-channel inverse problem. The relation to the reflection seismic inverse problem, which is approximately solved by the solution of the communication-theoretical inverse problem in the normal-incidence case, is still an open question and needs further research.

5. REFERENCES

Clarke, T.J.: The P-SV inverse problem for a layered medium: what data are required ?. Geophys.J.R.astr.Soc. 79, 1984, p.565-572

Ferber, R.-G.: Stabilization of normal-incidence seismogram inversion removing the noise induced bias. Geophys. Prosp. 33, 1985, p.212-223

Frasier, C.W.: Discrete time solution of plane P-SV waves in a plane layered medium. Geophysics 35, 197o, p.197-219

Robinson, E.A. and S. Treitel: The spectral function of a layered system and the determination of the waveforms at depth. Geophys. Prosp. 25, 1977, p.434-459

Treitel, S., P.R. Gutowski and D.E. Wagner: Plane-wave decomposition of seismograms. Geophysics 47, 1982, p.1375-14o1

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Q.3

A NOVEL APPROACH TO 3-D SEISMIC PROCESSING

C.P.A. Wapenaar and A.J. Berkhout

Delft University of Technology, Dept. of Applied Physics
P.O. Box 5046, 2600 GA Delft,
The Netherlands

INTRODUCTION

Exploration seismology is based on analysis of seismic waves reflected from different layers in the earth's subsurface. Seismic energy, radiated by a seismic source into the subsurface, encounters discontinuities between the layers and is partially reflected back to the surface. The returning reflections, which contain indirect information on the elastic parameters of the subsurface, are detected and stored on magnetic tapes. Generally many seismic experiments are carried out for different positions of the seismic souce. The aim of seismic **migration** (a multi-dimensional seismic inversion technique) is to resolve a structural image of the subsurface from seismic measurements.

2-D VERSUS 3-D MIGRATION

In many practical situations seismic data acquisition is carried out along a straight line. Subsequently, seismic migration is carried out only for a vertical cross-section of the earth's subsurface below the acquisition line. In this case seismic inversion becomes a two-dimensional (2-D) technique. This may be very attractive from a computational point of view, however, the earth is three-dimensional (3-D). Particularly in areas with complicated structures 2-D migration techniques give a poor image of the subsurface. Actually, 2-D migration techniques may only be used when the subsurface model approximately satisfies the 2-D assumption, that is, when the elastic parameters depend on two spatial coordinates only. Unfortunately this assumption is rarely met in practice and therefore the reliability of many 2-D migration results is questionable.

POST-STACK VERSUS PRE-STACK MIGRATION

In the seventies and the early eighties much effort has been spent in the development of both 3-D data acquisition and 3-D inversion techniques. A typical 3-D marine survey is visualized in Figure 1.
Up to the present day, however, only the 3-D extension of the conventional migration approach has got serious attention, because conventional
migration is efficiently carried out after a so-called stacking process (post-stack migration), that is, after data reduction.

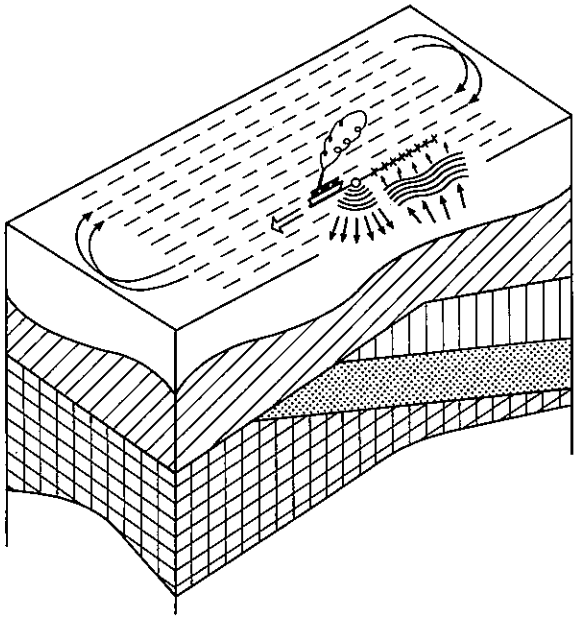


Figure 1: In 3-D marine seismics, data are often gathered along straight lines. Along each line many seismic experiments are carried out for different positions of the seismic source. In each seismic experiment many seismic signals are registered by the seismic detectors.

For a geologically complex subsurface, however, the resolution of 3-D post-stack migration techniques is far from optimum. A much better resolution may be expected from 3-D migration before stack (pre-stack migration), that is, before data reduction. However, even with nowadays fast vector computers full 3-D pre- stack migration is still unthinkable because of the enormous amount of data to be processed. For instance, a typical 3-D marine survey (see also Figure 1) consists of

200 seismic lines,
200 seismic experiments per seismic line,
100 traces per seismic experiment,
2000 samples per trace,
4 bytes per sample,

hence, the total survey contains 32 Gbyte of data. It is obvious that, given the limitations of computer hardware, a more practical approach to 3-D pre-stack migration is required.

TARGET ORIENTED 3-D PRE-STACK MIGRATION

In many practical situations seismic interpreters are mainly interested in a high resolution image of a pre-specified target zone. Hence, much work can be saved by the following two-stage procedure:

1. Apply conventional post-stack migration for an initial 3-D evaluation of the subsurface.
2. Apply full 3-D pre-stack migration to specific areas of interest ("target-oriented" stage).

During the presentation the details of target oriented 3-D processing will be discussed. Particularly the practical aspects of data-management will be emphasized. It will be shown that high quality 3-D images of the 'target zone' can be obtained in realistic processing times.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

KNOWLEDGE-BASED IMAGE PROCESSING FOR GEOPHYSICAL INTERPRETATION

I.Pitas
Department of Electrical Engineering
University of Thessaloniki,Thessaloniki 540 06,GREECE
A.N.Venetsanopoulos
Department of Electrical Engineering
University of Toronto,Toronto M5S 1A4,CANADA

ABSTRACT Geophysical interpretation is part of geophysical oil prospecting. It evaluates and analyses various kinds of data (seismic, magnetic, geological, geochemical etc.), aiming at the detection of hydrocarbon reservoirs. This work requires a lot of experience and geophysical knowledge. The aim of our research is to develop an expert system which incorporates some of this geophysical knowledge and which can partly perform a knowledge- based automated geophysical interpretation. In its present state the system developed is able to detect and recognize various patterns useful in interpretation. It can also work interactively with an interpreter for improved performance.

1. INTRODUCTION TO SEISMIC INTERPRETATION

A usual method to find oil reservoirs is the seismic technique. Surface explosions are made and the reflections are recorded by geophones. This is repeated along lines on the earth surface. The data collected are processed in a sophisticated way [1].Finally, a subsurface map similar to that of Figure 3 is obtained. This map, together with other kinds of data is used by the interpreter for the interpretation. A detailed analysis of the seismic interpretation can be found in many books eg.[2].

The first task of the interpretation is to identify those lines (called horizons) which correspond to reflections, since they indicate the boundaries of different layers. Each horizon has several attributes eg. its length, strength, signature (shape of the reflection wavelet). The interpreter initially tends to pick up clear, strong and long horizons having distinctive signatures. These horizons indicate the interfaces of major rock layers. Subsequently, the interpreter picks up minor horizons and fills their gaps.

The second task of the seismic interpretation, called structural analysis, is to detect and locate geologic formations which are likely to contain oil reservoirs. Such geologic formations are anticlines, faults, salt domes, reefs, unconformities etc. The information used are the already identified seismic horizons. The detection fo these formations is a very difficult task because the seismic data are almost always noisy and the reflections not very clear. The decision making is based on the knowledge of the region and on the experience of the interpreter.

The third part of the seismic interpretation is the so- called stratigrafic analysis. This part includes the analysis of the texture of the seismic image and of the signatures of the reflections aiming at the information about the various rock layers. This information completes the 'model' of the region already created by the structural analysis.

Seismic interpretation is a labour intensive task. Unlike other parts of geophysical oil exploration, it has not been automated and it does not take advantage of digital signal and data processing techniques available to the scientific community in the last two decades because it is based on experience and it can not easily cast in mathematical form. In the last decade artificial intelligence has provided the means to deal with such cases of specific knowledge available to specialists (experts). Software systems, called expert systems, have been developed in several areas (medicine, education, mineral exploration etc.), which simulate quite successfully the special knowledge and the decision making of human experts [6]. The aim of this work is to construct such a system which incorporates the geophysical knowledge and automates (at least partly) the seismic interpretation.

Until now there have been several attempts to facilitate geophysical interpretation. Most of the work done corresponds to the use of computer graphics and for interactive interpretation [3]. Such systems, although they are not intelligent, speed up seismic interpretation. Another approach is to use image processing techniques (filtering, edge detection, texture analysis) to find global characteristics of the seismic image [4]. Such an approach has several limitations:

a) They cannot store and proccess information described in semantic or syntactic form, as it is needed in the search of seismic patterns

b) They cannot overcome the problem of fuzziness of the seismic data and thus often produce unreliable information.

However a combination of fuzzy set theory and pattern recognition techniques has been successfully used in the detection of seismic horizons [5].

The machine representation of geological knowledge has already been used in mineral exploration (eg. PROSPECTOR) [7], oil well measurement interpretation (eg. DIPMETER ADVISOR [8], LITO [9]) and in geological interpretation [10].

Our approach is to combine image processing techniques with symbolic structure manipulation and measurement of the fuzziness of the results in an expert system.

2. STRUCTURE OF THE SYSTEM

The structure of our system is described in Figure 1. It consists of two separate parts. The first part corresponds to the 'low level vision'. It is composed of image processing routines. The second part of the system corresponds to the 'high level vision'. It searches and encodes the various features of the image in a symbolic form and it stores them in data structures. The search performed is knowledge-based. Both part of the system can be completely automated or work interactively with the interpreter.

The 'lower vision' part of the system includes the seismic image filtering, line extraction and texture analysis. The system has the following image filtering techniques available:

a) 2-D linear low-pass filtering [13]

b) median filtering

c) linear directional filtering

d) nonlinear statistical mean filtering [12].

Linear directional filters are used for line enchancement along one dimension.

Nonlinear statistical filters can be used for line enhancement and thinning. Their use facilitates the task of the line follower.

Line extraction can be performed by the use of an edge follower [13]. The main limitation of the line follower is its sensitivity to the line sharpness. If the lines to be followed are not sharp enough, the seismic image must be filtered by a nonlinear statistical mean filter. Even in this case, the line follower produces sometimes jagged or broken lines. The lines produced must become smooth. This is performed by the use of the line filtering algorithm which filters the coordinates of the line. The results are satisfactory. The gap filling routine connects adjacent broken lines.

The last part of the 'lower vision' of the system is the texture analysis. Texture analysis is part of the seismic stratigraphy and can carry valuable information about the kind of the rocks present, their porosity and about the conditions of their formation (eg under high or low pressure). The results of this analysis can be combined with structural analysis to give better interpretation results. Therefore it can be postponed to be done just before the rock layer analysis of the knowledge-based part of the system. The texture analysis implemented sofar in our system, is quite limited. It can only decide if a region is reflection- free or if it is homogeneous. Thus it can be used in the detection of salt domes and reefs. It can also be used to check the homogeneity of a rock layer. The texture analysis routine is based on an operator which measures the local dispersion of the reflectivity of the seismic image [11].

3. KNOWLEDGE REPRESENTATION

The 'high level vision' part of the system is based on the geophysical knowledge which is stored in the system. The knowledge stored concerns the following geological entities and formations:

1) seismic horizons
2) anticlines
3) faults
4) salt domes
5) reefs
6) unconformities
7) rock layers

The following remarks should be made about these formations:

a) their patterns can be well defined and they are quite general and have modular form

b) elementary patterns are parts of more complicated ones (in a semantic, not in a geological sense). Thus seismic horizons are parts of a fault in the sense that they are needed in the definition of the seismic fault

c) some patterns are special kinds of others (eg. a strong horizontal seismic horizon is a seismic horizon)

d) some patterns appear similar to others in the seismic image(eg. reefs are similar to salt domes)

e) spatial relations are required between geologic formations (eg. neighbourhood relations, parallelism relations).

These remarks have led us to choose frames [16] as knowledge represantation scheme. Each pattern is described in a knowledge package called frame [16] or class [17]. The classes are connected in a semanics network by some relations.

The PART-OF relation connects the classes which are semantically part of a parent- class to their parent [15]. Thus the class 'CONCAVE_HORIZON' is PART-OF the class 'HORIZON'. The PART-OF relation constitutes the decomposition/aggregation axis of the knowledge representation.

The IS-A relation connects a class to its generalization class [15]. The class 'STRONG_HORIZON' is connected by an IS-A relation to the class 'HORIZON'. The IS-A relation constitutes the generalization/ specialization axis of the representation scheme.

The SIMILARITY relation connects classes that appear similar in the seismic images. The class 'SALT_DOME' is connected by a SIMILARITY link to the class 'REEF'.

The system in its current stage has the following spatial relations:

a) HORIZON_NEIGHBOURHOOD
b) HORIZON_PARALLELISM
c) LAYER_NEIGHBOURHOOD

Two horizons are neighbour if no horizon exists between them. Two horizons are neighbours to each other and parallel in the geometrical sense.

Finally, another basic relation is the INSTANCE_OF. It connects a particular entity of a seismic image (eg. a particular horizon) to its class (to the class 'HORIZON').

A procedure is attached to each class. It describes in a procedural form the knowledge about internal constraints of the class and about relations of its elements. Thus our system is a procedural semantics network [17].

The development of the knowledge representation of our system has greatly been influenced by two epert systems developed in the University of Toronto, namely ALVEN (A Left VENtricular wall motion understanding system) [18] and CAA (Causal Arrythmia Analyzer) [19]. Different knowledge representation schemes (eg. production rules) [6] can also be used. However we have not implemented a different scheme because the scheme used satisfied well our needs.

4. CONTROL STRUCTURE FOR PATTERN SEARCH

The control mechanism used in our system is the 'hypothetize and test' [18,19].

A hypothesis is formed when we try to create a new instance of a class. The hypothesis tries to verify itself. This means that the system tries to fill all slots necessary and to test if appropriate slot constraints are valid. If the hypothesis verifies itself then it is inserted in the class (it is instantiated). For example, if the system tries to instantiate a new instance in the class 'ANTICLINE_TRAP', it searches for an instance of the class 'HORIZONTAL_STRONG_HORIZON' which is related by NEIGHBOURHOOD link to an instance of the class 'CONVEX_HORIZON'. If such a combination is found, a new instance of the class 'ANTICLINE_TRAP' is created.

We have used two mechanisms for hypothesis forming: The data directed search and the hypothesis directed search. The data directed search traverses the PART-OF hierarchy in a bottom-up way. Activation of a hypothesis in this search mode activates other hypothesis along the PART-OF hierarchy. It also activates the IS-A parents of the hypothesis. An example of data directed search in our system is the picking of the seismic horizons.

The hypothesis directed search is a top-down traversal of the IS-A and PART-OF hierarchies. The activation of a hypothesis in this mode activates the hypotheses for its slots. The hypothesis is instanciated only when all its slots are filled. An example of hypothesis-directed search is the search of anticline traps described above.

5. PROGRAMMING CONSIDERATION AND EXAMPLES

The whole system in its present form has been programmed in 'C' language on a VAX 780 computer using the UNIX operating system. The reasons for using C is its capability to define a variety of data structures and its speed when it is combined with UNIX. Other languages (eg. PASCAL) can be used equally well. The various classes described in the knowledge representation have been implemented as data structures in C. The various relations (IS-A, PART-OF, SIMILARITY, spatil relations) have been implemented by using pointers from one data structure to the other. The insertion of an instance in a class is implemented by a procedure which creates an new object in the data structure. The interslot constraints and the certainty calculations are included in these procedures. This system organization is very efficient but it is quite complex because part of the knowledge is stored in the instantiation procedures which are visible to the programmer. This problem can be completely solve by using a high-level language for knowledge reprresentation (eg. PSN) [15].

The system has been tested with real data and its performance is very satisfactory. An example of its performance is shown in Figure 2. The lines detected by the line follower in the seismic image of Figure 2a are shown in Figure 2b. The system recognizes the seismic horizons, evaluates their certainty and tries to find geologic formations. It finally recognizes the two faults shown in Figure 2c.

REFERENCES

1. E.A. Robinson, S.Treitel 'Geophysical Signal Analysis', Prentice Hall, 1980

2. N.A. Anstey 'Seismic Interpretation: The physical aspects', Int. Human Res. Dev. Corp., 1977

3. H. Royce Nelson 'New technologies in Exploration Geophysics', Gulf Pub. Co., 1983

4. N.Keskes, A.Boulanmar, O.Faugeras 'Application of image analysis techniques to seismic data' Proc. IEEE ICASSP-1982, Paris

5. P.Bois 'Fuzzy seismic interpretation' IEEE Trans. on Geoscience and Remote Sensing, Vol. GE-22, No.6, pp.692-697, Nov. 1984

6. A.Barr, P.R.Cohen, E.A.Feigenbaum 'The Handbook of artificial Intelligence', vol.1,2,3, Heuristech Press 1981

7. R.O.Duda 'Development of the PROSPECTOR consultation system for mineral exploration' (final report), SRI Int.

8. R.Davis, H.Austin, I.Carlbom, B.Frawley, P.Pruchmik, R.Schneiderman, J.A.Gilreath 'The DIPMETER ADVISOR: Interpretation of geologic signals', Proc. Int. Joint Conf. Artificial Inelligence, pp.846-849, 1981

9. A.Bonnet, C.Dahan 'Oil-well data interpretation using expert system and pattern recognition techniques' Proc. IJCAI, pp.185-189, 1983

10. R.G. Simmons 'Representing and reasoning about change in geologic interpretation' M.Sc., MIT, 1983

11. W.K. Pratt 'Digital Image Processing', J.Wiley, 1978

12. I.Pitas, A.N. Venetsanopoulos 'Nonlinear statistical means in image processing' IEEE Trans. on ASSP, June 1986

13. A.Levy-Mandel, J.K.Tsotsos, A.N.Venetsanopoulos 'Knowledge-based landmarking of cephalograms' Proc. 12th Biennial Symposium on Communications, Kingston, June 1984

14. I.Pitas, A.N.Venetsanopoulos "Edge detectors based on nonlinear filters', IEEE Trans. on PAMI, in press

15. Special issue on knowledge representation, Computer, Oct. 1983

16. M.Minsky 'A Framework for representing knowledge', Psychology of computer vision edited by Windston, McGraw-Hill, 1975

17. H.Levesque, J.Mylopoulos 'A procedural semantics for semantic networks' in Representation and understanding: Studies in Cognitive Science (D.Bobrow, A.Collins editors), Academic Press, 1979

18. J.K.Tsotsos 'Temporal event recognition: An application to left ventricular performance evaluation' Proc. IJCAI 1981

19. T.Shibahara et al., 'CAA: A Knwledge-based system with causal knowledge to diagnose rythm disorders in the heart', Proc. Can. Soc. Comp. Studies Intel. Conf., 1982

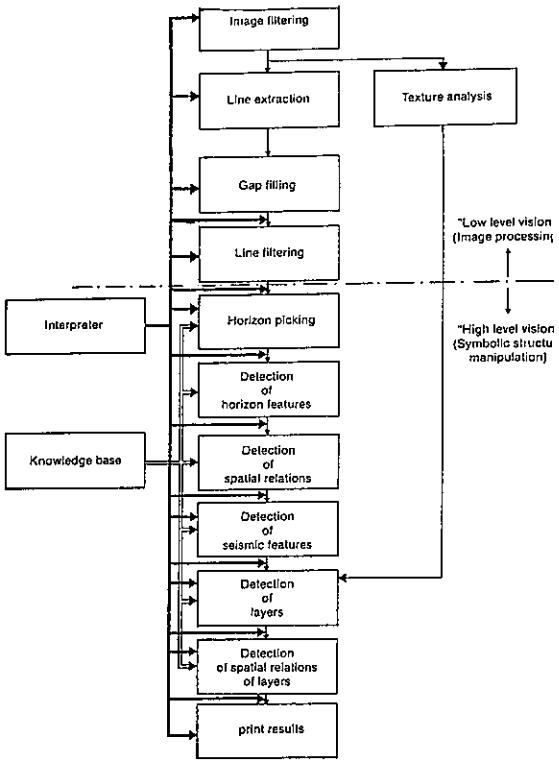


Figure 1: Structure of AGIS

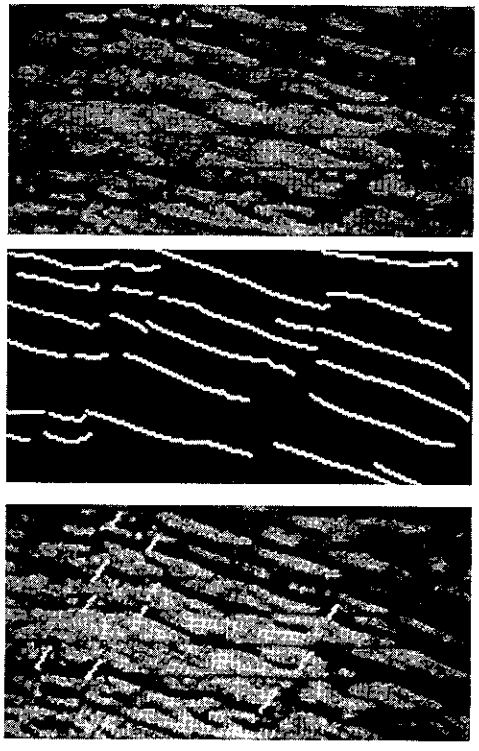


Figure 2: (a) Seismic image (b) result of the line follower (c) detected faults

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A DYNAMIC TIME WARPING CUSTOM INTEGRATED CIRCUIT FOR SPEECH RECOGNITION

Riccardo Cecinati (*), Alberto Ciaramella (**), Giovanni Venuti (**) and Cesare Vicenzi (*)

(*) ELSAG - Via Hermada, 6 - 16154 Genova (Italy)
(**) CSELT - Via G. Reiss Romoli, 274 - 10148 Torino (Italy)

ABSTRACT

A system for left to right connected word recognition is naturally suited to a master-slave implementation, where the master is charged with grammar and slaves charged with single words recognition. Slaves' task is characterized by heavy computation and memory requirements, though it is regular and well structured. In order to optimize performance and simplify system implementation, we are designing a custom integrated circuit performing efficiently the slave task, whose functional requirements and characteristics are explained in this paper.

1. INTRODUCTION

The single pass dynamic time warping algorithm [1] is an efficient method for implementing connected word recognition; it can be applied both when the words are represented by Hidden Markov Models (H.M.M.), and when the words are represented by templates [3].
The whole algorithm can be split into two levels, an high level SLDP (Sentence Level Dynamic Programming) that deals with the syntactic knowledge, represented as a regular grammar where each transition represents a word of the lexicon, including the silence, and a low level WLDP (Word Level Dynamic Programming) which deals with the acoustical knowledge of each word.
Fig.1 and 2 exemplify both the high level and the low level task description.

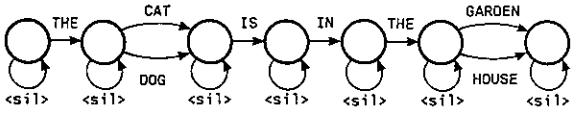


Fig. 1 - Simple example of a regular grammar

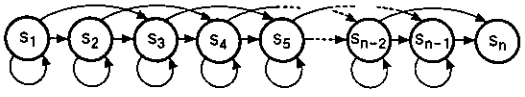


Fig. 2 - Example of word description

2. TASK CHARACTERIZATION AND SYSTEM ARCHITECTURE

The task is statically characterized by the number of words W of the vocabulary and by the average number of states Saw for word, and it is described by the syntax graph characterized by the number of nodes N and by the branching factor bm. The hierarchical organization of the whole algorithm (fig. 3) is naturally fitted to a master-slave tree architectur, where a single master implements the less regular, but also less computationally demanding task of the SLDP, and one or more slaves (*) implement the more regular, but also more computationally demanding task of the WLDP: each slave performs the WLDP for a subset of the whole vocabulary, permitting an easy system expansion.
The master memory contains a functionally read only area for storing the grammar description and a writable working area for storing the temporary scores and related information of survived paths across words; in a similar way the slaves contain a functionally read only memory for storing the acoustical descriptions

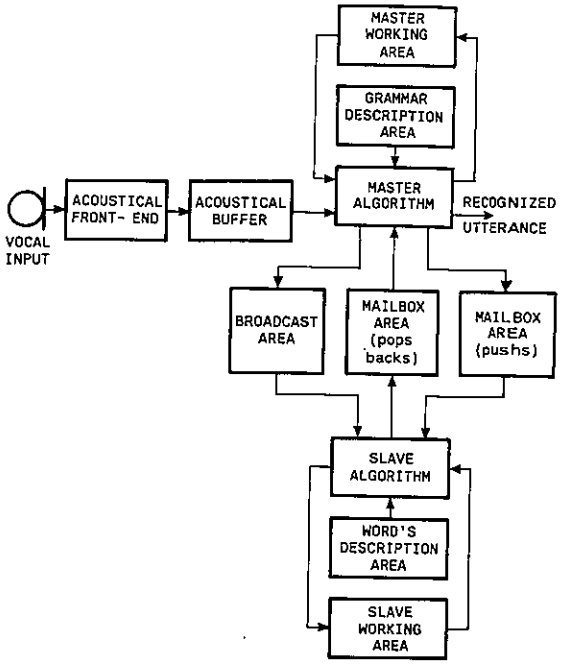


Fig. 3 - System block diagram

of each word and a writeable working area for storing the temporary scores of survived paths inside words. The master is charged with forward extending possible paths across grammar nodes and singling out the best survived path by backtracking; for reducing computational requirements backtracking is per-

(*) Patent Pending

formed from time to time.
The slaves are only charged with extending possible paths inside grammar nodes. A grammar node that in a given frame t is examined by a slave for extending paths is called an active node na; the slave performs an algorithm structured in vector form between the first and the last active state; hence the average number of states of an active node which are "visited" in a frame is Krid*Saw, with Krid less than 1. The total number of active nodes in a time frame is Na(t); the total number of active states instead is Krid*Saw*Na(t).
Broadcasted messages are constituted by parameters extracted by the acoustical front-end and by messages sent from master to all slaves in each frame, as for example the threshold that discriminates between survived and canceled states in the frame.
Information sent from a slave to the master is constituted by "pop" messages, which notify the possible end of a word in a given frame with the accumulated score and the reference to the starting frame in the word along the path that is now "popping" , and "back" messages, which resume words still surviving in the slaves.
After a "pop" message, the master singles out the words that can follow, according to the grammar, and in the following frame addresses the slaves in charge of these words with messages called "push" to start the recognition of these words.
The number Na(t), the number of push Npush(t), the number of pop Npop(t) characterize dynamically the system behaviour; their evolution frame by frame is modeled in fig.4. In each time frame t in fact we process the active nodes and the new "pushed" nodes; some of these nodes are discarded (with probability Pdis) because in them no state reaches the threshold; let these have the cardinality Ndis(t). From the others survived some (with probability Ppop) reach the final state and originate "pops". Every pop originates bm pushes as an average, hence in the following frame we have bm*Npop(t)=Npush(t).

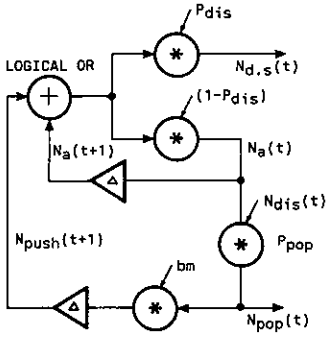


Fig. 4 - Probabilistic model of system traffic evolution

3. SLAVE ALGORITHM

The WLDP algorithm implemented by the slave is activated at each frame and is composed by a main program that orderly activates for each node a dynamic programming subroutine, structured in vectorial form. Temporary scores and related information of each active node na(t) in a frame are recorded in a vector form; vectors of active nodes are contiguously allocated in a circular buffer, updated frame by frame.
According to the system initialisation it is possible to perform both the prototypical and the markovian approach. In both cases having received the input j(t) we calculate for each active node the dynamic programming by evaluating for each active state i the best score S(i,j(t)) for every path reaching i; S(i,j(t)) is related to the score of the previous frame by the formula:

$$S(i,j(t)) = \min_{a=0,1,2} [S(i-a,j-1(t-1)) + L(i,j(t)) + P(i-a,i)] \quad (3)$$

in the markovian case these quantities represent probabilities (more exactly complemented logarithms of probabilities) and in the prototypical case they represent distances.
P(i-a,i) is a transitional score, that in the prototypical case is a time distortion penalty [1] and in the markovian case is the state transition matrix A [2]; L(i,j(t)) is a local score that depends on state i and on input frame j(t): this last score is calculated differently in the markovian and prototypical cases implemented.
For calculating L(i,j(t)) in the markovian discrete input case we use the state spectral emission matrix B(i,ks)[2], which describes the probability of the emission of each spectral codebook symbol ks for each state i of the H.M.M., and the state energy emission matrix C(i,ke), which describes the probability of the emission of each energy codebook symbol ke for each state i of the H.M.M.; hence we quantize the input j(t) by extracting the index ks of the nearest point of the spectral codebook and the index ke of the nearest point of the energy codebook.
In the prototypical case we can choose between several alternatives, i.e. to quantize or not the reference frame and to quantize or not the template. The more conservative approach would be to not quantize, but this involves a lot of on line distances computations; we found that a comparable performance at a reduced implementation cost is attainable by quantizing the template memory only.
Fig.5 summarizes the differences of the two implementations which are symmetrical: in the markovian case we store the distances (i.e. probabilities) between the template T and each codebook symbol Ci, and we approximate the input X with the nearest codebook symbol, in the prototypical case instead we measure the distances between the input X and each codebook symbol Ci, and we approximate the template T with the nearest codebook symbol.

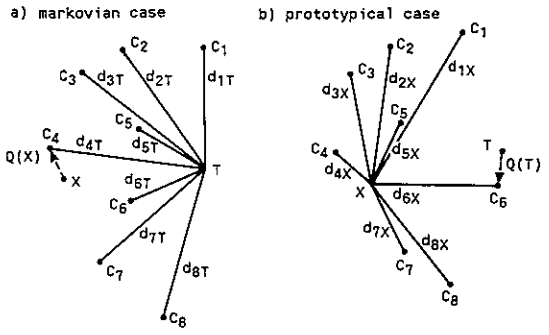


Fig. 5 - Comparison between local distance computation in the markovian and prototypical implemented

4. SYSTEM REQUIREMENTS

We found that computation and memory requirements of the master are manageable with the present technology and even the potential bus bottleneck is not a problem for the intended tasks.
The potential problems in the slave are the memory requirements, related to static task characteristics, and the computational requirements, related to dynamic task characteristics.
Memory requirements are mainly dictated by the acoustical memory occupancy, which is related to the number of words of the vocabulary.
In fact in the prototypical case we found:

W*Swapr*K1 (4.1)

and in the markovian case:

W*Swama*K2 (4.2)

given that Swapr is the average number of states of a word in the prototypical approach and Swama is the average number of states in a word in the markovian approach; for a quick evaluation we can take Swapr=50 and Swama=25.
K1 and K2 take into account the number of acoustical parameters needed for describing a state; in both cases we need 3 parameters for describing the transition matrix, then in the prototypical case we need only a codebook index, whereas in the markovian case we need a vector of scores of the dimensions Ks of the spectral codebook and a vector of scores of the dimensions Ke of the energy codebook; hence K1=4 and K2=3+Ks+Ke.
We can see that the markovian case is much more memory hungry; for example with a spectral codebook of 128 simbols and an energy codebook of 2 simbols an application of 256 words needs approximately 860K words of memory for acoustical parameter storing.
The statistics of Na(t) defines instead the computational requirements of the slave, at least as a first approximation: in fact if the dynamic programming of a state is performed in the time interval Tst, in order to reach the real time we must have:

Tst*Swapr*Krid*Naav =< Tframe (4.3)
or
Tst*Swama*Krid*Naav =< Tframe (4.4)

in the prototypical and the markovian approach respectively, where Tframe is the frame duration (in our case 10 ms.) and Naav is the average number of active nodes for frame.
Hence, given the memory and computation requirements and the present DSP limitations [4], we implemented the slave functions in a dedicated custom chip, called RIPAC, that is an achronim for "RIconoscimento del PArlato Connesso", i.e. "connected speech recognition". With a dedicated chip we estimate a performance improvement of an order of magnitude with respect to what can be done by today general purpose DSP; other than this the chip is designed with the aim of addressing a large amount (up to 16 Mbytes) of external RAM without wait states for ordinary access times.

5. CUSTOM ARCHITECTURE AND FUNCTIONS

The custom is implemented as an Harvard machine, with an internal microprogram ROM, an internal data buffer, an arithmetic unit designed for the efficient implementation of the dynamic programming and an independent unit for addressing external data RAM; the block diagram is reported in fig.6.

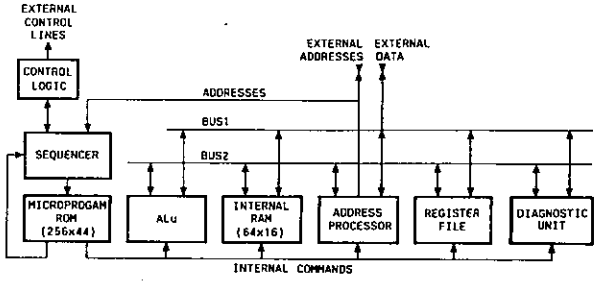


Fig. 6 - Custom block diagram of RIPAC

The Harvard architecture allows the contemporary addressability of the microprogram memory and of two data memories; in fact it is possible to access an internal fast data memory of 64 words by 16 bits and an external memory, whose maximum addressability is of 8 Mwords. The internal data memory records intermediate results, hence minimizes the data exchanges with the external data RAM.
The internal ALU is oriented to the task of dynamic programming, which basically consist of sums and comparisons; in this last case the chosen best score automatically carries also its own side information (i.e. pointers). Scores are manipulated as 12 bit unsigned saturated integers, since this precision has been found satisfactory in simulations.
External data RAM address logic optimizes the time access both to the acoustical parameters

area, recorded in a fixed area in matrix form, and of active words scores, recorded in a circular buffer in vector form: the address logic hence is provided with an internal RAM area for storing key entry point and pointers of the external data structure and with an address computation ALU of 24 bits, provided of circular buffer wrap-around control.
The external data RAM allows the allocation of the acousticparameters for 128 templates to 1024 templates and of the circular buffer sized from 8k words to 64 kwords; the actual sizes are programmed by a command word, which can also sets the prototypical or the H.M.M. approach.
A status word instead monitors the conditions of error and warnings (for example a specific warning is issued when the free space of the circular buffer is becoming too small).
The internal ALU, the internal data buffer and the external RAM address logic are linked by two internal busses and controlled by the internal microprogrammed sequencer.
The internal microprogram ROM is constituted by 256 words, of 44 bits each, with an instruction cycle of 125 nsec.: the parallel organization of words and the internal placement are aimed at the maximum speed performance.
The WLDP microprogram, which covers two thirds of the ROM area; the last thir of RAM area is used for initializing and diagnostic functions.
The chip has 55 input and output signals, plus power and ground, hence it can be conveniently packaged as a 68 pin grid array.

6. CHIP USE AND PERFORMANCE

A slave board using the described component is equipped with a RAM area, which is accessible both from the recognition component for performing frame by frame WLDP, both from system bus for initialization and messages exchanges between sections (pops, pushes, front-end parameters).
The implemented custom is optimized from the point of view of the size of the external RAM, and from the computational throughput: in fact we found that a first approximation we can control in a time frame of 10 ms. 3.2 k states, that also means, with Krid=0.5, to have 128 active grammar nodes in the prototypical case and 256 in the markovian case. This is in fact an upper bound that we have to reduce in practice for two reasons:

- the internal chip overhead due to managing pop and push messages and for updating active nodes,
- the external overhead for synchronizing master and slave and interchange messages.

However the predicted performance is quite impressive and we see that for a wide range of vocabulary dimensions and branching factors one chip for system is enough both for memory, both for throughput point of view.

ACKNOWLEDGMENTS

We acknowledge L. Fissore, who evaluated the required internal precision needed in the markovian case, and R.Fossati, M.Paolini, L.Licciardi, R.Tasso, A.Torielli who are developing the VLSI implementation of the chip.

BIBLIOGRAPHY

[1] J.S.Bridle, M.D.Brown and R.Chamberlain "An Algorithm for Connected Word Recognition" Proc. IEEE ICASSP-82, vol.2, pp.899-902

[2] S.E.Levinson, L.R.Rabiner, M.M.Sondhi "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Model to Automatic Speech Recognition" B.S.T.J., Vol. 62, n.4, April 1983, pp. 1035-1072

[3] M.Cravero, L.Fissore, R.Pieraccini, C.Scagliola "Sintax Driven Recognition of Connected Words by Markov Models" ICASSP-84, pp. 35.1.1-4

[4] J. Allen "Computer Architecture for Digital Signal Processing" Proceedings of the IEEE, may 1985, pp. 854-873

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

PARAMETER OPTIMIZATION OF THE CORDIC-ALGORITHM AND IMPLEMENTATION IN A CMOS-CHIP

G. Schmidt*, D. Timmermann**, J.F. Böhme*, H. Hahn**, B.J. Hosticka**, and G. Zimmer**

* Lehrstuhl für Signaltheorie
Ruhr-Universität Bochum
Universitätsstr. 15o, D 463o Bochum

** Fraunhofer Institut für Mikroelektronische Schaltungen und Systeme
Bismarckstr. 69, D 41oo Duisburg 1

1. INTRODUCTION

The CORDIC algorithm describes an iterative procedure to compute trigonometric, hyperbolic and inverse functions as well as multiplications and divisions. This variety of CORDIC functions is calculated with a unified set of recursive equations. Each iteration consists of simple add and shift operations and can be interpreted as a pseudorotation of a two-component vector (Coordinate Rotation Digital Computer) [1],[2].

The shifts in the recursions which we call CORDIC sequence define the angles of rotations and determine all the important properties of the CORDIC algorithm e.g. convergence, accuracy, scaling factor and region of convergence. CORDIC sequences suitable for special groups of CORDIC functions are well known, e.g. [3],[4].

The integration of a CORDIC processor with a pipeline architecture however requires CORDIC sequences with shifts which are independent of the CORDIC function to be calculated. These shifts can then be implemented as hard-wired shifts on a chip and barrel shifters or multiplexers which would increase hardware amount considerably become unnecessary.

This paper reports on a systematic search for unified CORDIC sequences with optimized properties and a CMOS integrated pipeline stage which will be used for testing and simulation of the CORDIC sequences. The test chip realizes one iteration and contains barrel shifters due to the chosen recursive architecture to execute the complete CORDIC algorithm.

Interesting realizations and applications which are not within the scope of this paper may be found in [3-1o].

2. The CORDIC algorithm

The CORDIC algorithm is given by the recursions [2], [4]:

$$\begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} 1 & -m\sigma_i\delta_{m,i} \\ \sigma_i\delta_{m,i} & 1 \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix}, \qquad (1)$$

$$z_{i+1}=z_i+\sigma_i\varepsilon\alpha_{m,i} \qquad i=0,\ldots,N-1.$$

where

- N number of microrotations,
- m parameter for the coordinate system (=-1, 0, 1),
- α CORDIC angle,
- σ direction of the rotation (=±1),
- ε constant to control the sign of the CORDIC functions (=±1),
- δ iteration step size.

Choosing the σ- values so that either $y_N \to 0$ (vectoring mode) or $z_N \to 0$ (rotation mode) diverse trigonometric (m = 1) and hyperbolic (m=-1) functions and products and quotients (m=0) of the initial values x_o, y_o, z_o can be calculated.

The CORDIC angles are defined by one or two non-negative integers S and S' [3]:

$$\frac{1}{\sqrt{m}}\tan(\sqrt{m}\alpha_{m,i})=\delta_{m,i}=2^{-S(m,i)}+\eta_\alpha(m,i)\,2^{-S'(m,i)}, \qquad (2)$$

$\eta_\alpha=-1,0$. Therefore execution of the x-y recursions results in performing simple add and shift operations.

Since the magnitude of the vector $(x,y)'$ is changed by each iteration (pseudorotation) the results x_N and y_N have to be multiplied by the inverse scaling factor

$$K_m^{-1}=\prod_{i=0}^{N-1}(1+m\delta_{m,i}^2)^{-1/2}$$

to get the correct results.

We call $(S(m,i),\eta_\alpha(m,i),S'(m,i))$, i=0,...,N-1 a CORDIC sequence for the coordinate system with parameter m.

3. The optimization problem

The CORDIC algorithm has to fulfill the following requirements:

(A) The angles $\alpha_{m,i}$ have to satisfy the convergence criterion:

$$\alpha_{m,l}-\sum_{j=l+1}^{N-1}\alpha_{m,j}<\alpha_{m,N-1}, \qquad l=0,\ldots,N-2.$$

(B) The resolution has to be large enough to guarantee full 16-bit accuracy of the CORDIC functions i.e. $\alpha_{m,N-1}$ has to be sufficiently small.

(C) The region of convergence RC_m must be at least $[-\pi,\pi)$ for trigonometric functions (m=1) and as large as possible for m=0 and m=-1 . An infinite region of convergence is desirable but cannot be realized with a finite number N of microrotations.

Further specifications should be met to improve algorithmic properties or to reduce hardware amount of a CORDIC processor realization with pipeline architecture:

(D) The scaling operation following the CORDIC recursions should be carried out in a simple way e.g. with a single shift: $K_m^{-1}=2^{-T(m)}$, where T is an integer [4].

(E) The CORDIC shift sequences should be independent of the coordinate system i.e. $S(m,i)=S(i)$, $S'(m,i)=S'(i)$.

(F) The CORDIC algorithm should be parametrized by CORDIC sequences fulfilling conditions (A)-(E) with a minimum number N of microrotations.

(G) The number N_d of microrotations with $\eta_\alpha \neq 0$ should be minimal since double shift iterations entail approximately twice the hardware costs of single shift iterations.

The new requirement (E) has been added to the catalogue because m-independent shifts can be implemented as hard-wired shifts in a CORDIC pipeline chip. Consequently the hardware amount is reduced and is mainly determined by the high-speed adders.

Since all the requirements are connected with the parametrization of the CORDIC algorithm by shifts and η_α-values the optimization problem is to determine CORDIC sequences which satisfy all the specified - partly contradictory (e.g. (B),(C) and (F)) - requirements.

4. A strategy to solve the optimization problem

4.1. Algorithmic extensions

The double shift concept [3] has been adopted because the increased number of parameters S, η_α and S' gives more freedom to influence the algorithmic properties. We further increase flexibility by two means:

- The number of CORDIC angles is approximately doubled by allowing a third value $\eta_\alpha(m,i)=+1$ in (2).
- Double shift scaling factors are introduced: $K_m^{-1}=2^{-T(m)}+\eta_K(m)\,2^{-T'(m)}$, where T,T' are integers and $\eta_K=-1,0,1$. Consequently there are much more combinations of K_{-1} and K_1 to be investigated and the chance to find the desired CORDIC sequences increases considerably.

4.2. Restriction of the parameter space

The search for unique CORDIC sequences is carried out by FORTRAN programs which essentially consist of nested DO loops for the parameters S, η_α and S' of the microrotations. Since the total number of combinations to be investigated equals the product of the loop ranges it is necessary to restrict the loop ranges of the parameters in order to avoid gigantic numbers of combinations.

It is easy to verify the following dependencies between the shifts:

- $S(i+1)\geq S(i)$,
- $S'(i)>S(i)$,
- $S(i+1)=S(i)+d$, $d\leq 3$,
- $S(i+n_r)=S(i)$, $n_r\leq 4$,

where the final two result from the convergence criterion and the hardware costs.

These dependencies restrict the parameter space considerably without loosing one of the desired CORDIC sequences. With the help of effective conditional branches in the outer loops of the programs concerning the scaling factor K_m we have succeeded to reduce computation time drastically so that the search can be carried out with a fast array processor.

4.3. Execution of the search for unique CORDIC sequences

A four stage process is used to search for unique CORDIC sequences:

- Hyperbolic CORDIC sequences (m=-1) were determined for N'=16 microrotations and a maximum number N_d=6 of double shift microrotations located at the beginning of the algorithm (i=0,...,5). The following list of scaling factors K_{-1} was investigated: 0.250, 0.258,0.267,0.285,0.333,0.400,0.444,0.471, 0.485,0.500,0.516,0.533,0.571,0.667, with a limit of relative error Δ_K=0.00001.
- Maintaining the shifts of the hyperbolic CORDIC sequences all combinations of η_α-values (3^6=729) were tested for convergence criterion and scaling factor condition. In case of positive results unique CORDIC sequences were found for m=-1, m=1 and N'=16. The list of scaling factors to be tested was: 1.333,1.600,1.778,1.882,1.939,2.000,2.065, 2.133,2.286,2.667,3.200,3.556,3.765,3.879, 4.000,4.129,4.267,4.571,5.333,6.400, with the same accuracy Δ_K=0.00001 for K_1. Requirement (C) concerning the region of convergence RC_1 is weakened to $[-\pi/2,\pi/2)$ because we plan to use a special stage in front of the CORDIC pipeline to perform a 180° rotation which is accomplished by sign reversion of the (x,y)-vector.
- Maintaining the shifts and η_α-values of the double shift microrotations which have the strongest influence on the scaling factors the single shift microrotations were optimized again and the CORDIC sequences are extended until S(N-1)=16 (N>N') in order to fulfill requirement (B).
- Finally the η_α-values were adapted for m=0 to maximize RC_0.

5. Results of the optimization procedure

The procedure described in 4.3 supplied us with a catalogue of CORDIC sequences which fulfill the formulated conditions (A)-(G). Further requirements are necessary to select the best among these sequences. Table 1 contains two of the best results.

	Example 1					Example 2				
i	S	η_α			S'	S	η_α			S'
		m=-1	m=0	m=1			m=-1	m=0	m=1	
0	1	1	1	1	3	1	1	1	1	6
1	1	1	1	1	3	1	1	1	-1	11
2	1	1	1	0	5	2	1	1	0	3
3	2	1	1	1	3	3	1	1	1	4
4	2	1	1	-1	8	3	1	1	0	6
5	2	-1	0	-1	7	4				
6	3					5				
7	4					5				
8	5					6				
9	6					7				
10	7					8				
11	8					9				
12	9					9				
13	9					10				
14	10					11				
15	11					12				
16	12					13				
17	13					14				
18	14					15				
19	15					16				
20	16									
K_m		0.44	1.00	1.78			0.67	1.00	1.33	
RC_m		3.21	2.91	2.67			2.00	1.88	1.65	

Table 1: CORDIC sequences with optimized properties

Example 1 shows a unique parametrization of the CORDIC algorithm with N=21 microrotations and N_d=6 double shift microrotations. Scaling of x_N and y_N with K_m^{-1} means multiplication by $0.444^{-1}=2^1+2^{-2}$ (m=-1) or $1.778^{-1}=2^{-1}+2^{-4}$ (m=1) and requires execution of shift and add operations.

In comparison with example 1 the second example saves one microrotation and one double shift microrotation: N=20, N_d=5. The examples reveal the fact that N and RC_m decrease as K_m tends to one.

Both examples have the additional advantage that the ratio K_1/K_{-1} is a power of two. This simplifies the architecture of the scaling unit because x_N and y_N can generally be scaled by either K_{-1}^{-1} or K_1^{-1} independent of m=-1 or m=1.

The correct result is then read out directly or one or two bits shifted from the output register by a multiplexer.

These and other CORDIC sequences will be simulated with fixed point arithmetic on computer and on the test chip described below.

6. Hardware implementation

The hardware design philosophy of the CORDIC processor was influenced decidedly by the underlying algorithm and by the realization technology. The architecture is based on a fully parallel concept. The implementation of the CORDIC operations requires adders, shifters, and registers. The circuit technique employed was synchronous static CMOS. The device is fabricated in an advanced CMOS silicon-gate n-well process with single poly and metal layer. Channel length is 2.5µm for n-type and 1.5µm for p-type MOS transistors. The gate oxide thickness is 40nm.

As far as the design style is concerned we have opted for the bit-slice technique. Bit-slice circuits are often used for microprogrammed processors. In this technique only one n-bit wide data path is to be designed. It can be readily extended to achieve a word length of kn bits, thus taking advantage of high CORDIC regularity. Hence, we must first specify the kn word length based on the algorithm itself and on its applications. Given this we must optimize for n with respect to area, power, and speed. As a good compromise between chip area, power dissipation, and dynamic range we have chosen 16-bit fixed point arithmetics. That requires approximately N=20 iterations which yields a word length N+ld(N)≅24 bit [2].

6.1 Hardware components

The most speed-critical component of the CORDIC arithmetic is the adder due to carry propagation. We have investigated device count, latency time, and propagation delay time for several types of adders. The results are summarized in table 2 for an 24-bit adder.

	Device Count	Latency [ns]	Propagation Delay [ns]
Ripple Carry	420	42	42
Carry Look Ahead (4 Bit-Slice)	1368	25	25
Carry Save (4 Bit-Slice)	3024	54	9
Carry Select	902	15	15

Table 2: 24 Bit Adders

While the device count determines the chip area and power consumption, the propagation delay time bears an effect on the throughput rate. The latter is influenced by the latency time as well, because we are using a recursive structure. Considering the table 2 the best compromise seems to be the carry-select concept.

As a result of the adder optimization procedure we have divided the 24-bit adder into four 4-bit sections plus one 8-bit section. This implied use of 4-bit slice technique for the entire chip. In this way the 24-bit adder achieves about the same propagation delay time as an 8-bit adder, while the adder chip area increases only by a factor of 5/3. The 4-bit and 8-bit sections use the Manchester carry principle.

The shifter is of a barrel-shifter type that needs no decoding. The barrel-shifter allows testing of various CORDIC sequences thus greatly enhancing the range of applications of the processor. It goes without saying that a CORDIC processor with hard-wired shifts would spare the chip area but the sequence would be fixed. Nevertheless, such a processor is feasible, if the proper sequence has been found and tested. Hence, the presented concept can be looked upon primarily as a vehicle for CORDIC sequence testing.

The registers are of master-slave type consisting of inverters.

The last design aspect to be addressed is testing. Our implementation of the CORDIC allows extensive testing using scan-path method. This technique allows prototype measurements and functional tests. The overhead includes two transistors per register.

6.2 Layout

The layout style was largely dictated by the bit-slice technique employed in the design. Attention had to be paid to short interconnection lines because only single metal layer was available. The layout contains approximately 4o% of the hardware needed to implement a full CORDIC processor (i.e. we need 2,5 chips).

The characteristic data are:

2x24 bit adders, 2x24 bit barrel-shifters, 2x24 bit registers, total of 5ooo transistors, 1Omm^2 of silicon area, and 2OMHz clock rate.

Our estimate for a hard-wired full CORDIC is as follows: 9OOO transistors, 18mm^2 silicon area and 3O MHz clock rate using the above technology.

7. Summary

We have presented parameter optimization of the CORDIC algorithm and its subsequent implementation on a CMOS chip. Unified CORDIC sequences for the calculation of all CORDIC functions were presented. The emphasis during hardware design was on high regularity and modularity.

The use of barrel-shifters makes it possible to test various CORDIC sequences if desired. Scan-path technique was incorporated to ensure low testing costs.

Acknowledgements

The authors would like to thank Bin Yang for his assistance in developing the program system and the execution of the time-consuming search for unique CORDIC sequences.

REFERENCES

[1] Volder, J.E.: The CORDIC Trigonometric Computing Technique, IRE Trans. (1959), Vol. EC-8, No.3, pp. 33o-334

[2] Walther, J.S.: A Unified Algorithm for Elementary Functions, Proc. SJCC (1971), pp. 379-385

[3] Deprettere, E.F., Dewilde, P., and Udo, R.: Pipelined CORDIC Architectures for Fast VLSI Filtering and Array Processing, Proc. ICASSP (1984), 41A.6.1-41A.6.4

[4] Ahmed, H.M.: Signal Processing Algorithms and Architectures, Ph.D. Thesis, Stanford (CA), (1981)

[5] Ahmed, H.M., Delosme, J.-M., and Morf, M.: Highly Concurrent Computing Structures for Matrix and Signal Processing, IEEE Computer (1982), No.1, pp. 65-82

[6] Deprettere, E.F.: Synthesis and Fixed-Point Implementation of Pipelined True Orthogonal Filters, Proc. ICASSP (1983), 1, pp. 217-22o

[7] Haviland, G.L. and Tuszynski, A.A.: A CORDIC Arithmetic Processor Chip, IEEE Trans. on Computers (198o), No. 2, pp. 68-79

[8] Despain, A.M.: Very Fast Fourier Transform Algorithms Hardware for Implementation, IEEE Trans. on Computers (1979), No. 5, pp. 333-341

[9] Hosticka, B.J., Büddefeld, J., and Kleine, U.: Power-Wave Digital Filters Using CORDIC Adaptors, AEÜ (1985), No. 4, pp. 242-244

[1o] Hahn, H., Büddefeld, J., Hosticka, B.J., and Kleine, U.: CORDIC Realization of Power-Wave Digital Filters, Proc. ECCTD (1985), pp. 5o7-51o

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A VLSI BUILDING BLOCK FOR SIGNAL PROCESSING

B.BARAZESH T.R.T., LE PLESSIS ROBINSON - FRANCE

J.C. MICHALINA, THOMSON-SC - GRENOBLE - FRANCE

A new high performance digital Signal Processor is described in this paper. Architectural solutions are highlighted that enhance the throughput for complex algorithms and allow to build efficient multiprocessor architectures. System architectures are discussed and benchmarks are presented.

1. INTRODUCTION

Single chip digital signal processors have taken on an ever more prominent role in signal processing applications. The advent of the first generation of these processors [1], [2], [3], allowed to implement simple algorithms more or less easily on silicon. But more complex algorithms boost fur higher througputs in programmable architectures.

A digital signal microcomputer "PSI" is described in this paper that offers new architectural solutions to enhance the performance for digital signal processing algorithms. Section 2 presents the architecture of the PSI. The instruction set and benchmarks for standard algorithms are described in section 3. General system architectures are discussed in section 4.

2. ARCHITECTURE

2.1. Internal architecture

The internal architecture of the PSI (Fig. 5) is based on a highly parallelized three data bus (3x16) structure. This architecture is controlled by a 32 bit wide instruction allowing to realize concurrent operations in a single instruction cycle, 160 ns.

Multimode computations are feasable in the PSI easily since the architecture is configurable dynamically in three modes : Real 16 bits, complex 2 x 16 bits (real part, imaginary part), and real 32 bits (lower part, higher part). It's important to note that the same instruction set is available in all three modes. A single instruction realizes a complex multiplication (4 real multiplications) in a doubled (320 ns) instruction cycle. This high throughput (12,5 x 10^6 multiplications/s) is achieved by a four stage pipelined parallel multiplier and a dedicated adder running at 80 ns/multiplication + addition. The program length can be reduced significantly by using the approrpriate computational mode.

Another important feature is the algorithmic architecture. Optimized architectures for a set of kernel algorithms has been studied (FIR, adaptive FIR, IIR, FFT, Lattice forms, ...) in a programmable architecture. Theoretical bounds can be found for these algorithms in terms of memory access cycles or computation cycles. Optimized architectures achieve the theoretical bounds. The internal architecture of the PSI is the synthesis of optimized architectures with more general possibilities required for a programmable DSP. As an example the gradient FIR adaptive filter needs only two cycles to update a coefficient and realize a multiplication/accumulation for output computation.

Figure 2 gives a closer insight of the ALU. The barrel shifter on A input scales data coming form BUSL or register P. Two 32 bits accumulators and a pipeline FIFO (4x16 bits) memory are provided at ALU output. A replace code register (RC) allows to execute dynamically an ALU code from data RAM memory. As a result data dependent operations are accelerated in time critical loops by avoiding jump instructions. This can be used efficiently in a data-driven echo canceller, [4] (FIG 1).

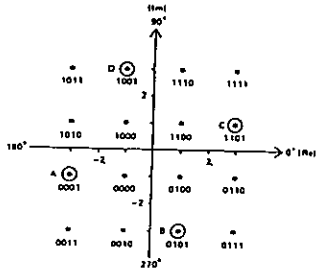


FIGURE 1 - V32 SIGNAL SPACE DIAGRAM

Non redundant coding

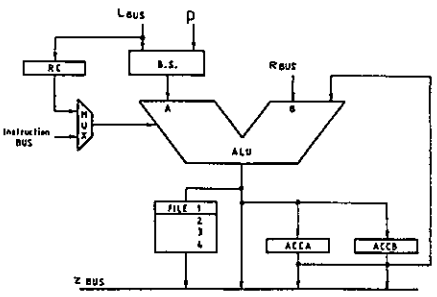


FIGURE 2 - ALU STRUCTURE

The repetitive and regular structure of digital signal processing programs led us to propose an adequate structure to repeat rapidly (without branch instructions) a block of compact code. An initialization instruction declares the loop by loading three counters : LCI (loop length 4 bit), LCR (N times 8 bit), and LCD (Delay 3 bit). The delay counter is useful in pipelined architectures where the instructions preceding the loop may be different from those inside the loop.

No interruption is provided in the PSI since the polling concept allows to synchronize programs rapidly with external regular clocks (Figure 3). Interruption results often in time loss for context switching operations. Besides interruption in a highly pipelined architecture needs more hardware to be controlled properly.

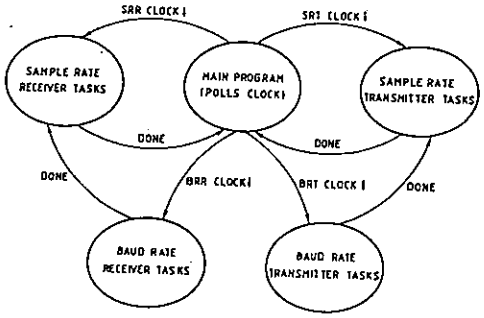


FIGURE 3 - MODEM TASK MONITOR

2.2. External architecture

The PSI is provided with three external 8 bit buses. An 8 bit access (AD0-AD7) called system bus is wired to a higher level processor while the two others,named local buses can be wired to lower level processors or peripherals. So a hierachical arborescent multiprocessor architecture can be built directly without any external circuit. The lower level processor has an integrated mail-box that can be accessed by the master processor without interruption. An external flag ($\overline{IRQ}$) indicutes to the higher level processor that the mail-box is available.

This flag has to be wired to a level sensitive branch condition input of the higher level processor (figure 4). Each processor can be wired to an external data memory at the same speed as internal memories. In this case the local buses must be defined as two concatenated buses to make a 16 bits access.

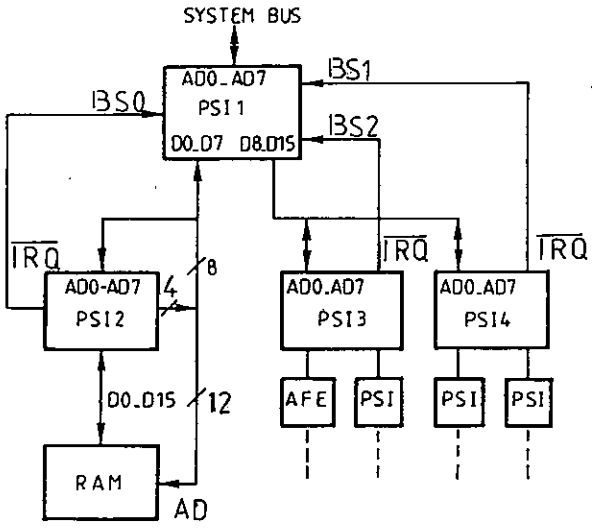


FIGURE 4 - MULTIPROCESSOR ARCHITECTURE

3. INSTRUCTION SET

All the instructions are 32 bit wide and are executed in one 160 ns single cycle under the real 16 bit mode and in one 320 ns double cycle under the real 32 bit mode or complex mode except for the branch instructions which always require 320 ns.

There are five arithmetic and move instruction formats, two branching instruction formats and one initialization instruction format. The first arithmetic format provides the following actions with indirect addressing all in one instruction cycle :

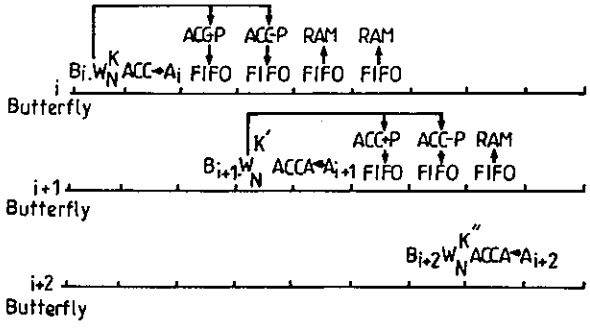
Fetch of operand on L BUS
Fetch of operand on R BUS
Individual multiplier operands refresh (M,N)
Designation of sources of ALU
ALU operation code
Destination of ALU output (FIFO, A, B, NONE)
Designation of source of Z BUS
Destination of Z BUS data
Individual post modification of addresses in the three internal address computation units.

The other arithmetic instruction formats are derived from the precedent one and offer other types of addressing (immediate for OPIM and direct for OPDI) operation codes (barrel shifter for shift operations) and a special move instruction (SVR) to save all address registers. The two branching instruction formats allow respectively direct and indirect or calculed conditional branching.

The initialization insctruction format programs dynamically several functions of the PSI in on 160 ns cycle. Among the main functions are :

- the hardware computational mode,
- automatic loop sequencer,
- address calculating units,
- access mode on external buses to assure full compatibility with an external RAM or peripheral device (slow/fast, 16 bit/8 bit x 2, AD0-AD7 use, RAM/PSI timing).

Table I gives the benchmarks for standard algorithms. As an example FFT butterflies are pipelined to achieve three complex cycles per butterfly.



FUNCTION	EXECUTION TIME
Fixed real FIR	160 nsec/TAP
Fixed complex FIR	320 nsec/TAP
Adaptive real FIR	320 nsec/TAP
Adaptive complex FIR	640 nsec/TAP
BIQUAD Cell real	960 nsec
BIQUAD Cell complex	1920 nsec
Treillis Form	480 nsec/Cell
Exp (jØ) (16 bit precision)	2720 nsec
FFT Butterfly	960 nsec
16 points complex FFT	38 µsec
64 points complex FFT	265 µsec

TABLE I - BENCHMARKS

4. SYSTEM ARCHITECTURE

One important feature of the PSI is it's efficient system architecture. Figure 4 shows a multiprocessor architecture. This signal processing machine can be wired as a peripheral function on the standard bus of a microprocessor system.

This microprocessor executes control tasks such as data processing and monitoring operations. An analog Front End designed for the application integrates the necessary interface circuits.

In speech applications a specific AFE circuit coupled with a single processor realize a speech coding application. An external memory can be added directly for a speech echo canceller requiring more precision for the coefficients.

For modem applications the AFE integrating suitable interface for echo cancellation has been developped.

5. CONCLUSIONS

A VLSI signal processor "PSI" is described. The highly parallel internal architecture of the PSI fits the inherent parallelism of the algorithms. This architecture is optimized to a set of kernel algorithms currently used in signal processing applications. The open architecture of the PSI allows easy extension of internal technology limited resources by built-in multiprocessor architecture and external data RAM addressing capability. This VLSI has been realized in a 2 micron NMOS process.

The external clock frequency is 25 MHz which provides a 160 nsec instruction cycle in the real mode. The device has a total of 120.000 transistors on a silicon are a of 57 mm^2.

KEY FEATURES

THREE 16 BIT INTERNAL DATA BUSES L, R,Z
160 NSEC INSTRUCTION CYCLE
32 BIT WIDE INSTRUCTION BUS
THREE DEDICATED ADDRESS COMPUTATION UNITS
TWO SEPARATE INTERNAL RAM MEMORIES 2X128X16
EXTERNAL MEMORY EXTENTION UP TO 4KX16
INTERNAL DATA ROM 512X16
120 000 TRANSISTORS ON 57 mm^2 IN A 2 μ NMOS PROCESS

COST EFFECTIVE MASKED VERSION (48 PIN DIL) TS68930 WITH 1280X32 PROGRAM ROM
ROMLESS VERSION TS68931 (84 PIN CC) CAN ADDRESS UP TO 64^KX32 BIT EXTERNAL PROGRAM ROM

REFERENCES

[1] S.S.MAGAR et AL "A Microcomputer with Digital Signal Processing capability", ISSCC 82

[2] T.TSUDA et AL "A High Performance LSI Digital Signal Processor for communication" ICC 83

[3] S.S.MAGAR et AL "An NMOS Digital Signal Processor with Multiprocessing Capability" ISSCC 85

[4] GUIDOUX and B.PEUCH, "Binary Passband Echo Canceller in a 4 800 bit/s 2 Wire Duplex Modem" selected areas in communications IEEE Vol. SAC-2, no5, sept 1984.

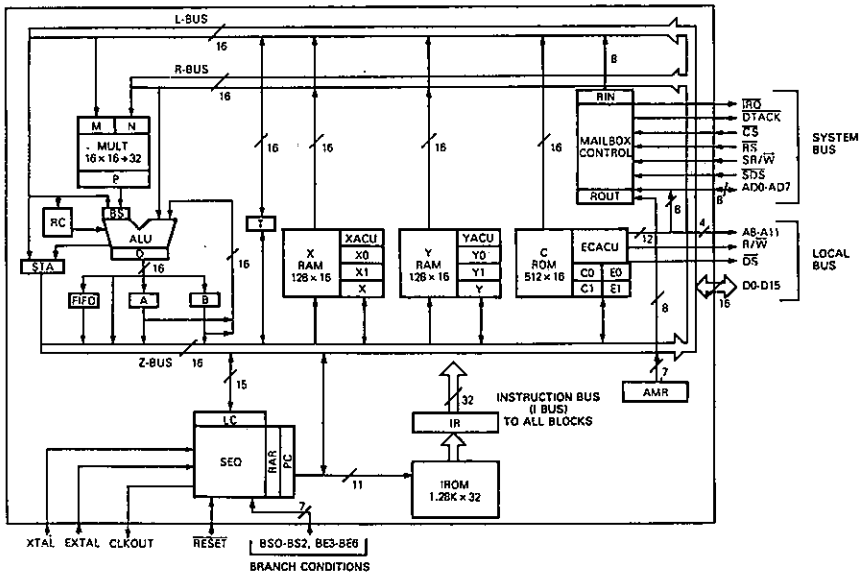


FIGURE 5 - INTERNAL ARCHITECTURE

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

On the Optimization of Pipelined Silicon CORDIC Algorithm*

Jichun Bu, Ed F.A. Deprettere and Fons de Lange

Delft University of Technology
Department of Electrical Engineering
Mekelweg 4
2628 CD Delft, The Netherlands

ABSTRACT

In this paper we discuss some aspects of the optimization of the silicon CORDIC algorithm. First, some constraints that are believed to be optimal for the VLSI implementation of the algorithm are discussed. The set of constraints turns out to provide an optimization problem that is NP-complete. Instead of conducting an exhaustive search, we have searched for satisfactory sub-optimal solutions only. Based on these solutions, a pipelined CORDIC processor architecture, which provides full 16 bits accuracy in fixed point arithmetic, is discussed. Three error sources are identified in the implementation : (1) the error due to the quantization of the angle parameters, (2) the error due to the finite word length effects and (3) the error due to the finite precision of scaling factors. The resolution of the angles determine the computational precision. The other two sources of errors can be eliminated by extending the internal data path width. The contractivity of the circular rotation can be guaranteed by a norm contraction step at the end of the pipe. The CORDIC processor is modeled and simulated in software and works correctly.

1. Introduction

The CORDIC computation technique was first introduced by Volder[1], who proposed a bit-recursive algorithm for fixed-point execution of the Givens plane rotation :

$$A(\theta) = \begin{vmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{vmatrix}, \quad |\theta| \leq \pi.$$

Later on, Walter [2] generalized this technique to norm preserving operators in other coordinate systems as well:

$$R_m(\alpha) = \begin{vmatrix} cos(m^{\frac{1}{2}}\alpha) & -m^{\frac{1}{2}}sin(m^{\frac{1}{2}}\alpha) \\ m^{-\frac{1}{2}}sin(m^{\frac{1}{2}}\alpha) & cos(m^{\frac{1}{2}}\alpha) \end{vmatrix}$$

where m is either 1, 0 or -1, representing circular, linear or hyperbolic coordinate systems respectively.

Early works on single chip implementation of the CORDIC iteration algorithm can be found in [3,4,5]. The basic idea is to execute a sequence of shift-and-add operations, each realizing a primitive CORDIC operation, called elementary unnormalized rotation:

$$R_m(\sigma_j,\alpha_j) = \begin{vmatrix} 1 & -m^{\frac{1}{2}}tan(\sigma_j m^{\frac{1}{2}}\alpha_{m,j}) \\ m^{-\frac{1}{2}}tan(\sigma_j m^{\frac{1}{2}}\alpha_{m,j}) & 1 \end{vmatrix}$$

where $\alpha_{m,j}$ are m-dependent positive constant, called base angles and $\sigma_j = \pm 1$. The base angles for each m should satisfy the inequality $|\alpha - \sum_{j=0}^{p} \sigma_j \alpha_{m,j}| \leq \alpha_{m,p}$ for any $\alpha \in D_m$, where in our case, $D_m = [-\pi, \pi]$. $\alpha_{m,p}$ is the smallest angle and hence determines the angle resolution. The result of the $p+1$ elementary rotations has to be scaled by a factor K_m which is a function of the basis angles: $K_m = \prod_{j=0}^{p} k_{m,j} = \prod_{j=0}^{p} cos(m^{\frac{1}{2}}\alpha_{m,j})$.

Recent work [5] on pipelined CORDIC processor design and usage shows that a CORDIC-module is a good candidate to be used as a processor element (PE) for modern systolic/wavefront implementation of solvers for systems of equations and for various signal processing algorithms. A few of such algorithms are described in [7,8]. For real time applications, arrays with very high throughput are required. For example, in the RPE speech coding algorithm [10], the CORDIC processor can be used to solve coupled sets of least-square fitting problems in real time [7]. Since a pipe-

* This work has been supported by the Dutch National Applied Science Foundation under Grant STW DEL 44.0643.

line CORDIC is rather costly in terms of power dissipation and silicon area, one would like to impose a number of optimization constraints as follows:

1. in order to minimize the chip area and the latency of the pipeline, the number of pipeline cells should be as small as possible and independent of the parameter m.
2. the signal propagation time in any of the basic cells realizing the unnormalized rotations have to be independent of the parameter m, the "shift indices" (see later) and the cell architecture.
3. the computational accuracy should be independent of the parameter m.
4. the contractivity of the circular rotation should be guaranteed.

A design of a silicon CORDIC algorithm that satisfys all these constraints is accomplished in three steps: (1) translate the imposed constraints into an optimization problem in terms of the CORDIC parameters and solve this optimization problem; (2) design a VLSI architecture with the optimized parameters; (3) simulate the obtained processor. The simulation is not only necessary to show that the hardware will operate correctly, but also to show that the required numerical accuracy is obtained.

In the next section, we will formalize the optimization problem in terms of the CORDIC parameters. A suboptimal solution is presented. Then, a VLSI architecture is described in section 3. Section 4 analyzes the numerical aspects of the algorithm. The last section gives the concluding remarks. Also, the evaluation of a simulation of the described CORDIC architecture is discussed.

2. An optimal solution for pipelined architecture

In order to satisfy the optimality conditions described in the previous section, we impose the following constraints on the base angles and the scaling factors:

1. for any $\alpha \in D_m$, the base angles have to satisfy the inequality : $|\alpha - \sum_{j=0}^{p} \sigma_j \alpha_{m,j}| \leqslant \alpha_p$, for $m = \pm 1,0$ and $\sigma_j = \pm 1$, where α_p is the angle resolution common to all three coordinate systems.
2. the basis angles should satisfy the convergency condition $\alpha_{m,k} \leqslant \sum_{j=p}^{k-1} \alpha_{m,j} + \alpha_p$. Notice that this condition is weaker than the condition $\frac{1}{2}\alpha_{m,k-1} \leqslant \alpha_{m,k} \leqslant \alpha_{m,k-1}$ which guarantees the fastest convergence.
3. $m^{-\frac{1}{2}} \tan(m^{\frac{1}{2}}\alpha_{m,j}) = 2^{-S_j} + \eta_{m,j} 2^{-S'_j}$, $\eta_{m,j} = -1,0$ or 1.
4. the scaling factor $K_m = \prod_{j=0}^{p} \cos(m^{\frac{1}{2}}\alpha_{m,j}) = 2^{-S(m)}$, with typical values for $S(m)$: $S(-1) = -2$, $S(0) = 0$ and $S(1) = 1$.

TABLE 1

The CORDIC parameters (m)				
cell nr.	$S,S',\theta(-1)$	$S,S',\theta(1)$	basis angles (−1)	basis angles (1)
0	0 4 -1	0 3 +1	1.716994	$\pm\pi/2 \pm 0.844154$
1	1 10 -1	1 10 +1	0.550609	0.464429
2	1 6 -1	1 6 +1	0.528685	0.476069
3	2 7 +1	2 7 +1	0.263764	0.252318
4	3 12 +1	3 12 -1	0.125905	0.124115
5	3 9 -1	3 9 +1	0.123674	0.126278
6	4 12 -1	4 12 -1	0.062336	0.062176
7	5	5	0.031260	0.031240
8	6	6	0.015626	0.015624
9	7	7	0.007813	0.007812
10	8	8	0.003906	0.003906
11	9	9	0.001953	0.001953
12	10	10	0.000977	0.000977
13	11	11	0.000488	0.000488
14	12	12	0.000244	0.000244
15	13	13	0.000122	0.000122
16	14	14	0.000061	0.000061
$K_{-1} = 2^2(1-2^{-20}) \approx 4$				
$K_1 = 2^{-1}(1+2^{-12}+2^{-20}) \approx 2^{-1}(1+2^{-12})$				

From now on we omit the trivial case m=0. It can be shown [6] that the above optimization problem is NP-complete for fixed-point implementations. We have found many solutions for either of the coordinate system m=1 or m=−1, but only an exhaustive search can reveal whether both solution spaces have an intersection. We have not conducted such an exhaustive search, but we have found solutions which almost satisfy the constraints and that are satisfactory anyway. For example, in the case of 16-bit implementation, there exists at least one general purpose solution [5] that satisfies all the above conditions, except that $S(-1) = 4[1 + \sum_{i=1}^{2} \sigma_i 2^{-S'_i}]$ instead of 4. In table 1 we present another solution, where K_{-1} is equal to 4 and K_1 is of the form $2^{-1}[1+2^{-s}]$. Only one extra addition is needed, although the top-most slice is slightly dependent on m. A VLSI architecture designed with the optimized parameters in table 1 will be described in the next section.

3. Implementation of a pipelined CORDIC processor with full 16-bit accuracy

There are basically two types of cells. The first one (type-I cell) realizes the following micro-rotation:

$$\begin{cases} x_{i+1} = x_i + (-m\sigma_i)2^{-S_i} y_i \\ y_{i+1} = (\sigma_i)2^{-S_i} x_i + y_i \end{cases}$$

The second cell (type-II cell) realizes the micro-rotation:

$$\begin{cases} x_{i+1} = (x_i + (-m\sigma_i)2^{-S_i} y_i) + (-m\sigma_i \eta_{m,i})2^{-S'_i} y_i \\ y_{i+1} = ((\sigma_i)2^{-S_i} x_i + y_i) + (\sigma_i \eta_{m,i})2^{-S'_i} x_i \end{cases}$$

A pipelined CORDIC consists of a chain of such parameterized cells. The values of the parameters are taken from table 1.

We use two's complement arithmetic. A combination

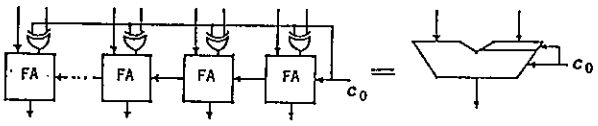


Figure 1 the ADD/SUB device.

of a n-bit full adder and n exclusive-or ports, called ADD/SUB device, is used to perform addition and subtraction on n-bit numbers (see Figure 1). For the ADD/SUB device, we code the combinations $(-m\sigma_i)$, (σ_i), $(-m\sigma_i\eta_{m,i})$ and $(\sigma_i\eta_{m,i})$ with binary 1 if they are negative (operation SUB) and with binary 0 if they are positive (operation ADD). The relation between the carry-in signal c_0 and these parameter combinations are given in the Table 2 (θ denotes bit-wise exclusive-or operation).

TABLE 2

combination	c_0
$-m\sigma_i$	$\overline{\sigma_i \theta m}$
σ_i	σ_i
$-m\sigma_i\eta_{m,i}$	$\overline{m \theta \sigma_i \theta \eta_{m,i}}$
$\sigma_i\eta_{m,i}$	$\sigma_i \theta \eta_{m,i}$

TABLE 3

α	β	$\alpha x+\beta y$	operation	c_0
+1	+1	x+y	$x+y+c_0$	0
+1	-1	x-y	$x+\bar{y}+c_0$	1
-1	+1	-x+y	$\bar{x}+y+c_0$	1
-1	-1	-x-y	$\bar{x}+\bar{y}+c_0$	2

The angle parameter is coded as a σ–*strings* [5]. In a given coordinate system, the CORDIC has two operation modes, rotation and vectoring. The rotation operation rotates a given vector over the specified angle, while the vectoring operation calculates the argument and the norm of a given vector. The block diagrams of the type-I and type-II cells are depicted in the Figure 2 and Figure 3, respectively. Only the control circuitry for the rotation operation is shown in the figures. It is quite easy to include the control for vectoring with little additional hardware (see Figure 4). In this case, the parameter σ_i is equal to the complement of the sign bit of the y input.

As shown in table 1, the first cell implements the rotations $\sigma\pi/2$ and $\sigma_{1,0}\alpha_{1,0}$ when the coordinate system is circular. Its operation can be described by the following equation :

$$\begin{cases} x' = -\sigma y + (-\sigma_{1,0}\sigma)(2^{-s_0}+\eta_{1,0}2^{-s'_0})x \\ y' = -\sigma_{1,0}\sigma(2^{-s_0}+\eta_{1,0}2^{-s'_0})y + \sigma x \end{cases}$$

In this case, we need to calculate $\alpha x+\beta y, \alpha,\beta = \pm 1$. A summary of the operations in two's complement arithmetic is given in Table 3.

We cannot directly use the ADD/SUB device given in Figure 1 to calculate $\alpha x+\beta y$,since when α and β are both equal to -1, we have a carry-in signal of weight 2. We overcome this problem by using modified full adders in the ADD/SUB device, while keeping the total

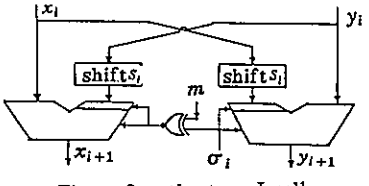


Figure 2 the type-I cell

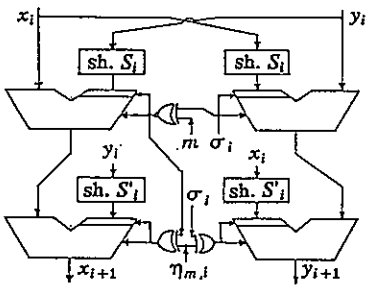


Figure 3 the type-II cell

propagation delay of the carry signal the same. The logical circuitry of the modified full adder cell is shown in Figure 5.

The CORDIC pipeline is constructed by a cascade of type-I and type-II cells. The last cell in the pipe is used to (1) scale the output with $K_1=\frac{1}{2}(1+2^{-12})$ in case of circular rotation; (2) scale the output with $K_{-1} = 4$ in case of hyperbolic rotation; (3) guarantee contractivity in case of circular rotation.

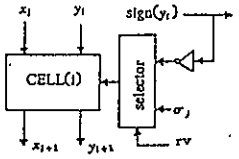


Figure 5.

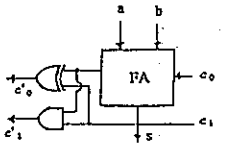


Figure 6.

4. Computational accuracy

In the fixed point CORDIC implementation described in the previous section, we identify three sources of errors :

1. quantization of the angle parameters (step size α_p);
2. the finite word-length; In our case, truncation of the shifted quantities.
3. the finite precision of the scaling factor K_m $(m = \pm 1)$.

In the implementation, the angle resolution α_p is equal to 2^{-14}. In the worst case, this angle will introduce an error $|E_{max}| = R\alpha_p$ in one of the components x or y, where R is the norm of the vector. For example, when the vector (1,0) is rotated over an angle $\pi/2$ in the circular coordinate system, the x component of the resulting vector will not be exactly equal to zero, but will stay within the interval $[-R\alpha_p, R\alpha_p]$, even when the rotation is carried out with infinite precision. In this case, $R\alpha_p$ will affect the LSB bit. The same is true for the hyperbolic rotation. Due to this fact, we require that the total error due to the finite precision arithmetic and truncation at each iteration step may not affect any of the 16 significant bits any more. To satisfy this requirement, we have to compute internally with higher precision. In fact, we have to extend the internal data path width at both the LSB side and the MSB side.

Extension of the data path width at the MSB side is necessary to avoid overflow. Since K_1 is equal to 1/2 , we have to extend the data path width with at least 1 bit at the MSB side. In case of hyperbolic rotation, the overflow could be very large since :

$$x' = cosh(a)*x + sinh(a)*y$$

$$y' = cosh(a)*y + sinh(a)*x$$

or,

$$\begin{aligned} |x'| &= |\ cosh(a)*x\ + sinh(a)*y\ | \\ &\leqslant |\ cosh(a)*x\ | + |\ sinh(a)*y\ | \\ &\leqslant |\ cosh(a)\ |\ (\ |x| + |tanh(a)|*|y|\ ) \end{aligned}$$

suppose $|y| < |x| = MAX$, and a is equal to π. Since $|\tanh(a)| < 1$:

$$|x'| \leqslant 2*|cosh(\pi)|*MAX < 2^5*MAX$$

And similarly for $|y'|$.

Extension of the data path width at the LSB side is necessary to eliminate truncation errors. Suppose that internally, the weight of the LSB bit is 2^{-M}. The two's complement truncation error is bounded by $-2^{-M} < E_T \leqslant 0$, where $E_T = Tr[x] - x$ for both the positive and negative numbers. The symbol $Tr[.]$ denotes the truncation operator. The truncation will take place at each shift operation. Since there are 7 type-II cells and 10 type-I cells, we will have 7*2 + 10 = 24 truncations in total which will give at most 5 bits truncation error accumulation. Since the first two extended bits at the LSB side are significant for hyperbolic rotation ($K_{-1} = 4$), we have to extend the internal data path at least 7 bits at the LSB side; Due to the error in the scaling factors, we need one bit more to eliminate this error. The internal data path width is thus 1+16+7+1 = 25 bits.

5. Concluding Remarks

The pipelined CORDIC processor is a promising candidate to be used as a PE in dedicated systolic/wavefront systems. It has a wide range of applications in real time digital signal processing systems, from digital filtering to matrix solvers. The described pipelined CORDIC architecture is optimal with respect to the imposed constraints. In this architecture, the control circuitry is kept minimal. The throughput is determined by the carry-propagation time from the LSB to the MSB in the ADD/SUB device. The complexity of this silicon algorithm is of order $O(n^2)$. Although, we can reduce it to $O(n)$ by using redundant binary arithmetic [11] instead of two's complement.

During the design, we have developed a small library of parameterized routines, emulating the hardware modules such as NAND, EXCLUSIVE-OR, and FULL-ADDER etc. The CORDIC architecture is simulated with these software modules as basic building blocks and it works correctly.

The numerical behavior of the described CORDIC architecture is also studied [9] with this software. We have simulated the systems of linear equation solver, described in [7], in which the CORDIC modules are used as PE's. In the simulation, this matrix solver is used as a part of the RPE speech coding algorithm [10]. The simulation data are sampled from a real speech signal. First, the simulation showed that 16 bits were insufficient to represent the signals due to the large dynamic of the speech signal. After the extension of the data path width, we obtained fairly satisfactory results, both numerically and subjectively (listening tests). From this simulation, we decided to implement this processor as a 32 bits PE and as a floating point PE. The design of the later is described in [9].

6. References

[1] Volder, J.E.,"The Cordic Trigonometric Computing Technique," IRE Trans. Electronic Computers Vol. EC-8(3) pp.330-340(1959).

[2] Walter, J.S., "An Unified Algorithm for Elementary Functions," Proceedings Spring Joint Computer Conference Vol. 38 pp. 397 AFIPS press,(1971).

[3] Haviland, G.L. and A.A. Tuszynski, "A CORDIC Arithmetic Processor Chip," IEEE trans. Computers Vol. C-29(2) pp. 68-79 (1970).

[4] Ahmed, H.M. and Morf, "Synthesis and Control of Signal Processing Architectures based on rotations," pp. 43-52 in VLSI 81, ed. J.P. Gray, Academic Press (1981).

[5] Ed. F.Deprettere, P.Dewilde and R.Udo "Pipelined CORDIC Architectures for Fast VLSI Filtering and Array Processing," Proc. ICASSP-84, pp. 41.A.6.1-41.A.6.4(1984).

[6] Christos H. Papadimitriou, Kenneth Steiglitz "Combinatorial Optimization: Algorithms and Complexity," Prentice-Hall, Inc. 1982.

[7] E.Deprettere and K.Jainandunsing, "Design and VLSI Implementation of a Concurrent Solver for N Coupled Least-Squares Fitting Problems," IEEE journal on SELECTED AREAS IN COMMUNICATIONS, pp.39-48, Jan. 1986.

[8] K.Jainandunsing and E.Deprettere, "Solving Sets of Linear Equations for Real Time Signal Processing," To appear in EUSIPCO-86 proceeding.

[9] F.de Lange, "Register Level Simulation of a 16-bits Pipelined CORDIC Chip," Internal report, Delft Univ. of Techn., Dept. of Electr. Eng. (April 1986).

[10] E. Deprettere and P. Kroon, "Regular excitation reduction for effective and efficient LP-coding of Speech," Proc. ICASSP'85, Mar. 1985.

[11] J. C. Bu, H. X. Lin, Ed. F.Deprettere, P.Dewilde, "A Fast VLSI CORDIC Implementation Using Redundant Binary Arithmetic," to be published.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

R.5

THE VLSI REALISATION OF A BINARY-IMAGE PROCESSOR

M.A. Kraaijveld, P.P. Jonker, R. Nouta, R.P.W. Duin.

Network Theory Section,
Department of Electrical Engineering,
Delft University of Technology,
Delft, The Netherlands.

This paper discusses the design of a fast pipelined pixel processing chip to perform cellular logic operations on binary images. The chip has a complexity of about 20,000 transistors and will be made in the NMOS laboratory of the Department of Electrical Engineering. The flexible architecture facilitates its use in many different environments, varying from very small and compact, to very large and powerful systems. Operating on a clock frequency of 10 MHz, the chip requires 100 ns to process a pixel. In a pipeline of N processors, there are N operations executed in 100 ns.

Keywords: VLSI design, cellular logic operations, binary image processing, pipelined processors, special architectures.

1. INTRODUCTION

For the processing of binary pictures in image processing applications, a special class of operations has been developed: the cellular logic operations [1]. Cellular logic operations are local binary transformations, the transformed value of a pixel being determined by the value of the pixel itself and its 8 neighbouring pixels.

Based on the experience with the Delft Image Processor, the DIP-1 [2,3], the Pattern Recognition Group of the Applied Physics Department, Delft University of Technology, has developed a (VME) processor board for cellular logic operations [4]. This Cellular Logic Processor (the CLP), operates as a part of a stand alone image processing system. The CLP is able to process binary pictures of 256 by 256 pixels in about 6.5 ms per processing cycle.

This paper described the development of a VLSI realisation of the CLP [5]. Taken into account the possibilities that a VLSI realisation offers and the whished and demands for the new generation, the profile of the CLP chip was sketched during discussions with experts of the "Centre for Image Processing Delft". These sketches were worked out in a new design and a new definition of the environment of the CLP. To verify the correctness of the new design, the circuit was simulated on register transfer level in APS and on switch level in SLS.

2. PROCESSOR ARCHITECTURE

To handle the complexity of the design, the concept of a hierarchical model of virtual machines was used [6]; see fig. 1. In the top of the hierarchy we find the controller, controlling the serial interface and the process module.

- The serial interface is used for the communication with the host computer: it fills the instruction registers and reads the status registers of the processor.
- The process module does the actual cellular logic operation: it processes the image information in the bitplanes with the help of a set of Hit or Miss masks stored in a PLA.

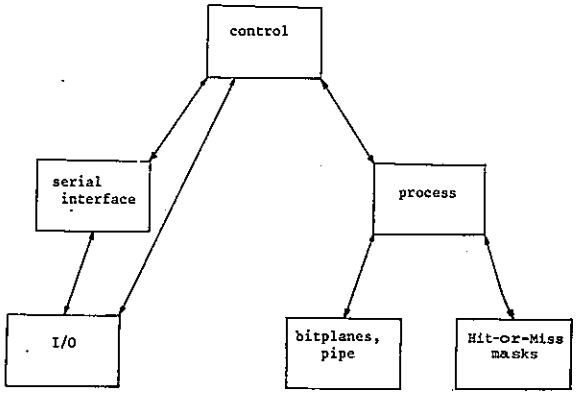


Figure 1.

2.1 The datapath

The datapath of the process module of the Cellular Logic Processor chip is sketched in figure 3. The central part consists of the shift register section. An input image is shifted serially into the shift registers, which are drawn off at the right places to generate the 3 by 3 neighbourhood of a pixel. Shift registers of variable length offer the possibility to process images of sizes in powers of 2, up to 512 pixels in 1 direction and 16 k pixels in the other. External shift registers can be connected to the processor to process images up to 16 k in both directions. The processor is provided with 4 bitplane inputs. With the help of 2 boolean function generators, the image that is shifted into the shiftregister section can be made of a logical combination of 3 arbitrary input images.

The 3 by 3 neighbourhood of a pixel is processed in a PLA to determine the new value of the central pixel The PLA contains sets of "Hit or Miss masks" for each cellular logic operation [7]. The "erosion 4-connected" and "contour 4-connected" operations, for example, are implemented as sets of the following masks:

```
. 1 .    . . .    . 0 .    . . .    . . .
1 1 1    . 1 0    . 1 .    0 1 .    . 1 .
. 1 .    . . .    . . .    . . .    . 0 .
```

E4-set CT4-set

Figure 2.

Whereas a 1 indicates a Hit, a 0 a Miss and a . don't care. The central pixel is set if one of the masks fits (i.e. the 0's and 1's of the mask correspond with the generated 3 by 3 neighbourhood). Each mask is stored as a product term in the PLA. Among the built in operations are erosion, dilation, contour extraction, skeletonization and others. A writable PLA (WPLA) is integrated for the implementation of newly developed algorithms or special applications.

The CLP chip is very suitable to be used as a part of a processor pipeline and has therefore 3 data outputs: the resulting image, the original input image and the mask image. With the help of the boolean function generators, a processor in the pipeline can perform an operation on a logical combination of the result, the original and the mask of its predecessor in the pipe. When a pipe of processors is used, the processing power is increased with a factor that is equal to the number of processors in the pipe. E.g. when the CLP would be used in a pipe of 50 processors the system would be able to execute 50 cellular logic operations on a 256 by 256 grid in about 6.5 ms.

2.2 Special features

There are some interesting features of the design to be mentioned here:

- A built-in syntax checker checks the syntax of the instruction that is currently in the instruction registers. The result of the test appears in the status registers of the processor.
- The processor generates a (maskable) interrupt ot the host computer when an operation is finished.
- The 3 instruction registers are encoded in such a way that all simple instructions can be defined with 1 instruction register. Only the more complex instructions (which are expected to be used less), are defined with 2 or 3 instruction registers.

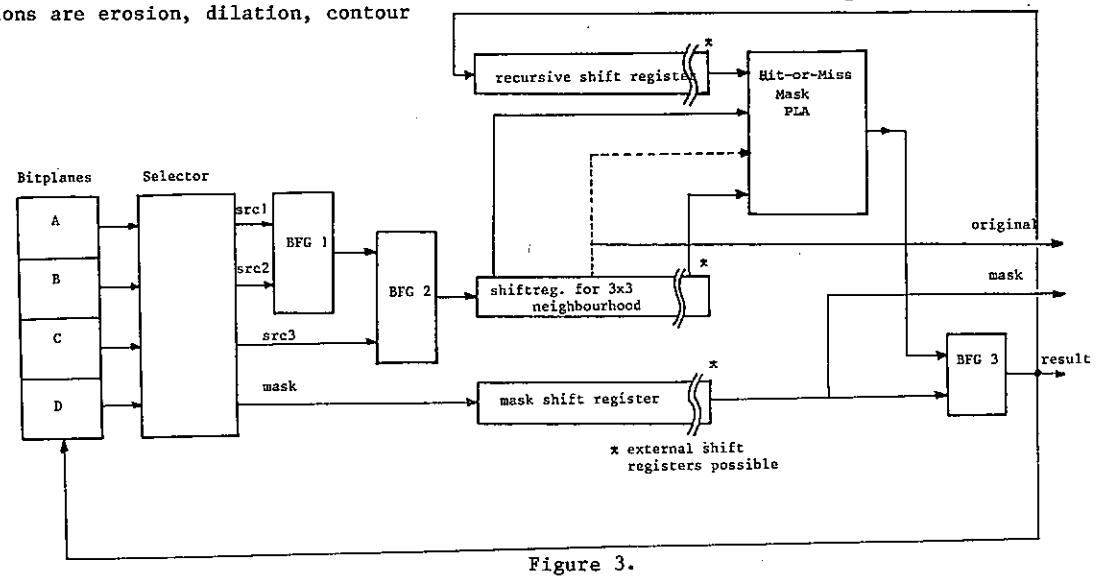


Figure 3.

- Two counters count the number of bits that are set in the resulting image, and the number of cycles that are required for object dependent operations. The value of these counters appear in the status registers of the processor.

3. SYSTEM ARCHITECTURES

Because of the limited number of available pins on the chip (i.e. 40), the CLP chip is equipped with a serial interface. When used on a PCB, e.g. in a VME environment, this serial interface is connected to the bus interface, which communicates in a parallel way with the host computer (fig. 4.).

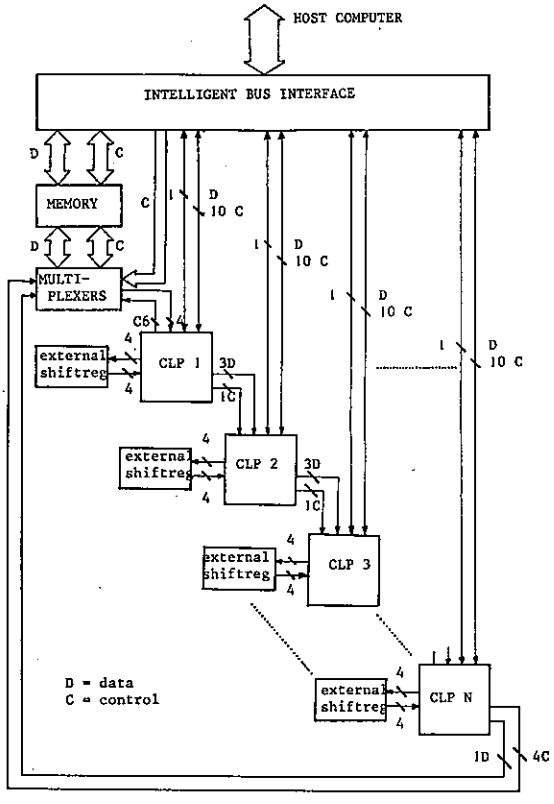


Figure 4.

The function of the bus interface in simple configurations is therefore:

- Accepting 16 bits parallel from the host computer. Transmitting these bits serially to the bitplanes or the processor.

- Serial reading of the status registers of the processor and the image data in the bitplanes. Transmitting the information in 16 bit words to the host computer.

The reason for this interface design is that in this way the bus interface can be tailored to the needs of the application. For example:

- The bus interface can be adapted to work with different bus systems.

- The interface can be made to handle a more complex memory section. E.g. several images per single bitplane.

- More than 4 bitplanes can be handled by using a multiplexer section.

- The bus interface can be adapted to work with a pipe of processors.

4. THE SIMULATIONS

4.1 The APS simulation

The design has been simulated on register transfer level in APS. APS is a version of ISPS that was rewritten in C by the Network Theory Section of the Technical University of Delft. There are two concepts integrated in the simulation:

- The hierarchical model.
  This model was essential to handle the complexity of the design.

- The model used by Mead and Conway [8].
  In this model data is transported in a pipeline of pass transistor arrays and logic, controlled by finite state machines. This implies that all (parallel) actions in the pipeline have to be simulated.

Feedback about the correctness of the design is provided by processing a test image. The actual simulator is therefore able to process a given input image. A post processor is used to convert the output of the simulator to a visually attractive format.

4.2 The SLS simulation

The switch level simulator (SLS) is a MOS circuit simulator that was developed by the Network Theory Section [9]. It is capable of simulating timing behaviour, behaviour of ratios, charge sharing, races and spikes. The switch level description is made by translating the routines that were defined in APS to transistor networks. Some switch level descriptions however, were made by extracting a layout. These layouts were made by student VLSI design groups or generated by a layout generating program.

5. CONCLUSIONS

The CLP chip will be able to work in many different environments, varying from very small and compact systems (a complete system in 12 chips) to very large and powerful systems (when used in a pipeline). Operating on a clock frequency of 10 MHz, the chip requires 100 ns to process a pixel (i.e. 6.5 ms for an image of 256 by 256 pixels). In a pipeline of N processors, there are N operations executed in 100 ns.
Because of the high processing speed and the flexibility to implement complicated algorithms, the CLP chip will be a powerful tool in image processing, pattern recognition and robotics. The chip has a complexity of about 20,000 transistors and will be made in the NMOS laboratory of the Department of Electrical Engineering.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude to all persons who contributed to the project. In particular, the significant contributions of the following persons are to be acknowledges: prof.dr.ir. P.M. Dewilde, ir. A.C. de Graaf, ir. T.G.R. van Leuken, ir. A.J. van Genderen, ir. R.J. van Munster, ir. R. Boekamp, B.G.M. de Lange.

REFERENCES

[1] I.T. Young, R.L. Peverini, P.W. Verbeek, P.J. van Otterloo, A new implementation for the Binary and Minkowsky operators, Computer Graphics and Image Processing 17, pp. 189-210, 1981.
[2] F.A. Gerritsen, Design and implementation of the Delft Image Processor DIP-1, Ph.D. thesis, Pattern Recognition Group, Applied Physics Department, Delft University of Technology, Delft, 1982.
[3] F.A. Gerritsen, L.G. Aardema, Design and use of DIP-1: a fast, flexible and dynamically microprogrammable pipelined image processor, Pattern Recognition, Vol. 14, Nos 1-6, pp. 319-330, 1981.
[4] R. Boekamp et al., Design and implementation of a cellular logic VME processor board, Proceedings of the SPIE 2nd Technical Symposium on Optical and Electro-optical Applied Science and Engineering, B596: Architectures & Algorithms for Digital Image Processing, Cannes 1985.
[5] M.A. Kraaijveld, The VLSI realisation of a binary-image processor, Department of Electrical Engineering, Delft University of Technology, Delft, 1986.
[6] G.A. Blaauw, Digital System Implementation, Prentice Hall, 1976.
[7] P.P. Jonker, R.P.W. Duin, Considerations on a VLSI architecture for Cellular Logic Operations, Proceedings of the IEEE workshop on Computer Architecture for Pattern Analysis and Image Database Management, Florida, USA, 1985.
[8] C.A. Mead, L.A. Conway, Introduction to VLSI systems, Addison-Wesley, 1980.
[9] P.M. Dewilde, A.J. van Genderen, A.C. de Graaf, Switch Level Timing Simulation, Proceedings of the ICCAD, 1985.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Compiling Silicon: From Software to Hardware

P. Dewilde, J. Annevelink, E. Deprettere
K. Jainandunsing

Department of Electrical Engineering
Delft University of Technology
Delft, the Netherlands

1. Introduction

Digital signal processing often requires very fast handling of data in a small physical area and/or with low power consumption. The most classical instance is that of bit-serial digital filters which are capable of achieving high throughputs with a very limited area, taking into account numerical considerations like stability and accuracy. In such very dedicated filters, the filtering algorithm used can be optimally adapted to the requirement of hardware minimization within the speed constraints. A very good example of this technique can be found in [1]. In the present paper we wish to take a rather more general point of view where we develop the design path from a general numerical algorithm down to hardware with the expressed goal of realizing a structure that is as regular as possible, however without impairing its performance. Our interest will go to rather general types of algorithms, e.g. a matrix solver or a system to compute eigenvalues, and we shall concentrate on the problems associated with mapping it on regular, dedicated hardware: hierarchical refinement, partioning and the generation of control structures. The emphasis will be on placing all the components of the design system in one consistent framework. As we shall see, the consistent framework is obtained by combining and modifying classical concepts to fit the general design situation. Foremost is the notion of *signal flow graph* which is classical in signal processing, but needs modification as will be explained further. Next are the notions of *functional calculus* and *applicative state transitions*, which are also classical to signal processing and has been rediscovered (of course with a different terminology) in computer science [2]. In the meantime both notions have obtained considerable attention especially in the computer science litterature [3], [4], [5]. On the other hand, the development of description languages for signal processing has also been considered - a prime example of which can be found in [6]. Needless to say, we have used freely many ideas already present in the litterature, whlile devoting most of our attention to the development of a consistent design system - the design system HIFI that was developed at Delft University over the last three years [7]. The first sections will be devoted to the description of the design methodology, while the latter sections will treat a specific example: a new type of matrix solver with special pipeline properties.

2. Refinements of System Design

One may define systems' design as the successive refinement from behavioral descriptions to structure, a process that finally ends in the complete definition of the hardware. This is especially true for the design of dedicated concurrent systems, like signal processors or matrix solvers as one has in speech coding, image coding or simulation [8,14]. Major problems with the design of concurrent systems are the assignment of tasks to processors and the partitioning of the algorithm. It often turns out that the complexity of the overall solution is much more dependent on the organization of the datatransport than on the efficient implementation of the computations themselves. Examples can be found in [12,14]

Refinements start out with what one may call the system semantics. At this level, the system is represented by a single node to which a behavioral description is attached in the form of what we have chosen to be an Applicative State Transition description, see next section. Out of that description, a node can be replaced by a signal flow graph (SFG), which makes the pipelined structure of the computations explicit together with the description of the states that are relevant for that level. The overall design procedure then goes as shown in fig.1 - for a description of the nodes (semantics) and the Signal Flow Graphs (structure), we refer to the next section.

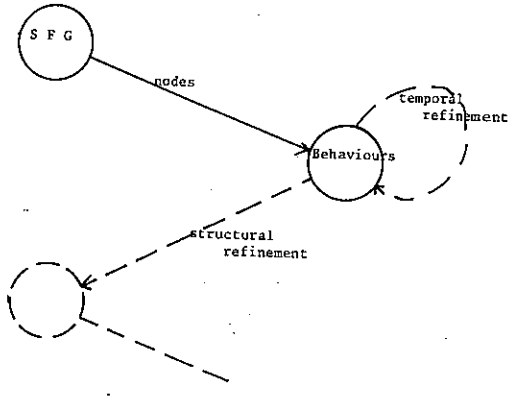


Figure .1. Hierarchical Design Procedure: from semantics to structure

3. Behavioral Description in Nodes

At a high enough level of abstraction, any deterministic system can be described by an I/O map in the form of "formal semantics" or equivalently, a behavioral description. Even at this level, the system has already some structure - it is reusable - which we catch by saying that it is an Applicative State Transition or AST system [2]. We represent it by a node, which is characterized by:

1. $F = \{f_\alpha\}$ - a set of functions which the node is able to execute (e.g. any internal state is captured in a functional way). Each f_α is a partial function: it maps some inputs to some outputs.
2. I a collection of input ports characterized by a (fixed) type.
3. O a collection of output ports also characterized by a (fixed) type.

In any of its histories, the node will execute a sequence of functions:

$$f_0 . f_1 . f_2 . \cdots$$

with each f_i in F. When f_i is executed, we say that we stand at "event i".
Let $V_I(i)$ be the values (ev. empty) of the Input ports at event i, and let $V_O(i)$ be the values (ev. empty) that the Output ports will obtain at event i, then the AST mechanism is characterized by the map:

$$f_i = Z * V_I \to V_O * F : (i, V_I(i)) \mapsto (V_O(i), f_{i+1})$$

It can be shown that the above AST description holds for any *deterministic R system.*

4. The Structure of Concurrent Systems

At any given level of the design hierarchy, the system structure is represented by a Signal Flow Graph (SFG) which makes functional relations and state that are relevant at this level explicit. The communication mechanisms used in our SFG's must be consistent with our previous AST description, and are as follows:

1. Self-timing by a single token pass discipline. For each node in the SFG, an actual partial function f_i may fire when the two following conditions are satisfied: (a) tokens are present on relevant inputs; (b) relevant outputs are free from tokens - see fig.2.

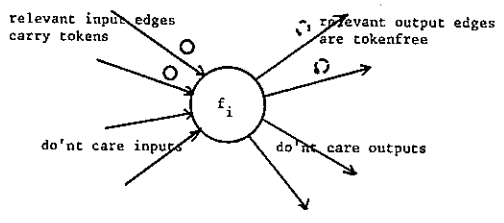


Figure 2. Node Firing Discipline

2. The state that is explicit for this level is to be put in registers that are marked as "Delays" and whose sole function is to make the registered data available to the next cycle (all f_α reduce to one function: $f : I \to 0 : V_I \mapsto V_O$ - fig.3.)

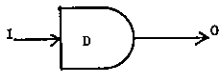


Figure 3. State Representation

3. Edges cannot accumulate data (there are no hidden states in the edges) nor can there be hidden states in the nodes, and the problem of side effects in the definition of the system is totally avoided.

This SFG model requires, however, timing verification because nodes are in no way synchronous, and mismatches between inputs and outputs are possible. We consider this a desired property, because it catches the relevant design problem at the present level of abstraction. It is easy to develop a theory of correctness for this situation based on edge trace theory, and it can be ascertained that an SFG which is trace correct is actually an AST machine. In this way the method proposed here is internally consistent.

5. Refinements

There are four types of possible refinements of a design, which are represented by the following diagram:

	temporal	spatial
general	Finite State Machine	Construction/Composition
regular	Clock	Array

Each of the possible refinements is shown as a node refinement in fig.4.

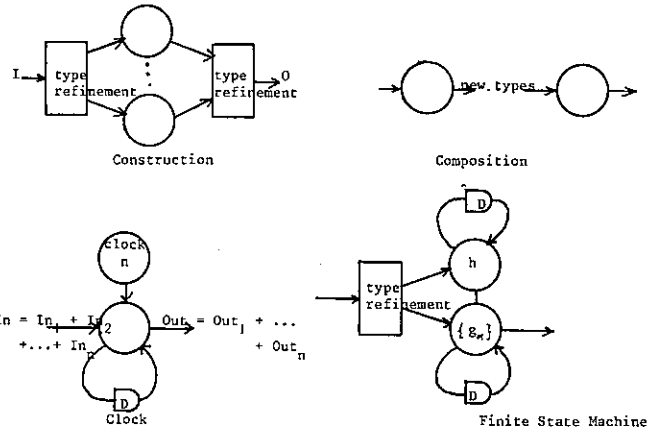


Figure 4. Possible node refinements

6. Solving Equations in a pipelined fashion

Classical algorithms for solving large sets of linear equations of the type A x = b compute the factorization of the matrix A to produce an upper triangular system which is then solved by a procedure called "backsubstitution". The resulting datatransport is very

unfavourable for parallel processing because the backsubstitution step needs the data in the reverse order than the factorization. In the following sections we present an algorithm which solves the system in one pass, thereby avoiding the backsubstitution step. The resulting algorithm does not require any intermediate accumulation of data, and is ideally suited for implementation on a dedicated array of processors. The paper also shows how the algorithm is mapped to the VLSI array, after further partitioning.

Given is a system of linear equations $Ax = b$ where the matrix A is n*n and typically a large bandmatrix (we shall treat the problem in full generality). The traditional method of solving the system is by factoring A as $A = QR$ where Q is a transformation matrix which we choose to be orthogonal for numerical accuracy and R is uppertriangular. If b is likewise transformed to $\beta = Q^t b$, then the system of equations is transformed to $Rx = \beta$ and x is found by backsubstituting on β. The latter operation starts with the last row in R, while the factorization produces the first row first. A conceptual architecture representing these operations is shown in fig 5.

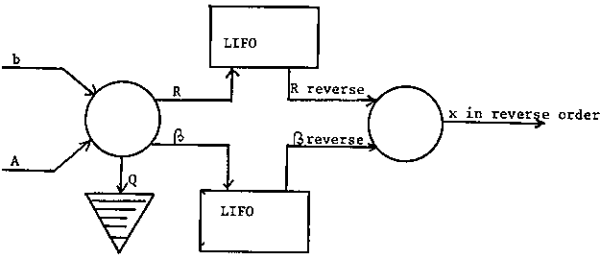


Figure 5. Architecture of the Classical Matrix Solver

By a clever arrangement of the data it is, however, possible to restrict the operations to factorization only. Inspired by the work of Faddeeva [10] who presented a Gaussian algorithm which incorporated the backsubstitution, we propose to factorize the matrix

$$A' = \begin{bmatrix} A^t & I \\ -b^t & 0 \end{bmatrix}$$

The factorization, with appropriate partitioning of the matrices, gives:

$$\begin{bmatrix} U_{11} & u_{12} \\ u^t{}_{21} & u_{22} \end{bmatrix} \cdot \begin{bmatrix} A^t & I & 0 \\ -b^t & 0 & 1 \end{bmatrix} = \begin{bmatrix} R' & U_{11} & u_{12} \\ 0 & x^t u_{22} & u_{22} \end{bmatrix}$$

The operations performed during the factorization follow the classical Householder algorithm for which we refer to [11] and are, for convenience summarized next:

1. On the first (block-)column of the matrix, "Vectoring" is done which transforms the block to an upper unit matrix.
2. All the following (block-)columns have to be processed by the first transformation.
3. The resulting second block can now be depleted of its top, and processed according to the same scheme.

The resulting parallel architecture, duly partitioned, and with the necessary datatransport and storage shown, is given in fig. 6

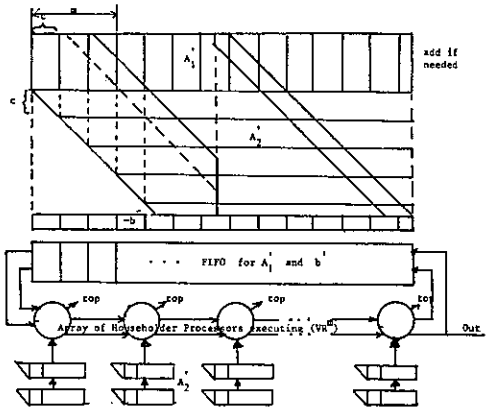


Figure 6. The Partitioned Architecture of the Single Pass Solver

In fig. 6 two methods of partitioning have been used, resulting in two partitioning parameters:

— the grouping of c consecutive columns of the A_{1sup}' and b', which we call LSGP for "local sequential, global parallel", and

— LPGS (local parallel, global sequential), whereby m parallel processors execute the first m cycles in parallel, but are then scheduled to execute the next m cycles.

Each of the processors executes a sequence $(VR^m)^*$, and consists of an Inner Product engine together with some control which takes care of the switch between vectoring and reflection, and of implementation of the different steps of the Householder algorithm. A further refinement of the design given so far is possible using the same HIFI methodology, which has been presented earlier.

7. Further refinement to local processing.

A further refinement can be obtained by decomposing the Householder nodes into an array of Givens nodes. Just as before, using partitioning of either the LSGP or LPGS type, new architectures are obtained with different storage properties as shown in fig. 7.

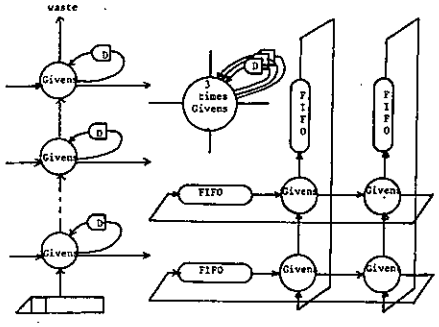


Figure 7. Refinement and Partitioning of the Householder node with Givens Processors.

Each of the processors shown in fig. 7 is of very simple type. Either it contains a complex multiplier, together with an iteration that performs the necessary square rooting and normalization or it contains a CORDIC processor which does all these operations at once and turns out to provide a more attractive solution. Again, each of these processors can be refined to the bit level, and converted into VLSI hardware.

8. Discussion

The methodology presented here is indeed capable of refining and transforming any deterministic algorithm into dedicated hardware in which space or timing constraints are satisfied. Although the present paper has given a rather loose description of the procedure, it has been formalized and is becoming available as a prototype design system. Two features are thereby of great importance: the ability to execute simulations at any level of the design hierarchy, and the ability to do the systems' definition both automatically and interactively. Both are being supported.

9. Bibliography

[1] J. van Genderen, H. de Man, F. Cathoor and S. Beckers, "A design methodology for compact integration of wave digital filters", Proc. 10th. Eur. Solid-State Circuits Conf. (Edinburgh, G.B., Sept. 1984), pp. 210-213.

[2] Backus, J., "Can Programming be Liberated from the Von Neumann Style? A Functional Style and its Algebra of Programs", Comm. of the ACM 21 (1978), 613-641.

[3] Milner, R., "Flowgraphs and Flow Algebras", J. of the ACM, Vol. 26, No. 4, Oct. 1979, pp. 794-818.

[4] A. Iizawa and T.L. Kunii, "Graph-Based Design Specification of Parallel Computation", Techn. Rept. Dept. of Info. Sci., Univ. of Tokyo, Dec. 1983.

[5] E.A. Ashcroft and W.W. Wadge, "Lucid A non procedural language with iteration", Comm. of the ACM., July 1977, pp. 519-526.

[6] P. le Guernic, A. Benveniste and T. Gautier, "Signal: un language pour le traitement de la parole", IRISA, Rennes, Publication Interne, no. 195, Mars 1983.

[7] Kung, S.Y., J. Annevelink and P. Dewilde, "Hierarchical Iterative Flow-Graph Design for VLSI Array Processors", IEEE International Conference on Computer Aided Design, Santa Clara, Califo., Nov. 1984.

[8] Kung, S.Y., H.J. Whitehouse and T. Kailath, Editors, "VLSI and Modern Signal Processing", Prentice Hall, 1985.

[9] Kung, S.Y., "On Supercomputing with Systolic/Wavefront Array Processors", Proceedings of the IEEE, Vol. 72, No. 7, (July 1984).

[10] V.N. Faddeeva, "Computational Methods in Linear Algebra", Dover Publications, New York, 1959, pp. 90-99.

[11] J.H. Wilkinson and C. Reinsch, "Linear Algebra", Springer Verlag, New York, 1971.

[12] H.T. Kung, B. Sproull and G. Steele, "VLSI Systems and Computations", Computer Science Press, 1981.

[13] K. Hwang and F.A. Briggs, "Computer Architecture and Parallel Processing", McGraw Hill, 1984.

[14] T. Kailath, "Modern Signal Processing", Springer Verlag, 1985.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ARCHITECTURAL USER ASPECTS OF THE SINGLE CHIP DIGITAL SIGNAL PROCESSOR PCB5010

K. Hellwig, P. Vary
Philips Kommunikations Industrie AG,
Nuernberg, W.-Germany

P. Anders
Valvo Applikationslabor,
Hamburg, W.-Germany

J.v. Meerbergen, R. Sluyter, F.v. Wijk
Philips Research Laboratories, Eindhoven, The Netherlands

The PCB 5010 is a new single chip digital signal processor (DSP) designed primarily for speech and audio processing. The CMOS chip is based on fixed point arithmetic with an instruction cycle time of 125 ns.
A detailed description of the highly parallel DSP architecture is given from the user's point of view and the implementation of complex algorithms is addressed.

1. INTRODUCTION

Due to the rapid advances of technology the application of digital signal processing is getting more and more attractive.
On one hand conventional analogue solutions are replaced by digital systems on the other hand completely new features can be implemented.
Because of the wide scale of applications and in view of the tremendous design effort for dedicated custom VLSI circuits, programmable digital signal processors (DSPs) seem to be the universal solution.
However, in designing this kind of DSP economical as well as technological constraints have to be considered. Thus a limited set of application requirements has to be covered efficiently using a limited chip area.

According to this premise the single chip DSP PCB5010 had been developed primarily for real time processing of speech and audio signals.

The architecture of this chip is presented from the user's point of view.
Benchmarks of key algorithms as well as complex applications in telecommunications are described.

2. INTERNAL ARCHITECTURE

The internal structure of the DSP is characterized by a high degree of parallelism according to a Harvard architecture with two independent databuses and a separate instruction bus as shown in the simplified blockdiagram of fig. 1.

The basic features are:

- Harvard structure with two 16 bit buses
- two's complement 16 x 16 multiplier with 40-bit accumulator and multi-precision operation support
- 16-bit two operand ALU with multi-precision support and division step
- Program memory: 1024 x 40 bit
  a) PCB 5010: RAM(37x40) + ROM(987x40)
  b) PCB 5011: external memory 1024x40 extension up to 64K x 40 possible
- Data memory: 512 x 16 bit on-chip ROM; 2 x 128 x 16 bit on-chip RAM; fast access to large external memories
- 3 independent address calculation units
- 16-bit parallel I/O to access external data memory
- 4 independent serial ports (2xIn, 2xOut)
- maskable interrupt
- 4 user input flags
- 5 level stack
- instruction cycle 125 ns or 250 ns according to 2 modes of execution:
  FAST: 125 ns ; pipeline visible
  NORMAL: 250 ns ; pipeline not visible
- 2 μ CMOS technology
- Package: 68 PLCC (PCB 5010)
  144 PGA (PCB 5011)

With a single instruction e.g. the following operations are carried out within 125 ns:

- accumulation of the previous product
- transfer of operand X
- transfer of operand Y
- calculation of X*Y
- address calculation RAM A
- address calculation RAM B
- address calculation ROM

For the program development the ROM-less version PCB 5011 with external program memory and data ROM is used.

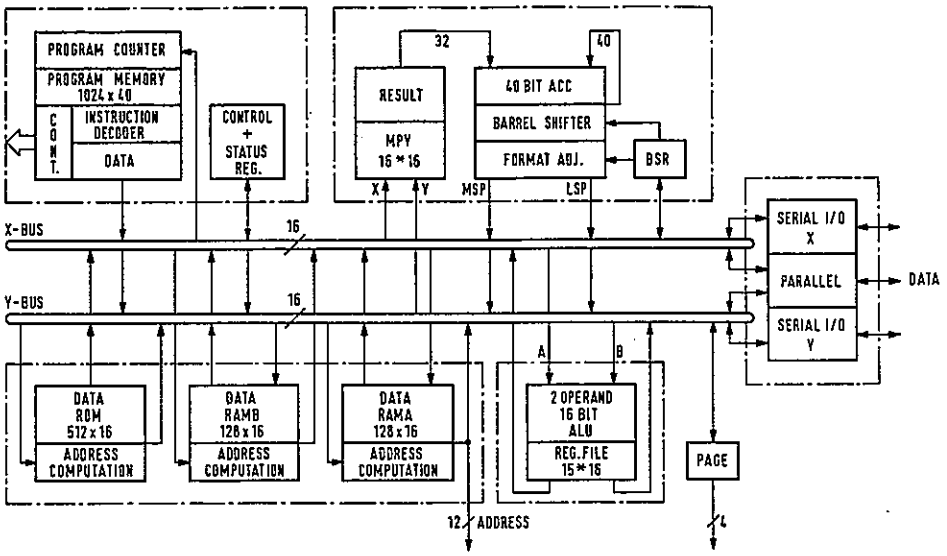


Figure 1: Block diagram PCB 5010

Multiplier-accumulator section:
The core of the processor is the multiplier-accumulator section which delivers a 32-bit product in 125 ns. Two latches at the input and some additional logic enhance the instructionset e.g. by optionally inverting and/or storing the operands.
The result is latched in a 32-bit pipeline register and fed to a 40-bit accumulator. In this way the summation of products can be done in parallel with full accuracy up to at least 255 terms without overflow. A 32-bit barrelshifter and a format adjuster at the output of the accumulator register can be used to extract the interesting parts of the result, detect overflow of the MSP (Most Significant Part) and adjust the format of the LSP (e.g. bitreversal for serial I/O, shift for multiprecision). Some logic in the accumulator feedback allows adding/subtracting the contents of the accumulator register. For multi-word precision this contents can be shifted right by 15 bits.

ALU-section:
The ALU-section contains a 16-bit ALU together with a three-port register file with fifteen 16-bit registers. The ALU can perform 31 different operations. In addition to conventional ALU-operations specific operations for multiprecision calculations and dedicated signal processing tasks are available, like rounding and divisionstep.
The register file allows up to three accesses (two read and one write) simultaneously.

Data memories and address calculators:
Two internal RAMs and one ROM are used to store data and coefficients. The capacities of both RAMs (RAM A and RAM B) are 128 words of 16 bit each while the ROM has 512 words. Both databuses have readaccess to all memories, the X-bus is allowed to write to RAM A, the Y-bus to RAM B. A powerful address calculation unit (ACU) for each memory enables fast access in algorithms like filtering or FFT. The instruction set of each ACU is identical, but the address wordlengths are different (9 bit (ROM), 8 bit (RAM B) and 12 bit (RAM A). The basic address operation supports modulo arithmetic for cyclic addressing in a ring:

$$A(k) = B + [\ R(k) + I\ ]_{\mathrm{mod}\ M}$$

with
- A(k) = actual address
- B = base address
- R(k) = relative address
- I = increment (pos. and neg.)
- M = lenght of the ring

The modulo operation is very useful e.g. for FIR-filtering, because any memory shift of the state variables can be avoided.
The lenght M is restricted to a power of 2. Additionaly the actual address can be used in a bitreversed order.

Program control section:
The processor is controlled by a horizontal microcode of 40 bits to provide maximum flexibility and speed. The instruction memory consists of 1024 words of 40 bits. Most of this memory is ROM, but for flexibility 32 RAM words are loadable from external memory (PCB 5010 only).
The program control logic contains a 5-level stack, a condition code multiplexer, a repeat register and an interrupt logic. The program counter (PC) and top-of-stack are available to the X-bus. Thus stack extension by software is possible. The repeat register allows fast repetition of single instructions without further loop control. The condition code multiplexer provides a lot of conditions for

branches, calls, and returns.
The interrupt logic supports a simple scheme for one maskable interrupt input. Besides that external program control can be accomplished by polling four user input flags and/or by setting the four output page bits. Finally it should be mentioned that the program control unit can be switched by software between two different modes.
In the 'FAST' mode the DSP performs 8 million instructions per second. However, in this mode one step of the internal pipeline is visible (e.g. in the multiply-add-operation). In the 'NORMAL' mode each instruction cycle takes 250 ns, but the pipeline is invisible.

3. DSP configurations

The interfaces of the PCB5010 offer a very high degree of flexibility to implement efficiently stand alone or host controlled single- or multi-DSP systems with or without external data memory.
The 16-bit address port for the external world is devided into 4 page bits and 12 bits produced by the ACU A. Thus external memory can be accessed with the same mechanism and the same speed as the internal storage (RAM A).
Three examples of system configurations are presented below.

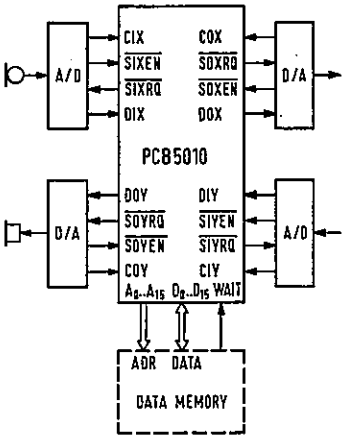


Fig. 2: Configuration for duplex processing

The PCB5010 has two serial input and two serial output ports which can be used autonomously. Fig. 2 shows how to connect two ADCs and two DACs to implement duplex signal processing as required e.g. in a handsfree telephone or in a modem. If the internal data RAM is too small external memory can be connected to the parallel port using the internal address unit A (s. Fig. 1).
If the external device is too slow the DSP can be halted via the "WAIT" input.

In some applications the computational power of a single DSP is not sufficient. In general the algorithms can be devided into subtasks which are carried out in a pipelined fashion by several DSP's as shown in fig. 3.
In this configuration the advantage of independent serial ports becomes evident.

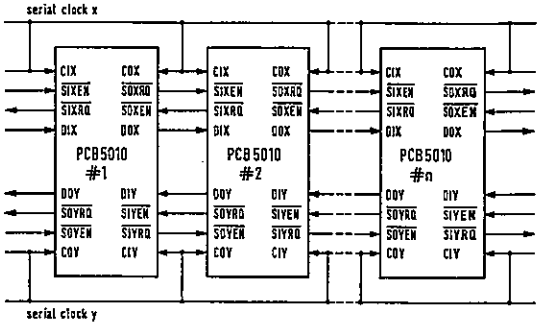


Fig. 3: Serial Multi-DSP configuration

A permanent running common serial shift clock is used for each transfer direction. However, the actual transfer between two processors is controlled automatically by a powerful handshaking mechanism (SIXEN=Serial Input X ENable, SIXRQ=Serial Input X ReQuest,....) without extra software.
The transfer is initiated by writing or reading the internal transfer registers of the serial ports (s.Fig.1). Thus the transfer can be achieved in a synchronous or an asynchronous mode, sample by sample or by blocktransfer. No additional hardware is required.

If in any of these DSPs the internal data storage is too small external storage can be connected additionally to the parallel ports.

An example of a host controlled configuration is illustrated in fig.4. The host and the DSP communicate with each other via a global memory section. Arbitration of the global memory is achieved by using the WAIT input of the DSP.
More sophisticated multi-DSP configurations can be implemented with e.g. several global memory sections and e.g. I/O ports in the external address space.

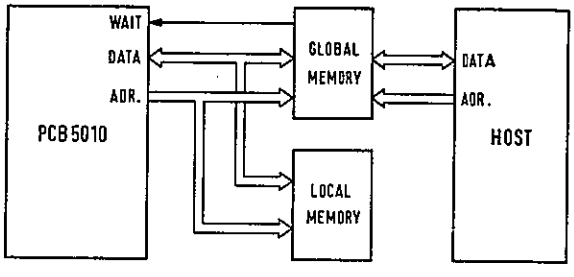


Fig. 4: DSP under host control

4. Applications

Some computational benchmarks of basic algorithms are given in table 1.

Algorithm	Execution time (µs)
2nd-order IIR filter section. . (kernel only)	0.625
8th-order IIR filter. . . . . (cascade of 2nd order sections, including I/O)	3.125
64-tap FIR filter, incl. I/O. .	9.250
16-point complex FFT. . . . . (kernel only, straight line)	28.375
16-point complex FFT. . . . . (kernel only, looped program)	76.625
128-point complex FFT *).	1100
256-point complex FFT *).	2300

Table 1: Benchmarks /1/
*) looped program including windowing, I/O, bitreversal

The PCB5010 is appropriate for complex real time processing of speech and audio signals, as shown by few examples below.
However, it can be used as well for other application areas like biomedical signal processing, control systems, robotics etc.

Wideband Speech Coding at 64 kbit/s:
In the future digital telephone network (ISDN) a transparent digital 64 kbit/s channel is available to the user. Thus for speech transmission more sophisticated coding techniques than the conventional 64-kbit/s-PCM can be used. A new 64 kbit/s 'wideband coding scheme' for speech signals having a bandwidth of 7 kHz will be standardized soon /2/. The DSP operations are: decomposition of the speech signal into a highpass and a lowpass subband using QMF-filters, ADPCM coding of the two subband signals with adaptive prediction and adaptive quantization (6 bit in the lowpass band, 2 bit in the highpass band). The complete codec can be implemented with a single PCB 5010.

Medium Bit Rate Speech Coding:
One candidate scheme for the digital mobile telephony is the 16 kbit/s Regular-Pulse Excitation Linear-Prediction Codec (RPE-LPC, /e.g. 3/). The basic functions are LPC-analysis, interpolation of log. area ratios, inverse filtering (order p=12), FIR filtering (p=10) of the residual, adaptive decimation, and APCM coding. A duplex codec requires a single DSP PCB5010 and some additional external data memory.
An alternative scheme is a subband codec with polyphase-network filter banks (PPN-FB) /3/. A duplex codec with 8 subbands can be implemented with a single PCB5010 without external memory. For the analysis-synthesis PPN-FB about 50% of the data RAM and roughly 10..15 % of the available computation time. are used. Thus, rather complex quantizers and bit allocation procedures can be realized.

Noise Suppression:
Short-time stationary noise in speech can be reduced by spectral analysis (PPN-FB), adaptive amplitude control of subband signals, and spectral synthesis (PPN-FB) /e.g.4/. The hardware expenditure for a system with 32 subbands is a single PCB5010 plus external data memory.

Echo Cancellation:
The most common implementation of echo cancellers uses the least mean squares algorithm (LMS) /e.g. 5/. The basic functions per sampling interval are: n-tap FIR filter, LMS-adaption of n coefficients, calculation of error signal and adaptive stepsize (inculding one division). An n=128 tap echo-canceller requires a single PCB5010, using 100% of the internal RAM and 70% of the computational capacity if the input signal is sampled at 8 kHz.

Speech Recognition:
Due to the fact that external memory can be addressed and accessed at the same speed as the internal RAM A, the PCB5010 can be used efficiently for speech recognition. The number of DSPs is depending from the application requirements like response time and size of vocabulary.

Acknowledgements
Part of this work has been supported by the German ministry of research and technology (BMFT).
The authors gratefully acknowledge the contributions of many colleagues especially of Mr. K. Rinner, J. Schmid, W. Wägener, F. Welten and J. Wittek

References:
/1/ F.J. van Wijk, F.P. Welten, et al.: On the IC Architecture and Design of a 2µm CMOS 8 MIPS Digital Signal Processor with Parallel Processing Capability: The PCB5010. Proceedings ICASSP-86.
/2/ Draft CCITT Recommendation G72x: Wideband Speech Coding. Nov. 1985
/3/ R. Sluyter, P. Vary: Sprachcodierung f. mobile Telefonsysteme. NTG-Fachtagung Bewegliche Funkdienste, München 25.-27. Nov. 1985, Proceedings pp.172-177
/4/ Vary, P.: Noise Suppression by Spectral Magnitude Estimation - Mechanism and Theoretical Limits. Signal Processing, 1985, pp 387-400
/5/ Widrow, B. et al.: Stationary and Non-stationary Learning Characteristics of the LMS Adaptive Filter. Proc. IEEE, 1976, Vol. 64, No. 8, pp 1151-1162

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DESIGNING A CHIP FOR A SYSTOLIC ARCHITECTURE PERFORMING COORDINATE MAPPINGS*

C. Braccini, A. Maestrini and T. Vernazza

Department of Communication, Computer and System Sciences
University of Genova
Viale Causa, 13 16145–Genova, Italy

A parallel architecture and its basic element are presented, designed for geometrical image transformation in real time. The architecture, suitable for VLSI technology implementation, performs efficient cartesian to polar (or log–polar) and affine transformations. The overall system is seen, from outside, as a smart memory capable of receiving ordered data and of sending them out in a different order. After a description of the basic hardware and software features of the overall system, the internal structure of the elementary cell is illustrated.

1. INTRODUCTION

Coordinate transformation represents one of the basic operations in image processing and is needed in a variety of applications, both for image analysis and synthesis: besides the interactive adjustment of display parameters, it is worth mentioning the compensation of geometrical distortions, the registration of multiple images of the same scene (acquired through different sensors or at different time instants), the construction of "computational spaces" suitable for certain classes of processing. Among these latter, the log–polar plane is considered in particular in this paper, beeing it the space where scale and rotation changes in the original x–y domain become simple shifts and where, therefore, scale and rotation invariant representations and recognitions are easily performed [1,2]. Moreover, shift–variant form–invariant processing, of interest in artificial vision, can be implemented in this plane by means of classical shift–invariant filters [3].

Concerning the other applications previously mentioned, the mapping we consider here is the linear (or affine) transformation, that includes simple translation, rotation, scaling (zooming or reduction) and general axonometric projections (where parallel straight lines map into parallel straight lines), and that is often used as a basis for approximating nonlinear mappings.

From the computational viewpoint, coordinate mapping is an expensive task not only for the amount of data to be processed (all the image pixels), but also because the need of addressing each pixel separately makes it impossible to exploit the ordered structure (i.e. raster scan) generally used to represent images, thus preventing the use of array processors.

In the past, we have considered algorithms for the efficient implementation of affine transformations [4]. More recently, we have studied a multiprocessor architecture, suitable for VLSI implementation, that performs cartesian to polar (and to log–polar) coordinate mapping over raster–scan images at TV rates [5].

* Work supported by ESPRIT and by the MPI of Italy

The coordinate transformer we are implementing is meant to be integrated within a specific image processing system (VDS 7001 EIDOBRAIN), whose characteristics of modularity and reconfigurability are particularly suited to integration with special–purpose processors, that directly fit into the basic structure and increase its performances [6]. The architecture of the system is based on a four buses data transfer system capable of an overall transfer rate of 128 Mbyte/sec.: one master bus is used to transfer non–structured data (representing, in our application, parameters, commands and the addressing algorithm specifications) among the subsystems of the processor, and three slave buses carry structured data (like rows of a raster–scan image).

Shortly, the proposed solution to the above mentioned access problem is a smart memory, handling an addressing algorithm rather than a simple address. The preliminary study reported in [5] has evolved, from the general architecture and an optimized methodology for region allocation among the processors, towards a more detailed analysis of the single processor structure and a more general class of coordinate transformations, that include now the affine mapping. In section two the overall structure of the system is described. In the following sections we discuss the architecture and the internal structure of the basic component and its implementation.

2. SYSTEM OVERWIEW

The overall architecture of the proposed system is depicted in figure 1. It is seen from the outside as a memory that can receive ordered data and send them out in a different order operating in real time, i. e. with time delays of no practical importance for the application. From a logical viewpoint, the system can be divided into:

- an interface processor, based on a standard microprocessor, that receives from the host the information needed to choose the optimal task allocation among the elementary cells, and to supply these latter with the addressing algorithm as well as with the suitable

parameters;

- an array of elementary cells, which is seen by the interface processor as a set of peripheral units and by the host as a part of its central memory (where, however, only structured data can be stored). Each elementary cell consists of a standard memory chip and a controller, which is the active element capable of image data processing.

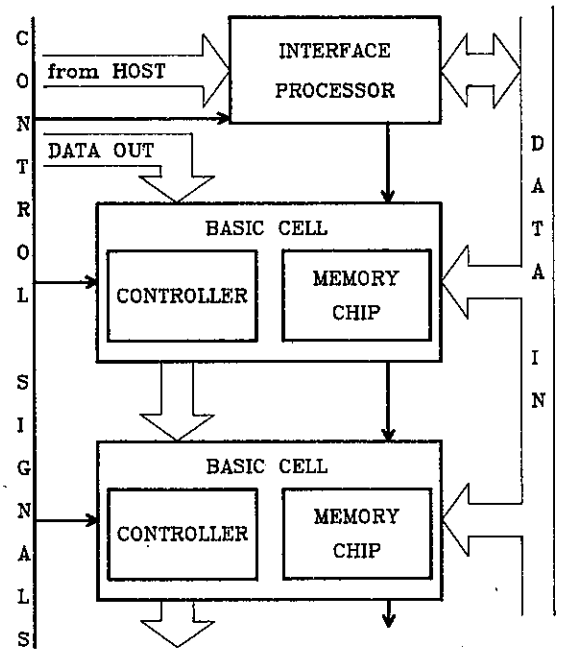


Figure 1

The sequence of the basic operations of the system is the following:

- the host supplies all the above mentioned information;
- depending on the chosen task allocation, the elementary cells are supplied with the addressing algorithm, as well as with all the necessary parameters. These two steps are performed off−line, i.e. before starting the processing step;
- input data are broadcast to all the elementary cells at the same time, as rows of a raster−scan image;
- each elementary cell stores its data in the local working memory and then processes them according to the algorithm provided by the host through the interface processor;
- at the end of the processing step, all the output data are redirected to the host computer in a structured order (to obtain a fast transfer) that can be different from the one used to send them to the system.

As figure 1 shows, all the elementary cells are connected in a sequential fashion, this mode being specially suited for a low and fixed *fan−out*. It is worth noticing that no alignment network is needed, due to the absence of communications among the elementary cells and of a common memory.

This special architecture is directly derived from the basic structure of the algorithms which are to be executed by the system. More precisely:

- the source image is partitioned among the elementary cells; e.g., figure 2 depicts the case where each cell processes an image region whose boundaries are defined by two arcs of radii r_1 and r_2, and two segments at angles θ_1 and θ_2. This partition is best suited to perform the cartesian to polar transformation and to output the processed image by rows (constant radius) or by columns (constant angle), allowing a fast transfer of the data.
- each elementary cell computes the value of an output pixel by interpolating the closest pixels of the input data that it has stored dinamically while the source image has been scanned.

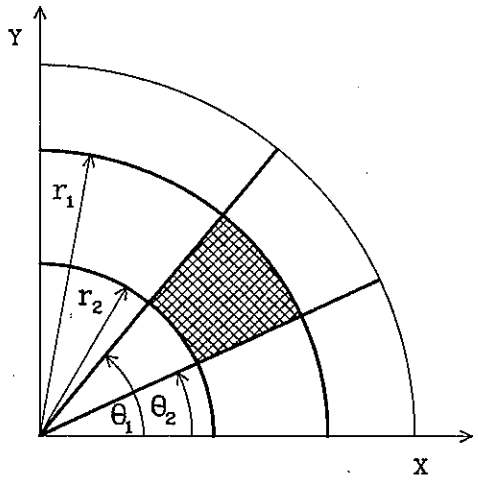


Figure 2

The coordinate mapping algorithms considered here, basically require computing the grey level of the pixels on a straight line of arbitrary orientation with respect to the original axes. As shown in [4], this allows implementing the affine (or linear) transformation, of the form

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ b_1 & b_2 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \tag{1}$$

as well as the cartesian to polar mapping, described by the equations

$$\begin{cases} x = r\cos\theta \\ y = r\sin\theta \end{cases} \tag{2}$$

$$P_{out}(r,\theta) = P_{in}(x,y) \tag{3}$$

where P represents the grey level. In this latter case, a set of lines is defined, intersecting in a common point that becomes the origin of the polar plane.

Instead of the nearest neighbour procedure expressed by Eq. (3), more sophisticated interpolation tecniques can be used, requiring the computation of weighted sums over pixels surrounding (x,y). If the log−polar mapping is performed to implement scale−invariant processing by means of shift−invariant filters, this again requires the computation of weighted sums (convolutions). In conclusion, since the algorithms implementing the above operations only require multiplications and additions, a greatly simplified ALU of the controller is needed.

Thus, it has been shown that each elementary cell acts as a smart memory: in fact it stores in the output plane the pixels (or a linear combination of them) which have been read in the input plane. The great advantage of this system lies in the programmable processing capability of each cell, i.e. the input image can be partitioned among the cells in such a way to optimize the architecture with respect to the task to be performed.

3. BASIC CELL ARCHITECTURE

The overall structure of the basic cell is synthetically described in figure 1 where the flow of data and control signals is also sketched.

The elementary cell of the smart memory consists of two chips: a standard static memory chip of 64 Kbit capacity, and a controller managing the non standard addressing algorithm.

The input data flow can be depicted as a continuous stream of pixels, i.e. it has a pixel serial, grey level bit parallel format, as most of the real time video interface can provide. In other words, the source image is scanned, usually by rows, by a camera and an A/D converter or from an image memory, and an n−bit grey level serial output is provided, where 2^n are the available grey levels.

The control signals are basically intended for I/O synchronization and monitoring during the normal operating mode. Also, in the programming stage, when algorithms and parameters are sent to the elementary cells, they allow non−pictorial data to be transferred from the host computer to each processing element over the data bus.

The output data flow has the same format of the input flow and is therefore readily available to be directed to an image memory for storing, displaying or further processing purposes.

4. STRUCTURE OF THE CONTROLLER

A simplified block scheme of the controller is found in figure 3. It can be seen that its basic elements are: an input unit (including a working memory) that manages the data coming from the outside; a special purpose arithmetic unit, optimized for the most common operations needed in image processing, i.e. multiplications and additions; a microprogrammed sequencer with its microcode memory; an interface towards the memory chip and an output unit that operates concurrently with the microsequencer, in order to overlap the processing and outputting steps.

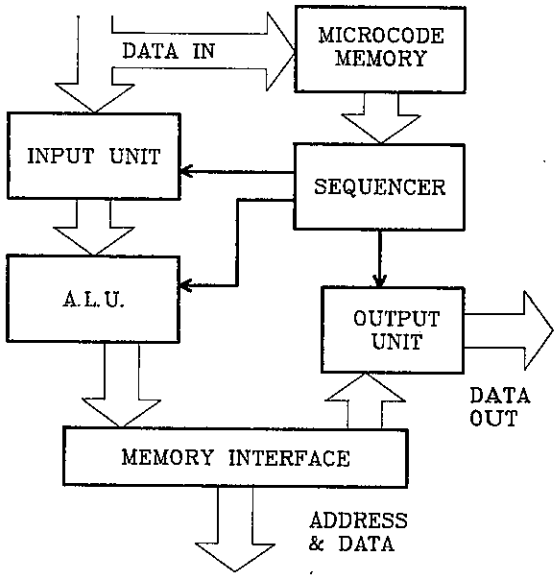


Figure 3

4.1. The input unit

The input unit consists of a double management memory, which can be read with random access, as a RAM, and written in a list (FIFO) fashion. The memory can be viewed as a bank of shift registers, which are serially loaded from the external world and randomly accessed from the processing element.

4.2. The processing unit

The processing unit is a finite−state machine, consisting of a sequencer and a microcode memory. It generates all the control signals needed to fully operate the processing element. A read/write implementation is chosen for the microcode memory, so that the microinstructions can be optimized to the specific task. Obviously an initialization step is needed in the system operations, that is an off−line loading of the microcode memory.

The machine instruction set can be loaded using the data bus and special control signals that enable external data to be written into the microcode memory. A shift register is needed to convert n−bit formatted data coming from the data bus into q−bit formatted data, where n is the input image resolution and q is the size of the microcode word.

4.3. The output unit

The tasks of the output unit are:

- to give fast access to the contents of the output memory
- to multiplex the data coming from the previous processing elements with those coming from the output memory, inserting the latter data in the appropriate time slots of the output sequence.

The first task is easy because the output memory is organized according to the order chosen for the output data.

The second task results from the fact that the output data must be organized as a continuous stream of pixel blocks, characterized by the constancy of one coordinate, in order to obtain optimally formatted data for fast access to the output image memory. Therefore the outputs of the basic cells are synchronized according to the image area which they process. Each cell controls a fixed number of time slots in the output sequence, which are to be filled with its data.

4.4. The arithmetic logic unit

As it has been concluded in section 2, the ALU of the controller must perform only additions and multiplications, provided, of course, that tables containing the values of the trigonometric functions and the radius on a log scale are available.

The ALU has to deal with integer and fractional numbers, but this can be done reasonably easily, since:

- data and addresses are integers (always positive and therefore virtually unsigned)
- the results of equations (1), (2) and (3) must be rounded to the closest integer.

Hence, to multiply an m−bit integer number and an n−bit fractional number (as in (2)), it suffices to multiply them as integer numbers and to round the (m+n)−bit result to the m most significant bits. To add numbers having a fractional and an integer part (as in convolution), it is possible to add the fractional parts separately and round the result to the closest integer. Then, this partial result is added to the integer parts to obtain the final result. It follows that a very fast integer multiplier and adder can be built, with some dedicated logic for managing integer and fractional parts, for saving temporary results and for rounding operations.

Finally, a block for the logical operators is needed to perform data comparison and typical boolean operations.

4.5. The memory interface

This is a very critical part of the project because the output memory cannot be integrated on the chip due to its size limitations. Therefore, the memory interface must be carefully designed, to overcome the bottleneck represented by the fact that data are transferred using standard communication signals and taking also into account that the output memory must be capable of being time shared between two functions: (1)writing the new data coming out from the processing unit, (2)reading the old data to be sent to the output. Since, however, all the access procedures require to output a column or a row of the memory as a continuous sequence, it is possible to store *potential* data coming out from the processing unit in a temporary buffer. Therefore, an addressing procedure is needed, which can read the output memory and send its contents to the output unit with the appropriate throughput. Assuming that the output data rate coincides with the input rate, it is possible to compute the maximum allowed access time, taking into account also the necessary latching of data in the output unit. Commercially available RAM with 30−60 ns access time are fast enough for typical applications (e.g. 1 pixel/125ns), provided that the output unit has a dedicated address generator.

5. VLSI IMPLEMENTATION

The controller will be implemented using CMOS technology; this choice is derived from the clear general advantages of this technology over the others when clock rates up to 10 or 20 MHz are used. These advantages are very well described in [7]. However, the following two considerations are worth mentioning:

- First, a basic requirement of our solution is a large number of active elements, and the CMOS technology provides the highest devices density.
- Second, the operating frequency of our system is well within the frequency range where CMOS components operate without criticity. Moreover, it allows us to obtain a very low−power chip.

References

[1]P.S. Schenker, K.M.Wong, E.G.Cande, *Fast Adaptive Algorithms for Low−level Scene Analysis: Application of Polar−exponential Grid Representation to High Speed, Scale and Rotation Invariant Target Segmentation*, in: SPIE vol. 28, Tecniques and Applications of Image Understanding, 1981, pp.47−57

[2]C.Braccini, G.Gambardella, A.Grattarola, *The Use of Computational Spaces for 3D Object Recognition*, in: "Digital Signal Processing − 84", Proc. of the Florence Int. Conf., North−Holland, Amsterdam, 1984, pp.759−763

[3]C.Braccini, G.Gambardella, A.Grattarola, *The Form Invariant 2D Filtering and its Applications to Pattern Recognition*, in: "Signal Processing II: Theories and Applications" Proc. of Eusipco 1983, North−Holland, Amsterdam, 1983, pp.207−210

[4]C.Braccini G.Marino, Comp. Graphics and Image Processing, vol.13, no.2, June 1980, pp.127−141

[5]C.Braccini, A.Grattarola, A.Maestrini, T.Vernazza, *A Systolic Architecture for Cartesian−to−polar Coordinate Mapping*, in: 3rd Int. Conf. on Image Analysis and Processing, Rapallo, Sept.30−Oct.2, 1985

[6]VDS 7001 − EIDOBRAIN System Manual, VDS (Video Display Systems), Florence, Jan. 1985

[7]P.M.Solomon, Proc. of the IEEE, vol.70, no.5, May 1982, pp.489−509

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MUCOM - A WORKSTATION FOR THE COMMUNICATION ENGINEER

T. Schaub, J. Adame

Research and Develpoment, Landis & Gyr, 6301 Zug / Switzerland

MUCOM is a tool which helps the communication engineer to design an optimum data transmission system for a "real world" channel. The system enables the user to test the performance of several synchronization and decoding methodes in parallel. It combines the realism of a prototype with the flexibility of a simulation program.

1. Introduction

Determining the optimal communications system for a "real world" channel can be very laborious. In particular, when dealing with non-stationary channels, it is nearly impossible to construct an accurate model which can be used in computer simulations. The only remaining possibility is to measure the performance of a certain system in its "natural environment", meaning that most of the system has to be built in hardware. In the following we will present a communications work station which can be used on "real" channels without losing the flexibility of a simulation program. The workstation is based on two Motorola 68000 microprocessors which work in parallel. All time-critical demodulation operations are performed by specially designed coprocessor, leaving the main processor enough time to update the system's parameters (transmission rate, type of modulation, error correcting code, synchronization method ,etc.), to measure the system's performance and to store and display the results.

2. Features

The heart of the MUCOM (multi-user communication) system is a receiver which is capable to process signals from several transmitters in parallel. The transmitters can use completely different synchronization-, modulation- and error correcting schemes. This concept allows the user to perform two types of tests:

a) System performance test:

where signals from several transmitters are received by several independent receivers. The system performance test should answer questions like:
- How does the system as a whole perform?
- Which modulation- and coding schemes are robust against collisions?

b) Method performance test:

where one transmitted signal is received and processed by several different methods. It allows the designer to compare different synchronization methods or decoding algorithms on a time varying channel. The parallel structure makes it possible to compare the different methods always under the same channel conditions.

The following features are implemented in the system:

System Parameters:
Up to 4 independent receivers can be run in parallel. They can use their individual transmission speed (< 100 baud). The carrier frequencies must not exeed 25 kHz.

Modulation:
Besides conventional techniques like Binary Frequency Shift Keying, also Frequency Hopping is implemented. The hopping patterns can be chosen for each receiver individually; allowing the user to optimize the patterns for different point to point connections and therefore to keep the number of collisions at a minimum.

Synchronization:
The system is designed for the case where all transmitters send their messages in a completely unsynchronized manner. (Only symbol synchronization is assumed, which is achieved by means of the zero crossings of the mains). In order to detect the beginning of a mes-

sage, the telegrams are preceeded by a synchronization sequence.
This sequence is divided into two subsequences. The first of which consists of a periodic binary sequence with a short period (e.g. 1,0,1,0,...). It is used to detect the presence or absence of a transmisssion and to obtain a presynchronization.
The presynchronization sequence is followed by a synchronization word, indicating the beginning of the telegram. (In cases where a convolutional code is used the synchronization word starts the Viterbi decoder.)
The synchronization word has to be chosen such that it will not erroneousely be detected during the time the presynchronisation sequence is sent. This condition can be best met in using a long word which has a low correlation with all possible distinct shifts of the periodic presynchronization sequence. On the other hand the synchronization word should be detectable even if it has some errors in it. Otherwise the missynchronization rate would become higher than the telegram error rate. It turns out that the two conditions are very hard to be satisfied simultaniousely. Particularely when a error correcting code is used, it becomes necessary to use the error correcting capabilities of the code also to detect the beginning of a telegram.
A new synchronization method was developped which is designed to be used in conjunction with rate 1/2 convolutional codes. The key point of that method is the fact that for every optimum rate 1/2 convolutional code, the ...1,0,1,0,1,... sequence is a valid code sequence, during which the encoder stays in the same internal state [11...1]. This means that when the encoder's shift register is loaded with "ones" and only "ones" are put in, the output sequence will be ...1,0,1,0,... . The decoder which receives a ...1,0,1,0,... sequence will constantly decode a "one" and stay in its internal state [11...1]. Applied to our situation this means that during the presynchronisation sequence(...1,0,1,0...), which is also a valid code sequence, the receiver decides if a transmission is taking place and if so, the Viterbi decoder is started in the [11...1] state. As long as the decoder output stays "one", we are still receiving the presynchronization sequence. The first "zero" bit at the output indicates that the next decoded bit will be the first bit of the telegram. Figure 1 shows the blockdiagram of the described algorithm. The upper comparator checks if the presynchronization sequence is

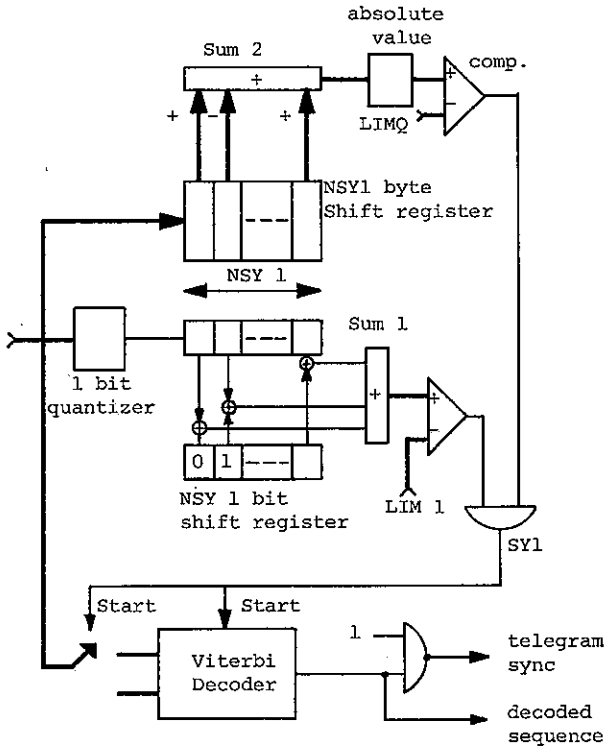


Fig. 1 Telegram synchronization

received such that the average quality of the last NSY1 received symbols exeeds the limit LIMQ. The lower comparator tests if the received binary sequence is in phase with the code sequence. When both thresholds are exeeded the demultiplexer and the decoder are started. After a certain delay (given by the surviving path length) the decoder will put out "ones" during the rest of the training sequence. As soon as a "zero" leaves the decoder, the "telegram sync" output becomes "one", announcing that the next decoded bit will be the first bit of the telegram.
Both synchronization methods are implemented. For the cases where no rate 1/2 convolutional code is applied, the first method must be used. If a rate 1/2 convolutional code is used, the designer has the choice between the two methods.

Coding:
Each of the receivers may use its own error correcting code. Block and Convolutional codes are possible. All optimum convolutional codes of rate 1/2 to rate 1/4 and of constraint length 3

to 9 are stored in a data bank and can be selected by the user according to their free distances.
As mentioned above, the system provides each receiver with its individual Viterbi decoder. The surviving path-lengths are chosen according to the codes' constraint lengths. Decoding is done with soft decisions.

Man - Machine Interface:
The user has always the possibility to force the system into one of the following states:
- Stop all receiving processes. From here on the parameters can be updated or the actual states of the receiving processes can be analysed in detail. It is also possible to configurate a whole new system. The receiving frequencies, the number of activated receivers, etc. can be redefined. The operator is guided through all the updating processes by menue techniques.
- Display the receiver's data. Without interrupting the receiving processes statistics on the performance of the different receivers are displayed. Tables of error rates, synchronization error rates and collision rates are shown on the terminal together with the time when they were measured. During the display process the tables are updated with new data.

3. System Architecture

The system is divided into two subsystems each of which uses a Motorola 68000 processor. One of the processors acts as a coprocessor. It is surounded by dedicated hardware to cope with the "time critical" parts of the signal processing.

3.1 The Coprocessor

The coprocessor acts as a filterbank. The quadrature demodulator (QD), like the one shown in figure 2 represents the basic structure of the filter. If the system uses more than one frequency (as in FSK, frequency hopping, etc.) several parallel QD modules are needed, one for each frequency. The hardware implementation of the QD module requires four multipliers and two integrators. Analogue multipliers and integrators suffer from offsets and drifts. Their digital counterparts do not share these problems and recently have overcome their own former disadvantages: limited speed and high price. In order to increase the processing speed, parallel processing becomes necessary. The QD structure was therefore divided into two sections. The first

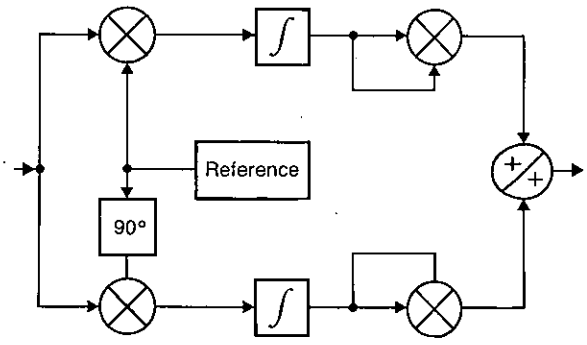


Fig. 2 Quadrature demodulator (QD)

one contains dedicated hardware which performs all time critical operations like AD conversion, storage, multiplication and integration. The second section, which takes care of the rest of the calculations, was built around a Peripheral Interface Processor (PIP) [1]. The latter provides an efficient way for preprocessing data which has to be transfered to a host computer for further calculations. A block schematic of the implemented QD structure is shown in figure 3. The received signal is sampled, converted to digital, and stored. At the same time (in parallel), the last fully received symbol is processed. Multiplication and integration for each branch are performed by a MAC(multiplier/accumulator). The reference signals are stored in EPROM and include those frequencies used by the system and their ninty degrees phase shifted versions. If some apriori information about the channel's characteristic is known, the references may

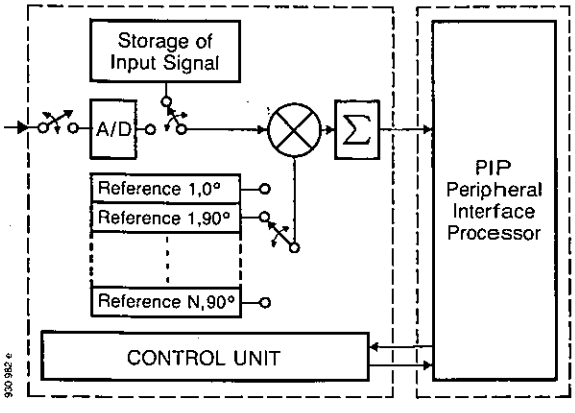


Fig. 3 Implementation of the QD

be matched to compensate the distortions. Operations which are shown in parallel in figure 2 are in fact performed sequentially in this arrangement. The filter outputs are stored in a cicular buffer which is updated each 10 miliseconds. The main processor reads its input data from that buffer. Time delays of the symbol synchronization between transmitter and receiver may be individually compensated due to the correlator structure of the receiver.

3.2 The Main Processor

The main processor uses the output of the filterbank (coprocessor) as input for all its activated receivers. It performs the synchronization algorithm, despreads the the received signal (for spread spectrum modulation), detects and corrects the errors and computes the performace statistics for each of the receivers. In addition it provides the interface to the operator. These tasks are performed by several parallel processes which interact with each other. The entire software is written in PORTAL [2], a high level realtime programming language which supports parallel processing.
The structure of the main processor's program is shown in figure 4. It consists of 6 main processes:

Sync-and-Fill collects the received and demodulated raw data from the coprocessor. It performs for each of the activated receivers its individual synchronization algorithm and - if synchronisation is achieved - the raw data is stored in the corresponding receiver module.
This process is the most time critical one and it limits the number of parallel receivers that can be implemented. The Sync-and-Fill process has to perform the synchronization algorithm for all receivers, before the coprocessor provides the new filter outputs (10ms).

Receiver 1 to 4 assemble the raw data to code symbols according to the used hopping patterns. They perform the decoding algorithm (Viterbi decoder for convolutional codes) and compare the decoded messages with their stored references. The resulting error rates, synchronization error rates and collision rates are written into a common results table.

Service handles all interaction with the operator. It allows to change the system parameters and to display error rates from the results table. All interaction through the service process is done without interrupting the receivers.

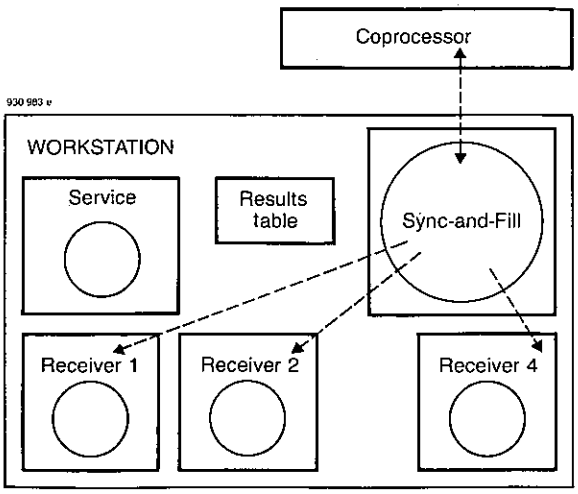


Fig. 4 Structure of the workstation

4. Conclusions

The MUCOM system provides the communication system designer with a flexible tool to determine the optimum data transmission system for a "real-world" channel. It allows the comparison of different transmission schemes in parallel. The signalling rates are limited to 100 bauds.
The parallel processing capability is extremly useful when dealing with time variant channels. The MUCOM system has been successfully used designing communication systems for data transmission over the power mains.

References:

[1] Lingg, H., Peripheral Interface Processor, Proceedings of Journees d'Electronique, Lausanne 1984.

[2] Lienhard, H., Schild R., PORTAL produces reliable software for realtime systems - and fast, Electronic Design, January 4,1980.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN EFFICIENT AND SYSTEMATIC TECHNIQUE FOR THE PARALLEL IMPLEMENTATION OF DFT ALGORITHMS

I. Pitas, M.G. Strintzis
University of Thessaloniki
Department of Electrical Engineering
Thessaloniki 54006, GREECE

Abstract

The discrete Fourier Transform (DFT) algorithms (Winograd Fourier Transform Algorithm, Prime Factor Algorithm, radix-2, radix-4 Fast Fourier Transform) have very canonical, modularized, parallel structures. Therefore they are suitable for parallel implementation. The present work proposes a systematic and flexible technique for the parallel implementation of these algorithms with or without hardware constraints. The technique is based on the "Critical Path Method" and the "Project Planning Under Limited Resources" Operations Research methods.

I. INTRODUCTION

The Discrete Fourier Transform of a sequence x(n) is defined as:

$$X(k)=\sum_{n=0}^{N-1} x(n)W_N^{nk} \qquad (1)$$

$$W_N=\exp(-i\,\frac{2\pi}{N})$$

The DFT can be efficiently computed using various fast methods such as the FFT developed by Cooley and Tukey [1], as well as more recent methods, including in particular, the Winograd Transform and the Prime Factor Transform Algorithm. We note that, generally speaking the last method is the fastest of the three [8]. These algorithms are not described here, because they are very well known [1,2].

The computation of the DFT means of the above-mentioned fast algorithms is very important for the fast calculation of one dimensional and multidimensional convolutions [10]. The speed obtained by the serial computation of the algorithms on special-purpose signal processors is not sufficient for certain applications, e.g. radar signal processing. Thus, these algorithms have to be implemented in a parallel way. A parallel or pipelined implementation of the radix-r FFTs has already been performed [10]. Also a parallel implementation of WFTA has been proposed in [6]. However the parallel implementation of these algorithm has not been performed in a systematic way. The aim of this paper is to provide a systematic method for the parallel implementation of the fast Fourier transform algorithms. The method proposed takes into account various technical or economical constraints and gives the optimal implementation under these constraints. The proposed method is besed on the scheduling techniques used in the activity networks of Operations Research [9]. A similar method for the parallel realization of digital filter structures has already been proposed by Zeman and Moschytz [5]. However [5] is mainly limited to the parallel imprementation of filters without transform techniques. Our work is concentrated on the parallel implementation of transforms.

Section II describes a structure suitable for the parallel signal processor proposed. Sections III,IV,V,VI, describe the application of the Operations Research techniques on the parallel transform implementation. Section VII gives examples of the parallel implementation of the PFA and the WFTA algorithms.

II. PARALLEL COMPUTER STRUCTURE

The structure of the computer proposed for the parallel implementation of the fast Fourier algorithms is illustrated in Figure 1. It consists of a parallel processing unit and a large Random Access Memory (RAM) for bulk data storage. The parallel processing unit consists of several multipliers and adders operating in parallel. Their number, which is of course limited by the technology used, depends on the specific fast algorithm to be implemented. In the next section we shall discuss how this number is chosen in a systematic way. The parallel processing unit also contains several fast registers which store intermediate results. The number of the registers depends also on the specific algorithm to be implemented and especially on the structure of its modules. Finally several buses and bus switches connect the registers to the RAM, to the multipliers and to the adders. The buses shown in Figure 1 can be substituted by a single time-multiplexed bus. This is a slow but economic and technically simple solution. Each multiplier of the parallel processing unit is connected to a coefficient memory. The characteristics of this parallel processing unit are the following:

a) The addition time T_a (in microcycles) and the multiplication time T_m (in microcycles) include the time to load and store the results from and to the registers.

b) The multiplication time is double than the addition time.

c) The time T_t required for the memory transfers from the RAM to the registers and vice versa is half than the addition time.

The parallel processing unit, as it is described here, is a basic structure for the parallel computation of any type. It has also been used in [5] for the parallel implementation of digital filters. It will be used here because it is very well suited to the modular structure of the fast Fourier transform algorithms.

III. TECHNIQUES FOR THE SYSTEMATIC PARALLEL IMPLEMENTATION OF DFT ALGORITHMS

The flow diagrams of the various DFT algorithms are essentially modular directed graphs [9]. Thus they have many similarities to the activity networks [9] used in the Operations Research. The activities encountered in the graghs of the DFT algorithms are the following:
a) Real additions
b) Real multiplications
c) Data transfers from the parallel processing unit to the RAM and vice versa.
Certain resources (adders, multipliers, buses) are needed for the accomplishment of these activities. The number of these resources is limited due to technical and economic reasons, as it has already been analyzed in the previous sestion. Thus the problem is the following:"How can the various activities be scheduled, so that the algorithm is computed in the fastest parallel way, with the limited resources available?". The solution to the problem can be found by two methods:
a) The Critical Path Method (CPM) which implements the algorithm in the fasted possible way, without taking into account resource constraints.

b) The Project Planning Under Limited Resources (PPULR), when resource constraints have to be taken into account.

IV. CRITICAL PATH METHOD

Each activity network has various parallel and serial works. In its directed graph there always exist a path called the "critical path". The works along this path must be done always serially and they cannot be parallelized. Associated with the "critical path" is the "critical time" T_c, i.e. the minimum time for the project execution. All the works can be organized in a parallel manner within the critical time, assuming that there exist no constraints for the resources used for the project execution. Thus, the critical time is the measure of the possibility of parallelization of the directed graph, since it depends only on its inherent parallelism.

The critical path method finds the critical path of the activity networks and tries to organize all its works around it. This can be done in four successive steps:
a) Computation of the earliest execution time $T_E(i)$ of each activity i, $1 \leq i \leq M$ (M is the total number of the activities).

b) Computation of the latest execution time $T_L(i)$ of each activity i, $1 \leq i \leq M$.

c) Computation of the time slack $T_s(i)$ of each activity i, $1 \leq i \leq M$.

d) Computation of the maximum number of the resources required.

We do not give futher details on the algorithms used for the steps (a)-(b) because they can be found in the operations research literature. The time slack of each activity is the measure of the freedom to move its execution forwards or backwards. The activities on the critical path have always zero time slack. This we have no freedom in their execution. The critical time T_c of the algorithm can be found in step (a) and it is given by the following formula:

$$T_c = \max_{1 \leq i \leq M} \{ T_E(i) \} \qquad (2)$$

The algorithm cannot be performed in time less than T_c, no matter how many resources are available. The algorithm can be computed in T_c only if the resources available are equal to that given by step (d). If the resources calculated in step (d) are above the economical and technical constraints, the PPULR method must by used.

We shall give, as an example, the application of the CPM to the parallel implementation of the preweave section of the DFT module N=7 shown in Figure 2. The critical path is denoted by a double line. The critical time is $T_c = 4T_a$. A time chart for the execution of the various activities is shown in Figure 3, requiring six adders. This is of cource an extraordinary amount of hardware. It will be drastically reduced by the PPULR method.

V. PPULR METHOD

The project planning under limited resources is an NP-complete problem [11]. However very good solutions can be found by some heuristic algorithms [9]. We have developed the following three figures of merit to measure the efficiency of a solution given by the heuristic algorithms:
a) The total hardware cost.

b) The deviation from the critical time T_c.

c) The efficiency of the use of the hardware resources.

The first figure of merit is given by the following formula:

$$c = N_a c_a + N_m c_m \qquad (3)$$

where N_a, N_m are the number of the adders and the multipliers used in the solution and c_a, c_m is their cost per unit.

The second figure of merit is expressed by:

$$a=\frac{T_{at}}{T_c} \tag{4}$$

where T_{at} is the time required for the algorithm computation in the solution found by the heuristic algorithm. Number a is always equal or greater than 1. It is equal to 1 only for implementations in the critical time.

The third figure of merit is expressed by the ratio:

$$b=\frac{T_u-T_g}{T_u} \tag{5}$$

T_u is the time which corresponds to the optimal use of a certain resource. T_g is the overall time in which a certain resource is inactive during the algorithm computation. The ratio b must be evaluated for each resource separately. Thus the ratio b for the adders and the multipliers is given by:

$$b_a=\frac{N_a T_{at}-T_{ga}}{N_a T_{at}} \tag{6}$$

$$b_m=\frac{N_m T_{at}-T_{gm}}{N_m T_{at}} \tag{7}$$

We have used a heuristic algorithm to solve the PPULR problem. It performs the CPM, which calculates the critical time T_c and the maximum number N_{amax} and N_{mmax} of adders and multipliers required. Then the algorithm starts from a fully parallel computation which requires N_{amax} adders and N_{mmax} multipliers. It performs a scheduling of the activities and it evaluates the figures of merit a,b,c. If they are within the acceptable levels, the solution has been found. If the cost c is within the acceptable level, but the speed ratio a is unacceptable, there is no feasible solution. If the cost c is very high the numbers N_a,N_m of the adders and multipliers are reduced until the cost and the speed figures of merit are within the acceptable levels. The criterion b simply evaluates the efficiency of the solution found. The key routine in the algorithm is the scheduling routine which is a heuristic one [5]. Therefore, the solutions found by the algorithm are neither unique, nor optimal. Thus if no feasible solution is found, or if the efficiency b of the solution is low, the algorithm must be repeated again several times, until an acceptable solution is found.

In the following, we give an examble of the application of this method for the pre-weave module N=7, which is illustrated in Figure 2. The schedule of the parallel implementation of this module with 1,2,3,4,5, and 6 ALUS is shown in Figure 3. An efficient implementation is achieved with 4 ALUS. This configuration achieves fast execution of the algorithm (a=1.25), combined with effective use of hardware (b=0.24) and a reasonable number of ALUS.

An important factor that has not been taken into consideration thus far, is the burden of memory transfer. This problem is addressed in the following section.

VI. DATA FLOW ANALYSIS

The data flow analysis is very important for the fast parallel implementation of an algorithm. It determines the following features of the implementation:

a) The number of buses required.
b) The number of registers of the parallel processing unit.
c) The time distribution of the data transfers.

Thus a good data flow analysis reduces the number of buses and bus switches required and the imput-output burden which deccelarates the execution time of the algorithm. The data flow analysis of the fast DFT algorithms is particularly simple because of their modular form. The computation of these algorithms can be done module-by-module. This approach facilitates extremely the control of the parallel processing unit and reduces the I-O burden. If the module i has N_i input data and M_i output data, the I-O required for its computation is:

$$T_i=2(N_i+M_i)T_t \tag{8}$$

for complex DFTs. If the DFT algorithm is an in-place algorithm and has L stages, $(1\leq i\leq L)$, the I-O time its computation is:

$$T_{IO}=2\sum_{i=1}^{L}\frac{N}{N_i}2N_iT_t=4NLT_t \tag{9}$$

The number of buses required in the parallel processing unit is equal to the maximum number of adders and multipliers operating in parallel. Thus 4 buses are required for the implementation of the N=7 preweave DFT module shown in Figure 2. One extra bus is needed for the I-O operations.

The number of the registers required is equal to the maximum number of the intermediate results which must be stored. It depends on the particular implementation of every module. After the PPULR and the I-O operations analysis, a diagramm of the register requirements can be easily constructed. The data flow analysis together with the CPM and PPULR methods are the tools for the systematic analysis of the parallel implementation of the fast Fourier Transform algorithms. We shall use these tools in the parallel implementation of the DFT algorithm of lenght N=120 in the next section.

VII. PARALLEL IMPLEMENTATION OF THE PRIME FACTOR ALGORITHM FOR 120 POINTS

The PFA for a transform length N=120=3·5·8 is a typical example of a transform algorithm of a data sequence of medium length. The modules N=3,N=5,N=8 used in the PFA calculation can be found in [2,8]. It can be easily proven that the critical path of the PFA N=120 is the sum of the critical paths of its modules. Its critical time is:

$$T_c=2T_a+T_m+2T_a+3T_a+T_m+3T_a+2T_a+T_m+2T_a=14T_a+3T_m \tag{10}$$

It is impossible to implement the PFA N=120 in its critical time, because it requires an extraordinary amount of hardware. Thus the three modules N=3,N=5,N=8 are calculated separately. Each module is analyzed by the PPULR method. Sufficiently good results can be obtained by using 2 adders and 2 multipliers. In this case the modules N=3,N=5,N=8 can be calculated in time $10T_a$, $22T_a$,$26T_a$ respectively. The time T_{at} required for this implementation is:

$$T_{at}=5.8\cdot 10T_a+3\cdot 8\cdot 22T_a+3\cdot 5\cdot 26T_a=1318T_a \qquad (11)$$

The speed ratio in this case is a=65.9
The efficiency of the use of the adders and the multipliers is 0.78 and 0.385 respectively.

REFERENCES

[1] E.O.Brigham,"The Fast Fourier Transform" Prentice Hall 1974.

[2] J.H.McClellan,C.M.Rader,"Number Theory in Digital Signal Processing"Prentice Hall,1979.

[3] H.Silverman,"A method for programming the WFTA",IEEE Trans.Acoust.Speech & Signal Processing, vol.ASSP-25,April 1977, pp.152-165.

[4] D.Kobla,T.Parks,"A prime factor FFT Algorithm Using high-speed convolution" IEEE Trans.Acoust. Speech & Signal Processing,vol.ASSP-25,No4, Aug.1977, pp.281-294.

[5] J.Zeman,G.Moschytz,"A systematic approach to the design & speed comparison of signal processor architectures for digital filtering" IEEE Trans.Acoust.Speech & Signal Processing,vol.ASSP-31, pp.1536, Dec.1983.

[6] Y.Wallach,A.Shimor,"Alternating sequential parallel versions of FFT",IEEE Transactions on Acoust.Speech & Signal Processing,vol.ASSP-28, No.2, April 1980.

[7] J.Skyttä,"Comparisons between FFT & WFTA programs",Proceedings of International conference on Digital Signal processing,Florence,Sept.1981.

[8] C.S.Burrus,P.W.Eschenbacher,"An In-place,In-order Prime factor FFT algorithm", IEEE Trans.on Acoustics,Speech & Signal processing,vol.ASSP-29, No.4,August 1981, pp.806-817.

[9] S.E.Elmaghraby,"Activity networks:project planning & control by network models"J.Wiley 1977.

[10] A.V.Oppenheim,"Applications of Digital signal processing" Prentice Hall, 1978.

[11] C.H.Papadimitriou,K.Steiglitz,"Compinatorial optimization" Prentice-Hall 1982.

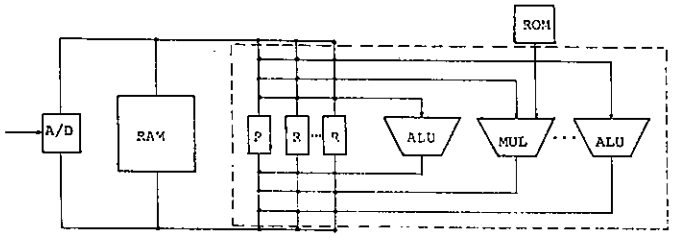


Figure 1:Proposed computer structure for the parallel computation of the DFT algorithms.
RAM:Random Access Memory for signal storage.
ROM:Read Only Memory for coefficient storage.
R:Registers of intermediate results.
ALU:Arithmetic unit
MUL:Hardware multipliers.
A/D:Analog to Digital converter.

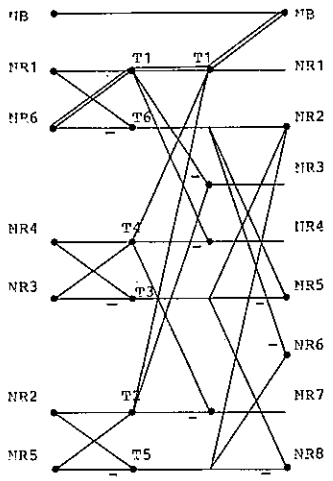


Figure 2: Preweave module of the DFT N=7.

ALU		a3	b3
1	T1 T6 T4 T3 T2 T5 T1 NR3 NR4 T1 NR7 NR2 NB NR2 NR5 NR6 NR8	425	100
2	T1 T4 T2 T1 NR7 NR2 NR2 NR8 T6 T3 T5 NR3 NR4 NB NR5 NR6	225	94
3	T1 T3 T1 T1 NB NR6 T6 T2 NR3 NR7 NR2 NR8 T4 T5 NR4 NR2 NR5	150	94
4	T1 T2 T1 NB NR8 T6 T5 NR3 NR2 T4 T1 NR7 NR5 T3 NR4 NR2 NR6	125	85
5	T1 T5 T1 NB T6 T1 NR2 NR2 T4 NR3 NR5 T3 NR4 NR6 T2 NR7 NR8	100	85
6	T1 T1 T1 NB T6 NR3 NR2 T4 NR4 NR6 T3 NR7 NR8 T2 NR2 T5 NR5	100	71

Figure 3: Order of operations of the DFT module N=7 for implementations using different numbers of ALUs.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A SOFTWARE PACKAGE FOR DESIGN AND ANALYSIS OF DIGITAL FILTERS

C. DARMOUNI, E. LOFFLER, L. DOUSSET

THOMSON INFORMATIQUE SERVICES
104 rue Castagnary 75015 PARIS
Phone : 48.56.31.11

SUMMARY

This paper presents a software package of digital filtering developped by Thomson Informatique Services. These programs solve most of filtering problems including the synthesis, the analysis and the optimization of filters in the aim of their hardware implementation. These programs are included, in the form of modules in a BANK structure. This allows an easier access and usage.

1 - INTRODUCTION

During the last years, the Data-Processing Department of Thomson Group has been commissioned to furnish Computer Aided Design's software packages, specialized in the electronic field, to the laboratories of Group. Its object: furnish to the technicians and engineers some design's modern tools well matched with the actual technical and economical constraints. In contact with the electronics specialists, this department has got an indisputable experience of laboratory and computerised design.

Born of this department, Thomson Informatique Services (Thom'6) continues and extends the "product" policy followed until now by the Thomson Group in relation to software, allowing all its customers access to the most advanced data-processing applications under economical and reliable conditions.

The Software Package presented in this contribution has been developped in this state of mind.

2 - THE BANK LTS

2.1 Presentation

The BANK LTS is a software package of signal processing, related to the analog and digital filtering. The eight programs furnished to the users, solve most of filters'synthesis and analysis problems. Largely tested through the Thomson Group, the BANK LTS finds its place in each department in charge of analog and digital signal processing devices'design.

Further more, the modular structure of the BANK allows an easy development corresponding to each user's specific requirements.

2.2 Structure

The BANK LTS includes two groups of programs :

- programs of analog filters synthesis and analysis
- program of digital filters synthesis and analysis

Each program consists of a module including several commands of :

- help for the use of every program
- synthesis, analysis and implementation of filters.

Only the digital programs of the BANK LTS will be presented in this paper.

3 - DIGITAL PROGRAMS PRESENTATION

The BANK LTS contains five digital programs :

CATI : synthesis of cascade and lattice recursive filters

ESIR : synthesis and ladder recursive filters

SOPI : optimized synthesis of cascade recursive filters

SIFR : synthesis of non recursive filters

ANFI : analysis and simulation of digital filters

3.1 Program CATI

The program CATI performs the synthesis of recursive digital filters according to a magnitude and/or phase frequency model. The program CATI offers, for the filters realization, several efficient structures.

The starting point of the general synthesis consists, either in computing an initial approximation of the desired requirements using classical analog functions (Butterworth, Bessel, Papoulis, Chebychev I, Chebichev II, Cauer) and bilinear or matched Z transformation, or in describing manually a Z transfer function [1] , [2] , [3] .

Then, the user describes the magnitude and/or phase's real requirements by straight segments or oblique segments.

An optimization program using the MIN-MAX algorithm [4] optimizes the initial solution, matching with the already described real requirement.

Thus, for given specifications, the user rapidly obtains several satisfactory solutions. Some analysis possibilities, proposed by the program CATI, allow to choose the best solution.

To realize the filter choosen by the user, the program CATI offers several cascade structures, whose the basic block offers more interesting characteristics than the classical direct structure, which uses a great number of digits. The user can choose between four sample lattice sections which have been integrated in the program, on account of their sensitivity properties. The first three sections have been proposed by H. GRAY and J.D. MARKEL, the fourth by S. K. MITRA [5] , [6] .

3.2 Program ESIR

A class of low-sensitivity digital filters may be realized by direct analogy with LC ladder filters which, in the analog domain, have the same property.

From an analog ladder filter (Butterworth, Papoulis, Chebychev, Cauer or any filter), the program ESIR generates the corresponding digital ladder filter with the help of L.T. BRUTON algorithms [7] , [8] .

To simulate, in the digital domain, the differentiators (s) and the integrators (s^{-1}), which constitute the analog ladder filter, the user may choose between two kinds of transformations :

- LDD (LDI) transformation :
  $s \longrightarrow (z - 1) / T.\ z^{\pm\frac{1}{2}}$
- DDD (DDI) transformation :
  $s \longrightarrow (z - 1) / T$

These transformations are acceptable If the ratio of the sampling frequency on the cutoff frequency is very high. Under such a condition, the filter behavior is remarkable.

3.3 Program SOPI

The program SOPI performs the optimized synthesis of recursive digital filters. The algorithmic part of this program results from DOREDI of IEEE program [9] .

First, the program SOPI performs a classical synthesis of recursive digital filters, using analog functions (Butterworth, Chebychev I, Chebychev II, Cauer) and the bilinear transformation. The structure of the computed filter is a cascade structure realized by a sequence of first and second order blocks.

Then, the program SOPI proposes two kinds of optimizations :

1 - an optimization of the registers' word length containing the coefficients. Indeed, in real practice, the hardware implementation of a filter needs to have its coefficients coded with a finite number of bits.

2 - a minimization of the computed noise by optimization of :

- the pairing of poles and zeros of the transfer function
- the order of the blocks sequence
- the constant gain factor of each block.

3.4 Program SIFR

The program SIFR performs the synthesis of non recursive digital filters with linear phase. The algorithmic part of this program results from PARKS, MAC CLELLAN and RABINER algorithms [10].

A first program calculates the exact and rounded coefficients of non recursive filters. The method used is an iterative method based on the REMEZ exchange algorithm whose solution leads to the desired ideal frequency response, according to the Chebychev approximation.

The program synthesizes the most common ideal filter-types as multi-band, band-pass filters, Hilbet Transform filters and differentiators.

A second program only synthesizes non recursive filters with rectangular magnitude model, but offers the possibility of an optimization of the registers'wordlength. The optimization algorithm includes two iteration loops : the first iteration estimates the filter length and the second estimates the coefficients wordlength. The filter synthesis, by the REMEZ's algorithm, is realized when the statistical estimation of the magnitude error (due to coefficient rounding) is such that the initial tolerances are not "violated".

3.5 Program ANFI

The program ANFI simulates and analyses a digital circuit described element by element (adders, multipliers, delays). The description takes care of characteristics of the digital machine which will effect the filtering operation, for instance : digits number used for the intermediate calculations, arithmetic types (fixed or floating point arithetic, overflows...)[11] ,[12] .

Simulation

A binary register (defined by LSB and MSB) is joined to each node of the filter description. Then the simulation is effected, sequentialy, in an order fixed by the user, as a digital machine including only an arithmetic unit, with the specified precision for each register.

Analysis

The program ANFI makes three kinds of analysis : analysis in the frequency domain, analysis in the time domain and analysis of the noise due to coefficients'quantization.

The program ANFI is compatible with the others synthesis programs of the BANK LTS that have already been described.

4 - DEVELOPMENT OF THE BANK LTS

Several developments are suitable :

- The creation of digital filtering new programs which would combine with the modular structure of the BANK LTS and would complete already-existent programs :
  - . program of digital filters sensitity's calculation to choose the best structure according to the initial requirements
  - . program of non linear phase FIR filters synthesis.
- The creation, following the same structure, of a computer aided training on the digital signal processing, related to the digital filtering.
- The interface of the BANK LTS with ILS (Interactive Laboratory System), that would allow to integrate the BANK LTS synthesized filters in the ILS processing devices. Thus, the user could process signals from ILS by the BANK LTS efficient filters.

5 - CONCLUSION

Therefore, the programs of digital filters synthesis, analysis and optimization have been inserted into a flexible and easily-accessible structure, allowing to get rapidly a satisfactory solution, matching given requirements (frequency requirements, coefficients wordlength, filter structure...).

REFERENCES

[1] M. BELLANGER
Traitement numérique du signal
Théorie et pratique
MASSON

[2] R. RABINER, B. GOLD
Theory and application of digital signal processing
PRENTICE-HALL

[3] R. BOITE et H. LEICH
Les filtres numériques
MASSON

[4] K. MADSEN, HCHJAER - JACOBSEN, O. NIELSEN et THRANE
Efficient Minimax Design of Networks without Using Derivatives (IEEE Microwave Theory and Techn., vol. MTT 23, n° 10, octobre 1975, p. 803-809).

[5] A.H. GRAY, JR Member IEEE and JOHN D. MARKEL, Member IEEE
Digital lattice and ladder filter synthesis. IEEE Transactions on audio and Electro-acoustics, vol. AU 21, n°6 , décembre 1973

[6] S.K. MITRA, P.S. KAMAT, D.C. HUEY
Cascade lattice realization of digital filters.
Circuit theory and applications, vol. 5, 3-11 (1977)

[7] L. T. BRUTON
Low sensitivity digital ladder filters IEEE Transactions on circuits and systems, vol. CAS. 22, p. 168-176, mars 1975

[8] E.S. LIU, L. E. TURNER and L. T. BRUTON
Exact synthesis of LDI and LDD ladder filters IEEE Transactions on circuits and systems, vol. CAS 31, n° 4, april 1984, p. 369-381

[9] Programs for Digital Signal Processing. Digital Signal Processing Committee IEEE ASSPS.
Chapitre 6.1 Program for the Design of Recursive Digital filters.

[10] J. H. MAC CLELLAN, J. W. PARKS and L. R. RABINER
A Computer Program for Designing Optimum FIR Linear Phase Digital Filters
IEEE Trans., on Audio and Electro acoustics, vol. AU 21, n° 6, p. 506-526, décembre 1973

[11] P. DUHAMEL et P. LESCAN
Deux programmes d'aide à la conception des filtres numériques OPTZ et ANA-Z. GRETSI 1977 p. 96

[12] S. S. LAWSON et A.G. CONSTANTINIDES
On the efficient analysis of digital filter structures in the frequency domain.
Proceeding of the Florence Conference on digital signal processing, September 11-13 1975

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REAL-TIME IMPLEMENTATION OF NONRECURSIVE POLYPHASE FILTER BANKS ON THE GENERAL-PURPOSE DIGITAL SIGNAL PROCESSOR FUJITSU MB 8764

Walter Kellermann* and Herbert Klump**

* Technische Hochschule Darmstadt, Institut für Netzwerk- und Signaltheorie, Fachgebiet Theorie der Signale, Merckstr. 25, 6100 Darmstadt , FRG.

** AEG, Communications Division, Eberhard-Finckh-Str. 11, 7900 Ulm, FRG.

In this paper a real-time implementation of an analysis-synthesis system of polyphase filter banks on a digital signal processor is presented. After a brief discussion of the analysis-synthesis system, programming is illustrated considering general ideas and special features of the signal processor related to the subject. Finally, the system is evaluated with respect to processing load and filtering quality.

1. INTRODUCTION

General purpose digital signal processors (DSPs) proved to be helpful tools for the development and evaluation of new methods for real-time signal processing. In that context, the implementation of digital filter banks is useful not only for short-time spectral analysis but also for certain narrow-band applications in communications where subband approaches have been proposed (e.g. speech coding [1,2,3], noise cancelling [4] or compensation of acoustical echoes [5]).

In recent years, two filter bank concepts gained importance in digital signal processing: quadrature mirror filters (QMF)[6,7] and polyphase filter banks [8,9], both of them well-understood as special forms of multirate network structures [10]. While flexibility of QMFs concerning decimation factor and subband processing is limited, polyphase filter banks are more general in this respect, and, therefore, are well suited for adaptive control and filtering in subbands.

2. THE ANALYSIS-SYNTHESIS SYSTEM OF POLYPHASE FILTER BANKS

In figure 1 a block diagram of the realized system is shown. After lowpass filtering the analog signal x(t) is sampled and the discrete-time signal x(k) is transferred to the DSP. There, x(k) is filtered by a polyphase filter bank for the analysis (FBA) leading to N subband signals $x_n(1)$ (n=0(1)N-1) that are complex in general. Due to their reduced bandwidth, their sampling rate is also reduced by a factor of r (r≤N). Since $x_{N-n}^*(1)=x_n(1)$ is valid (* denotes conjugate complex) the subband signals $x_{N-n}(1)$ (n<N/2) can be reconstructed at the

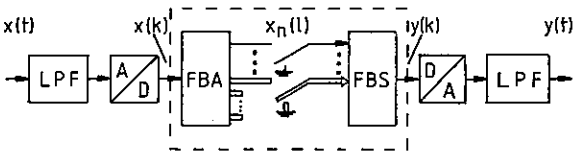


Figure 1: Block diagram of implemented system

input of the synthesis filter bank (FBS). Thus, every pair of subband signals $x_n(1)$ and $x_{N-n}(1)$ (n=0(1)N/2) can be masked before entering the synthesis part by masking $x_n(1)$. The FBS interpolates the subband signals and the DSP transfers the full-band signal y(k) to the D/A converter at the same sampling rate as x(k) is fetched from the A/D converter. The D/A unit - together with an analog lowpass filter - converts y(k) into a continuous-time signal y(t) again. In the case of ideal filter banks, x(k) and y(k) should be identical as long as there are no subbands masked.

The structure for the analysis polyphase filter bank used here is derived from a conventional filter bank consisting of N complex nonrecursive bandpass filters that are modulated versions of a prototype lowpass filter (figure 2):

$$h_n(i) = h(i) * \exp(-j2\pi*i*n/N) ,$$

where h(i) and $h_n(i)$ are the impulse responses of the prototype lowpass filter and the bandpass filter, respectively. According to the normalized stopband frequency Ω_S of the prototype lowpass filter the complex output signals of the bandpass filters can be decimated by a factor of r (with $r \leq \pi/\Omega_S$ to avoid subsampling).

In order to save storage and multiplication/accumulation operations, in a polyphase network the filtering is done only once every r

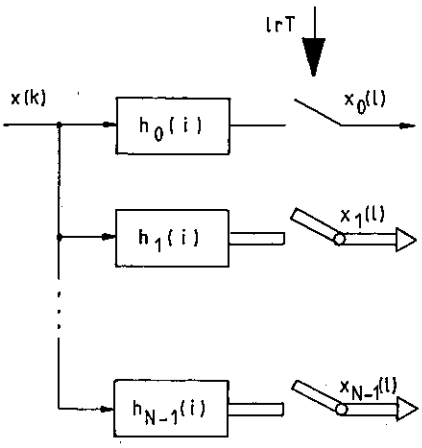


Figure 2: Conventional bandpass filter bank

sampling intervals T. To get the same output signals as in figure 2, the state variables of the filter have to be summed up appropriately and conveyed to an IDFT, where the 'polyphased' partial sums are modulated and summed up to yield the subband signals $x_n(l)$ (figure 3). The synthesis filter bank consists of an IDFT followed by the network transposed to the analysis polyphase network (for transposed networks see [10]).

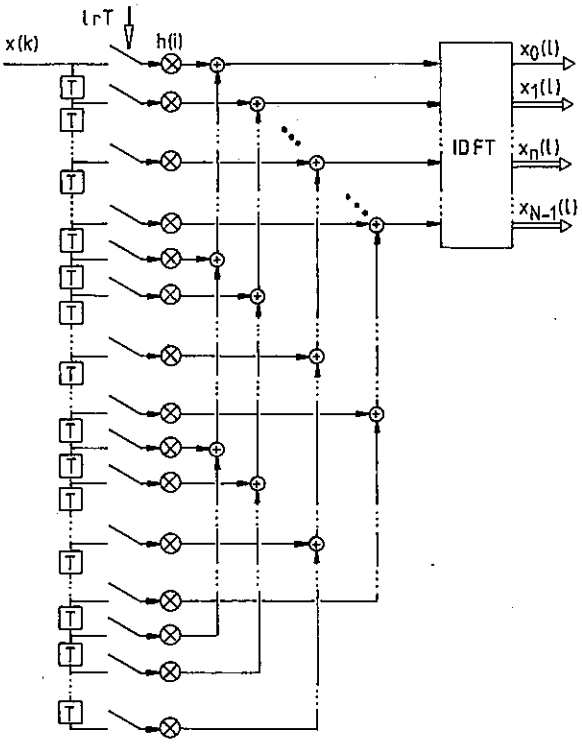


Figure 3: Polyphase filter bank for analysis

3. IMPLEMENTATION ON THE DIGITAL SIGNAL PROCESSOR MB 8764

As to be seen from figures 1 and 3 the main tasks to be performed by the DSP are nonrecursive filtering and inverse discrete Fourier transform. Additionally, the input data have to be fetched from the A/D converter and the output data must be transferred to the D/A converter. This leads to a program structure as shown in figure 4. After a start-up procedure, the succeeding program is running until stopped externally. In compliance with figure 3 it must be carried out only once for a block of r new input data.

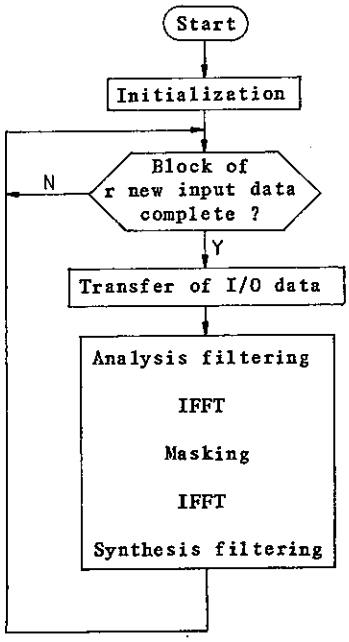


Figure 4: Structure of implemented programs

This straightforward structure was chosen for a set of programs which all realize analysis-synthesis filter bank systems and differ mainly in the number of subbands N, that is, in the dimension of the Fourier transform. Within a single program the sampling rate reduction r and the pattern of masked subbands are variable, while the coefficients of the prototype lowpass can be exchanged by changing a section of the instruction code.

To what extent the DSP MB 8764 is apted for the demands of the analysis-synthesis system, is now briefly discussed by means of a simplified block diagram of the DSP (figure 5). The silicon-gate CMOS technology applied allows operation of the DSP at an internal frequency of 10 MHz, that means, one cycle requires 100 ns. Among other input and output modes the processor offers direct memory access for data input.

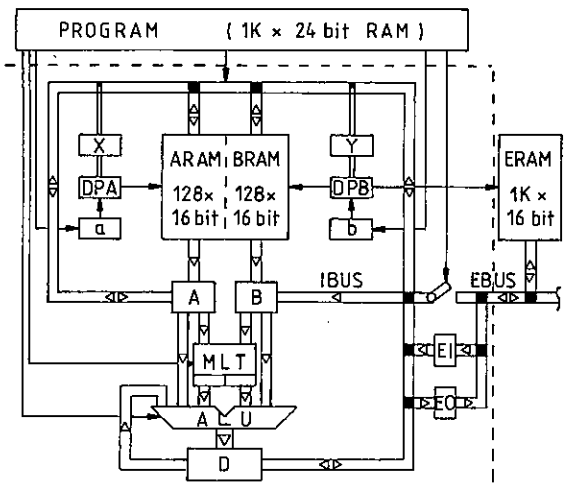


Figure 5: Simplified block diagram of MB 8764

Interrupt facilities, however, are not provided. As for the instruction code, arithmetic/logic instructions can be combined with most of the other instructions to give one instruction word. Loops should be avoided, since conditional jumps require three cycles while other instructions need only one or two cycles. For addressing the data RAMs (IRAM consisting of ARAM and BRAM, ERAM) two independent index registers X,Y are provided that, nevertheless, can be manipulated in parallel and allow -together with an additional offset- fast access to all available memories. The arithmetic/logic block features a pipeline architecture so that the 16 x 16 bit multiplier (MLT) and the arithmetic/logic unit (ALU) can be executing while the result of the preceding execution appears at the 26 bit D register. Thus, the DSP is especially well suited for nonrecursive filtering allowing one multiplication/summation operation every 100 ns.

For the implementation of the IDFT a radix-2 non-in-place FFT algorithm for real valued data [11] has been selected to take advantage of the real character of the analysis input and the synthesis output. Nevertheless, pipelining cannot be exploited optimally due to some necessary address calculation for the operands within a butterfly.

In filtering and FFT usually two problems caused by fixed-point arithmetic arise: roundoff errors and overflow control. Since measures against the well studied roundoff errors of the intended system [12,13] increase programming complexity considerably, additional processing had to be limited to overflow control('block floating').

To uncouple the analysis-synthesis process from input and output of data, the DMA facility has been expanded by some dedicated hardware, so that data output can be accomplished by DMA as well. For that, r words are reserved for the I/O data within the IRAM (I/O buffer). After having received a new input sample, the DMA logic first transfers the next output sample to the D/A converter and replaces it within the I/O buffer by the new input sample. Thus, real-time I/O is controlled by the ADC's sampling rate. As soon as the next block of r new input data is available it is copied to an analysis ring buffer before the output data are moved from the synthesis ring buffer to the I/O buffer. Now the analysis-synthesis procedure starts again. Therefore, a lower bound for the sampling rate reduction r is determined for real-time processing, since the running time of the analysis-synthesis procedure must be less than r sampling intervals.

4. RESULTS

Program versions according to figure 4 have been implemented for n covering all powers of two up to 128, while for the prototype lowpass of the polyphase networks up to 256 coefficients are allowed. Thereby, for analysis and synthesis the same coefficients are used.

Evaluation was carried out with regard to processor load and filtering quality for a sampling rate of 8 kHz. From table 1 storage load is to be read for various values of N and 256 coefficients (in %). For N=2 real-time requirements cannot be fulfilled with 256 taps, therefore a program version with 128 taps is entered (*). Same conditions hold also for table 2 where the maximum running times of the programs for one block of r=N input data are listed as well as the minimum decimation factor (see above).

N =	2*	4	8	16	32	64	128
PRAM	97	95	98	95	93	93	98
IRAM	9	13	19	35	67	77	100
ERAM	25	50	50	50	50	56	77

Table 1

N =	2*	4	8	16	32	64	128
t/µs	140	307	348	461	700	1550	4400
r_{min}	2	3	3	4	6	12	33

Table 2

For the evaluation of filtering quality a prototype filter design was chosen that provides for an ideal analysis-synthesis frequency response in case of infinite precision floating-point arithmetic [14]. The design procedure is based on an initial Dolph-Chebyshev lowpass filter with stopband frequency $\tilde{\Omega}_s$ from which the prototype lowpass filter with stopband frequency $\Omega_s = \tilde{\Omega}_s + \pi/N$ is derived. Therefore, the maximum decimation factor r within

the subbands is limited to $r = \pi/(\pi/N + \tilde{\Omega}_s)$ if aliasing of passbands is to be excluded.

Regarding figure 6, an example of a system according to figure 1 with N=128 channels and a prototype filter length of 128 taps can be examined. The analog lowpass filters of figure 1 have been replaced by bandpass filters of degree 10, whose characteristics are responsible for the smooth curvature in the passband of the magnitude response displayed. To demonstrate the frequency-selective properties of the system, groups of subbands (from one to four pairs of subbands) have been masked. Using a relatively small decimation factor r=48 no aliasing can be registered within the 4 kHz band in spite of a comparatively modest stopband attenuation of the initial lowpass (32 dB). Larger decimation factors, higher stopband attenuations, and/or smaller transition bands can be achieved by that system by increasing the length of the prototype filter without changing the results in principle.

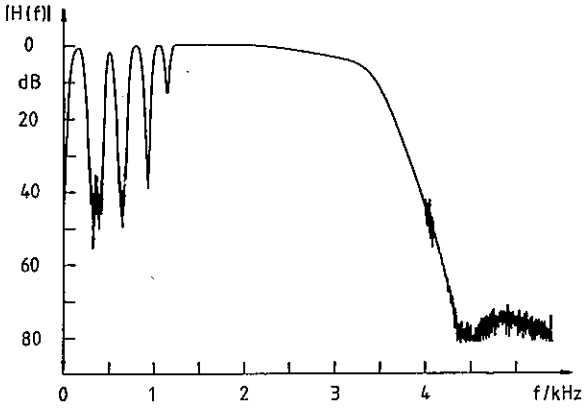


Figure 6: Magnitude response of a system according to figure 1

ACKNOWLEDGEMENTS

The authors thank J. Cezanne for inspiring discussions and W. Rohrmoser for assistance of many kinds. They are especially indebted to Prof. A. Papoulis, Prof. E. Hänsler, and Th. Becker for helpful suggestions and for carefully reading the manuscript.

REFERENCES

[1] Crochiere,R.E., Webber,S.A., and Flanagan, J.L., Bell Syst. Techn. Journ.,(1976)1069.
[2] Tribolet,J.M., and Crochiere,R.E., IEEE Trans. on ASSP, (1979)512.
[3] Honda,M., and Itakura,F., IEEE Trans. on ASSP, (1984)465.
[4] Vary,P., Signal Processing, (1985)387.
[5] Kellermann,W., Frequenz, (1985)209 (in German).
[6] Galand,C.R., and Nussbaumer,H.R., IEEE Trans. on ASSP, (1984)522.
[7] Jain,V.K., and Crochiere,R.E., IEEE Trans. on ASSP, (1984)353.
[8] Bellanger,M.G., Bonnerot,G., and Coudreuse,M., IEEE Trans. on ASSP, (1976)109.
[9] Vary,P., and Wackersreuther,G., AEÜ, (1983)29.
[10] Crochiere,R.E., and Rabiner,L.R., Multirate Digital Signal Processing (Prentice Hall, Englewood Cliffs, N.J., 1983).
[11] Brigham,E.O., The Fast Fourier Transform (Prentice Hall, Englewood Cliffs, N.J., 1974).
[12] Heute,U., Errors in DFT and FFT, (Select. Publ. on Communic. Syst. 54, Ed. H.W.Schüssler, Erlangen, 1983, in German).
[13] Heute,U., Signal Processing, (1984)119.
[14] Wackersreuther,G., AEÜ, (1985)123.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMPLEMENTATION OF RECURSIVE LEAST SQUARES IDENTIFICATION ALGORITHMS ON THE TMS 320

K. KASSAPOGLOU and P. HULLIGER

Ecole Polytechnique Fédérale de Lausanne
16, Chemin de Bellerive
CH-1007 Lausanne, Switzerland*

Abstract. The implementation of conventional recursive least-squares identification algorithms on the TMS 320 digital signal processor is described. The method used progressively transforms the algorithms from their high-level language formulation down to optimal linear assembler code. Correctness as well as optimal scaling are thus achieved. Results obtained in the context of an adaptive control application are also reported in the paper.

1. INTRODUCTION

Recursive system identification is required in most adaptive processing, for example adaptive signal processing or adaptive control. System parameters have to be identified on-line, given system input and output. The present work is done in the context of numerical control of machine-tools. These are modelled by pole-zero systems. Identified machine-tool parameters are used to adjust regulator parameters. Thus identification must be accurate and robust. Furthermore, in these applications there is a severe real-time constraint. Since the computational time per recursion must be low, fixed-point implementations are considered.

In this paper, we present a method of developing an algebraic floating-point algorithm towards a fixed-point implementation. This method has the advantage of solving most implementation problems at the highest possible level.

In section 2 we briefly present the considered algorithms. We then focus on problems that arise when fixed-point implementations are considered (section 3). The algorithm transformation method is described in section 4, followed by considerations on the TMS 320 implementations. Finally, in section 5, we report some results registered when identifying a real system.

2. THE ALGORITHMS

Several adaptive least-squares algorithms are given in the literature [1]. We select a cross-section of them and evaluate them according to their computational complexity and numerical properties.

Keeping the adaptive control application in mind, some remarks can be made on computational complexity: since the number of system parameters is low (usually 3 to 5), there is no significant difference between an $O(p)$ and an $O(p^2)$ algorithm. Thus, the conventional algorithms (including square-root factorization forms) are not penalized when compared to fast Kalman algorithms and to lattice forms.

Some earlier work on fixed-point implementations has been performed. Ling et al. [2] studied the numerical stability and accuracy of least-squares algorithms. Ljung et al. [3] also studied round-off error propagation for these algorithms. The conventional recursive algorithms and its square-root UD factorization are stated to have poor numerical accuracy when the word length is short, but to be numerically stable as long as an exponential weighting factor with value less than 1 is used. The fast Kalman-type algorithms may have good numerical accuracy when short data records are involved, but are numerically unstable and therefore not suited to continuous adaptation applications. Lattice forms are reported to be numerically stable and to have better accuracy than conventional algorithms.

Given what we have stated above, the fast Kalman algorithm is not considered any longer. Since the identification block is to be included in a larger control loop, lattice forms were not a first choice, although they promise very nice features (actually, they identify reflection coefficients instead of transfer function ones).

* This project is supported by the Fonds National de Recherche Scientifique Suisse, that is here gratefully acknowledged

We concentrated on the conventional recursive least-squares algorithm (CLS) and on its UD factorization form (UDLS). The CLS algorithm is given by:

$$\theta(k) = \theta(k-1) + L(k)[y(k) - \theta^T(k-1)\,\varphi(k)]$$

$$L(k) = \frac{P(k-1)\,\varphi(k)}{\lambda + \varphi^T(k)\,P(k-1)\,\varphi(k)} \qquad (1)$$

$$P(k) = 1/\lambda\ [\ P(k-1) - \frac{P(k-1)\,\varphi(k)\,\varphi^T(k)\,P(k-1)}{\lambda + \varphi^T(k)\,P(k-1)\,\varphi(k)}\ ]$$

where θ is the parameter vector, φ the observations vector, y(k) the system output at time k. The UDLS algorithm factors the covariance matrix P as:

$$P(k) = U(k)\,D(k)\,U^T(k) \qquad (2)$$

D being diagonal and U upper triangular with 1's along its diagonal.

These two algorithms work correctly provided that the following two conditions are fulfilled: The covariance matrix P has to be initialized as a diagonal matrix:

$$P(0) = \delta I \qquad (3)$$

where δ is large enough (the value of δ witnesses how much confidence we have on initial parameter values). Secondly, the control signal has to be frequency-rich and bear a certain amplitude level in order to allow significant adjustments on current parameter values.

3. FIXED-POINT CONSTRAINTS

When a fixed-point implementation is considered, numerical problems arise in addition to the algebraic ones.

First, dynamic range is significantly reduced, compared to the dynamic range available in a floating-point representation. Of course, we will use scale factors in order to adjust numbers to their optimal representation. Overflows must be prevented, underflows must be avoided. Let us remember here that the identification algorithms require a large dynamic range.

Next, accuracy is affected. In a fixed-point representation, all precision bits are not significant (leading zeros do not bear any information other than scaling). Scale factors should be thought of as a way to guarantee as much accuracy as possible. But, in no case should any on-line tests be made on variable values in order to improve the scale factors.

From what is described above, we can see that scaling is a difficult issue in the fixed-point implementation procedure. It is therefore desirable to be able to work on it separately. We do this in the method described below.

4. ALGORITHM IMPLEMENTATION

4.1. The method

We strongly believe that code generation cannot be efficiently performed in one step from high level algorithms. We insist on proceeding by small steps in order to deal separately with problems that are independent. Four different levels seem necessary to us. At each transformation from one level down to the next one, more specific information must be added. All transformation images must be compilable. They must be executed in order to make sure the latest information added is optimal. Development is processor independent down to the last level. The four steps are the following:

a. First, the algorithms are programmed in a high level language (PASCAL), in a floating-point representation. Convergence problems (initialization issues, dependence upon control signal, etc.) are examined and solved here. The dynamic range of variables is also evaluated.

b. For execution speed purposes, linear code is used whenever possible [6]. All loops to be unrolled are linearized at this step. An execution of the obtained code will confirm that the algorithm was not altered.

c. Fixed-point representation is introduced. The number of bits, N, is chosen. Scale factors are assigned to variables. The algorithm is simulated both in floating-point and N-bit fixed-point representation in order to observe how the original algorithm performance is affected. For this purpose high level language (PASCAL) routines were developed that simulate all basic operations (addition, subtraction, multiplication, division, negation, etc.) in two's complement arithmetic. Incorrect operation (overflow/underflow occurences, scaling incompatibility during addition) is detected. The floating point representation is used only for comparison purposes. No interaction is allowed between these two parallel runs. This way, we can evaluate and/or modify the number of bits, N and, more importantly, the scaling assignment.

d. Assembler code generation is not a compiler-like transposition. We prefer to first decompose the linear code algorithm into elementary operations. A set of assembler instructions with known characteristics (execution time, number of required memory cells, precision) corresponds to each of them. Elementary operations need be as global as possible in order to reduce memory transfers. They must also make the best use of specific processor facilities. This is the only

reason why the elementary operations may depend on the kind of processor used. Their determination is done in a heuristic way and may be improved after performing several iterations. Even if we cannot guarantee obtaining optimal global code with this method, the obtained one is close to optimal.

4.2. Data related to the fixed-point implementation of the selected algorithms

Linear code is always used within the time recursion, except for the implementation of division.

As far as precision is concerned, we limited our investigation to 16-bit words (including the sign bit). This was mostly directed by the structure of the processor used. Note that numbers were systematically rounded instead of being simply truncated. We noticed an improvement in convergence by doing this.

The set of scale factors we found to be optimal for the CLS algorithm is given in Table 1. The UDLS algorithm was found to be more sensitive to non-optimal scaling. Updating of diagonal matrix D is done using a multiplication (and not an addition/subtraction) and therefore, no occurence of underflow is tolerated. Moreover, this algorithm involves a recursion depending on parameter order within the observations vector φ. So, all parameters are not identified with the same accuracy. It turned out that the fixed-point implementation of this algorithm was not as promising as its original algebraic properties when a noisy system is identified. On the contrary, the CLS algorithm worked correctly on the models we used.

Table 1. Scale factors for the CLS algorithm.

Variable i	Number of bits assigned to its fractional part, f_i
$\varphi(k)$	$f_\varphi = 14$
θ	$f_\theta = 12$
P(k)	$f_P = 8$
λ, $1/\lambda$	$f_\lambda = 12$
$v = P(k-1)\,\varphi(k)$	$f_v = f_P + f_\varphi - 15 - E[\log_2(2d-1)]$*
$b = \varphi^T(k)\,v$	$f_b = f_v + f_\varphi - 15 - E[\log_2(2d-1)]$
ib = 1/b	$f_{ib} = 15 - f_b$
$w = ib\ v$	$f_w = f_{ib} + f_v - 10$
$\Delta P = v\ w^T$	$f_{\Delta P} = f_v + f_w - 5 = f_P$
$y,\ ye = \theta^T\varphi$	$f_y = f_\theta + f_\varphi - 15 - E[\log_2(2d-1)]$
$\Delta\theta = ye\ w$	$f_{\Delta\theta} = f_y + f_w - 5 = f_\theta$

* E[x] represents the entire part of x, d is the number of parameters

4.3. Implementation on the TMS 320

The processor to be used is not important until the last step of our procedure is reached. It this step, the choice of processor may affect how elementary operations are chosen, and of course, it determines the code that is finally generated.

We used the Texas Instruments TMS 320. In spite of its nice features (speed of computations, low price, etc.), it presents some serious limitations especially as far as bit handling and data processing are concerned.

The execution time we obtained for both algorithms as well as their memory requirements are given in Table 2.

Table 2. Execution time and memory requirements (for 4 parameters)

	CLS	UDLS
Cycles/Time	433 / 86.6μs	636 / 127.2μs
Data memory 16 bit words	34	42
Program memory 16 bit words	415	485

5. A REAL SYSTEM IDENTIFICATION

5.1. System description

The work presented in this paper aims to integrate identification algorithms into an adaptive control block designed for fast processes. Therefore, the CLS identification algorithm was tested on a testbed that simulates all components of a modern machine tool. This testbed is composed of a table mounted on pre-stressed low friction guide-ways, driven by a 1.5kW hydraulic motor using a ball bearing screw. An incremental 0.5μm resolution sensor is used to measure table position. Hydraulic motors are very well suited to machine-tool applications, because of their excellent dynamic performance and their small dimensions. However, they require more sophisticated control signals than DC electric motors do, in order to best exploit their performance. In this case, an identification algorithm allows the user to automatically adjust the optimal regulator parameters on-line, even if these are time-varying. This simplifies the bootstrap of such a system. The testbed is driven by a series of pseudo-random commands. Input-output pairs are entered to the algorithm which identifies system parameters.

5.2. Experimental results

The obtained results are quite satisfactory. The model used has two poles (a_1 and a_2) and two zeros (b_0 and b_1). Identified parameter values are shown in Figure 1. The testbed static gain, which is known, is approximated extremely well (2.997 instead of 3).

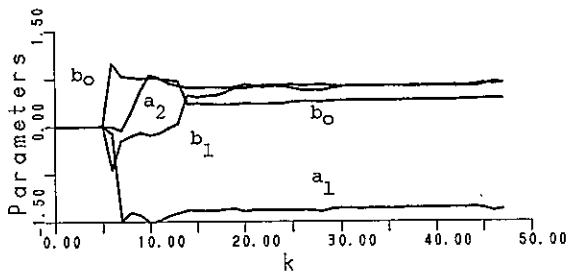


Figure 1

Figure 2 compares the output y_m (thick line) of the identified system, the parameters of which are the ones obtained at the last step of the identification procedure, to the initially observed output y_r (thin line). The control signal u is also represented. The average relative prediction error, computed ignoring the six first output samples, is 8%. The divergence observed in the first few steps is due to the fact that the system is on transient response and, in this case, there are some static friction components, that are not taken into account by our steady-state model.

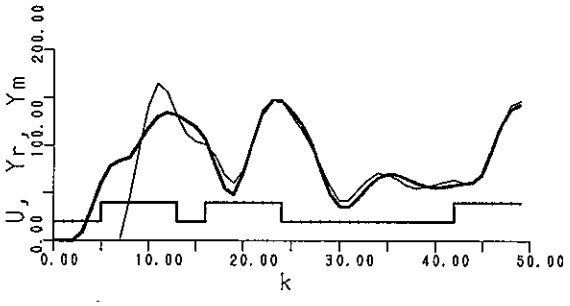


Figure 2

6. CONCLUSIONS

In this paper we have presented a method of developing algebraic floating-point algorithms towards their fixed-point implementation: starting with high level code, we obtain assembler linear code which is close to optimal. We transform the code in four separate steps, each step dealing with a separate problem. The transformations are all done at the highest possible level.

We have applied this method to the conventional recursive least-squares algorithm and to its UD factorization form. We aim at implementing an adaptive control application, where algorithmic stability is of highest relevence. The number of system parameters is low in these applications. Therefore, criteria on how to choose algorithms lead to different conclusions than in common signal processing applications. Both algorithms are successfully implemented on the TMS 320 digital signal processor. However, the UDLS algorithm is found to lack robustness when identifying a noisy system. Precision was held at 16 bit words including the sign bit. We have obtained optimal scale factors for our application.

Finally a real system was identified in an open loop configuration. The results obtained for convergence and execution time are quite satisfactory.

REFERENCES:

[1] L.Ljung and T.Soderstrom, "Theory and Practice of Recursive Identification", MIT Press, Cambridge, Mass., 1983.

[2] F.Ling, D.Manolakis and J.G.Proakis, "Finite Word Length Effects in Recursive Least Squares Algorithms with Application to Adaptive Equalization", Proceedings of the GRETSI Conference, Nice, May 1985.

[3] S.Ljung and L.Ljung, "Error Propagation Properties of Recursive Least-Squares Adaptation Algorithms", Automatica, vol.21, No 2, pp.157-167, 1985.

[4] D.W.Lin, "On Digital Implementation of the Fast Kalman Algorithms", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-32, No.5, pp. 998-1005, Oct. 1984.

[5] R.Alcantara, J.Prado and C.Gueguen, "Les algorithmes des moindres carrés récursifs rapides complexes", Proceedings of the IASTED Conference on Applied Signal Processing and Digital Filtering", Paris, June 1985.

[6] L.R.Morris, "Automatic Generation of Time Efficient Digital Signal Processing Software", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol. ASSP-25, pp.74-78, Feb. 1977.

[7] M.Vetterli, E.Debourse, M.Kardan, "Fast Fourier Transforms on the TMS 320 Signal Processor", Proceedings des Journées d'électronique 1985, Lausanne, Suisse, Oct. 1985.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DIGITAL FILTER DESIGN, SIMULATION AND EVALUATION SOFTWARE FOR PC BASED SYSTEMS

R. N. Zobel

Department of Computer Science, University of Manchester
Oxford Road, Manchester, England

A suite of software modules is presented for the design, simulation and evaluation of finite impulse response and infinite impulse response digital filters on a small personal computer. A graphics module allows display and hard copy of both time and frequency responses in a number of formats. A digital software test waveform generator and time and frequency domain convolution modules enable filter performance to be assessed. The package is used both to assist filter design and filter hardware development and to support teaching of digital signal processing.

1. INTRODUCTION

The advent of personal computers (PCs) and work stations has brought significant computer power to the desk of the designer, particularly where the CPU has a maths co-processor on board. Traditionally both finite impulse response (FIR) and infinite impulse response (IIR) digital filters have been designed on mainframe or larger mini-computers where both speed and software support are good. However such facilities, whilst adequate for program development, often leave much to be desired in terms of access, user interaction, and of screen and hard-copy graphics. A further problem is associated with the difficulty of interfacing subsequently designed hardware for commissioning, testing and evaluation purposes. The latter is vital due to the complexity of digital filter systems which are often programmable, multi-channel and multi-sample rate. Many PCs now provide at least good if not excellent software development support and adequate main and backing storage. This, coupled with medium resolution graphics and single user interaction, provides, at least in principle, a useful stand-alone facility on which digital filters may be designed and evaluated.

In developing the suite of software modules described in this paper two quite separate aims were specified. The first was to determine the limitations of a typical PC (ACT Sirius 1) in terms of speed and storage relative to the filter performance required, particularly with respect to parameter optimisation techniques. The second was to evaluate the usefulness of the resulting package, both for actual filter design and for teaching purposes, and in the process to assess the efficiency of the man-machine interface.

2. Overall Software System Concept

Several basic high level operations are required to permit design, test and evaluation of digital filters. The approach adopted here is that of providing self contained modules that are only loosely coupled. This aids software development as each code module is then of a size which is easy to comprehend and if, as in this case, the resulting suite is developed by a team and is extended during its life, changes and errors in one module have little effect on other modules. Because most of the modules are not required to work in real-time (this aspect is covered by a high speed buffer system for hardware testing) the coupling need be neither sophisticated nor fast. The simplest method, and the one adopted here, is have each module require files and menu driven keyboard commands as input, which has the further advantage of a potentially friendly and efficient user interface. All modules are written in Pascal to facilitate porting to other systems with perhaps different operating systems than that employed (CPM/86). The modular concept also offers a solution to the problem of differing screen and printer graphics facilities by isolating these to machine specific modules.

It was clear from the outset, that representation in both time and frequency domains was essential, since digital filters are mainly implemented in time domain form and understood in the frequency domain. Hence both forward and inverse fast Fourier transforms (FFTs) are provided. To avoid end effects a Blackman-Harris window is employed prior to transformation. This is a default option and the user may input and specify other windows. For the time domain evaluation of FIR filters it is necessary to implement convolution of an input waveform with a filter impulse response to produce an output filtered waveform, and hence simulate the operation of the filter. In the frequency domain the corresponding operation is reduced to that of complex conjugate multiplication. Both methods operate on two user specified data files, producing an output file named by the user. Figure 1 illustrates the general structure. It is worth noting that although specified in terms of filters the technique is general, a point that will be returned to later.

3. Graphics Facilities

Data arising from signal processing in general and filters in particular are best presented in graphical form and this has become increasingly popular with the advent of

low cost graphics of medium but adequate resolution associated with PCs and work stations. Waveform graphics have rather different requirements from general graphics although such a module can readily be built around basic graphic facilities such as those provided in a Graphic Kernal System (GKS) implementation. GKS was not available initially and the existing module was written in Pascal.

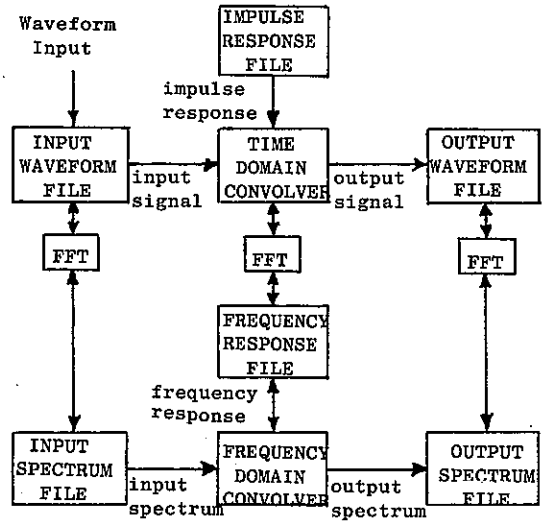


Figure 1. CORE MODULAR SOFTWARE SYSTEM.

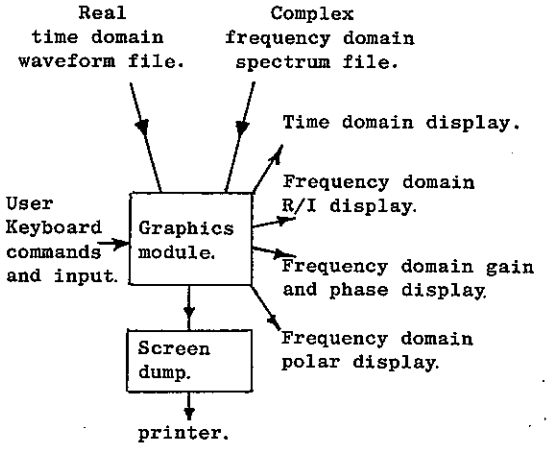
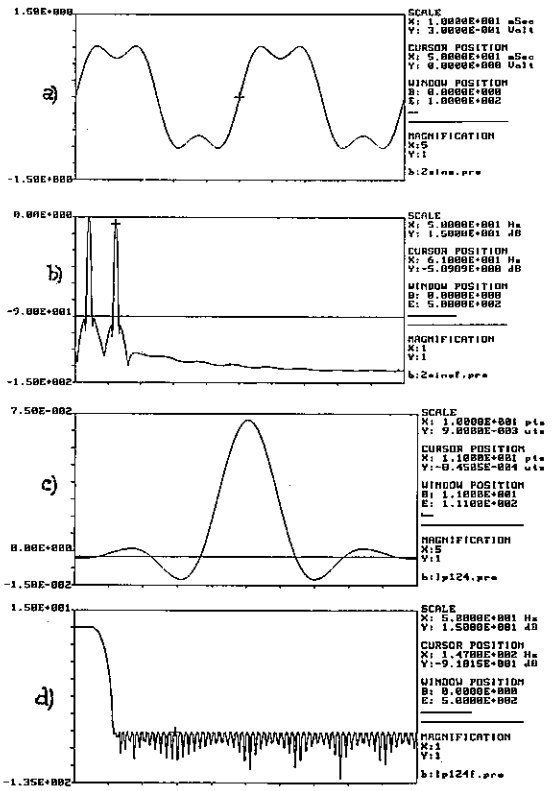


Fig.2 GRAPHICS MODULE.

What then are the specific requirements? Firstly several screen formats are needed. Time domain waveforms are real and fairly straightforward, although there are both addressing and scaling problems. Frequency domain waveforms are more difficult. Fourier transform results are in cartesian form with real (cosine) and imaginary (sine) components. This indicates a requirement for two waveforms with a common frequency axis. Further processing yields magnitude and phase, presented in the same way, but much more useful. However there are problems. Filter pass bands are conveniently studied relative to linear gain, but transition regions and stop bands are only meaningful if expressed on a logarithmic (decibel) scale. A further problem is the wide range of frequencies to be covered and the display is normally presented on a logarithmic screen scale. The final format is a truly polar representation, such as that for a Nyquist diagram. All of these formats are provided and figure 3 illustrates some of these and other features now described.



a) Input Waveform b) Spectrum of Input Waveform
c) FIR Filter Impulse Response d) Frequency Response
Fig.3. Graphics Facilities.

Let us return to the problems of address and scaling. A typical screen might have a resolution of 512 points (pixels) horizontally, and clearly many files will be much longer than this. A horizontal scroll with an on-screen indicator of position within the overall file is provided. Horizontal expansion (zoom) is also useful. The vertical direction has other problems. Firstly the data will be processed and stored as a floating point file which clearly cannot be directly displayed. Every file to be displayed or plotted must first be converted to fixed point format but even this has too much resolution. The solution is to employ fractional fixed point representation, and to multiplex a short word from this, conveniently a byte for 256 point vertical resolution, depending on the required magnification. The top 8 bits is selected by default. The screen processing must also avoid end-around folding of the waveform in the vertical direction. Vertical scrolling is also provided.

Textual information outside of the graphical area is used to indicate scaling, scrolling, and other helpful data stored in an separate file with the same root name. A graphic cursor with numeric screen read-out is used to

indicate precision waveform values in floating point format. The cursor will follow the waveform using the horizontal cursor keys, and maxima and minima found within a user defined region. All of these features are independant and a screen dump may be made at any time.

4. Input Waveforms

The next consideration is the production of input waveforms for filtering. Three possibilities are provided. The first is software generation of well known useful waveforms. Predictably these include sine, square and triangle, the latter with user defined rise and fall rates. The user may specify the file length and the frequency and amplitude of the signal. A more useful waveform type is the compound sinusoid with user specified frequency, phase and amplitude for each component. This can be quite useful when evaluating aliasing effects in cascaded filter/decimators. Finally a chirp waveform with user specified rate, and start and finish frequencies is provided. Ultimately it is desirable to test filters against real input waveforms and two methods are provided for this. First any data or file which can be input via an RS232 interface from an external data acquisition or computer system can be captured and stored as a waveform file, although not in real-time unless this is relatively slow or buffered externally. Secondly instruments and systems for data acquisition which are IEEE488 compatible may be used.

5. Filter Design

Two different modules are provided for filter design. For FIR filters the Remez exchange method [1] was adopted with the usual options for the various filter types. This was a translation of the well known Fortran program into Pascal and was tested by comparison of published coefficients for given input specifications. To achieve this accuracy alternative sine/cosine routines were necessary. Run times for the program depend on the grid density, number of taps and the filter specification. Results for a grid density of 16 were around 90 to 250 seconds for a variety of filters of lengths from 19 to 63 taps, which was considered acceptable for an 8088 based machine with 8087 maths co-processor and sufficient store (256Kb) to avoid disc access after entry.

An empirical low pass formula, [2], is implemented for optimisation of a user selected parameter (N, δ_1, δ_2, f_p or f_s). Run times for the optimisation of number of taps N, was about 10 to 15 minutes. The output from this module is a coefficient text file. For evaluation purposes this much be converted to a real (impulse response) file for processing, and to fixed point representation for display purposes. Time domain filtering using this filter may then be evaluated (simulated) by specifying an input test waveform file and employing the time domain convolver, which is efficient for filters of moderate length ($\leqslant$128). Frequency responses can then be obtained by use of the FFT with optional linear or logarithmic screen representation. Effects of finite word length may also be investigated. In this way FIR filters may be designed, simulated and assessed prior to implementation in hardware, as described in [3].

IIR filters present some different problems. The Fletcher-Powell optimising IIR filter design algorithm, [4], was implemented by conversion from Fortran. Simulation of the filter is implemented in terms of the canonical structures of the filter sections again in the time domain, and the frequency response assessed by inputting sine waves of various frequencies. Some IIR filters with relatively short impulse responses for a given resolution can be simulated using the time domain convolver, but simulation times are inevitably longer than for FIR filters, this being a consequence of a compromise between accuracy and time. Finite word length effects can be investigated, along with limit cycles and coefficient sensitivity, [5]. Typical run times for a four section, eighth order low pass filter were around two minutes.

6. Hardware Evaluation

It is abundantly clear that the data rate in and out of a PC is too slow for most hardware digital filter implementations. However, hardware testing and evaluation can be usefully carried out in burst mode by use of an input/output data buffer and the hardware program-

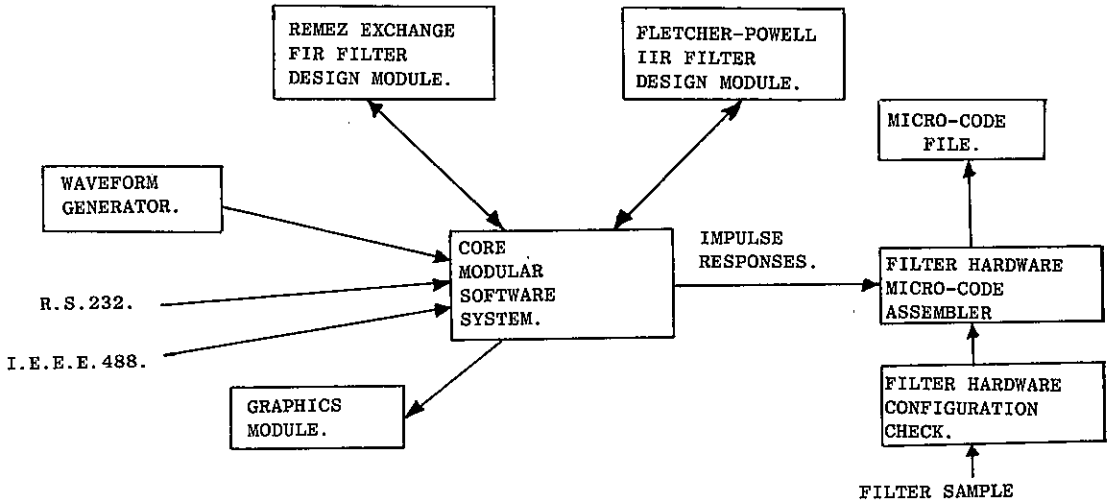


Fig.4. FILTER DESIGN AND HARDWARE SUPPORT SYSTEM.

med and initiallised via a set-up interface as in figure 4. The overall software system then appears as shown in figure 5 in which the core system of figure 1 has the waveform generator module and filter design modules added. Also featured are two new modules. The first checks whether the programmable filter hardware is capable of achieving the desired performance, in times of number of channels, sample rate, filter type and number of taps or sections. The second takes this information and the filter design module output files and assembles micro-code for the hardware. Clearly these latter two modules are necessarily filter hardware specific. When running the external tests both micro-code and data must be down-loaded, and execution in the hardware may then proceed at the design rate. The ability to plant software interrupts in the micro-code, and operate in single instruction mode, and then to read critical hardware registers or employ a logic analyser on halting helps with prototype and production debugging [6]. The buffered results of hardware filtering are subsequently input to the PC and evaluated using the package as for the simulated results.

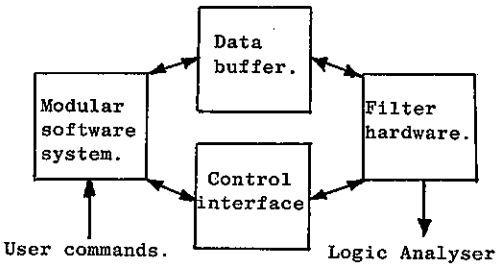


Fig.5 HARDWARE TEST SET UP.

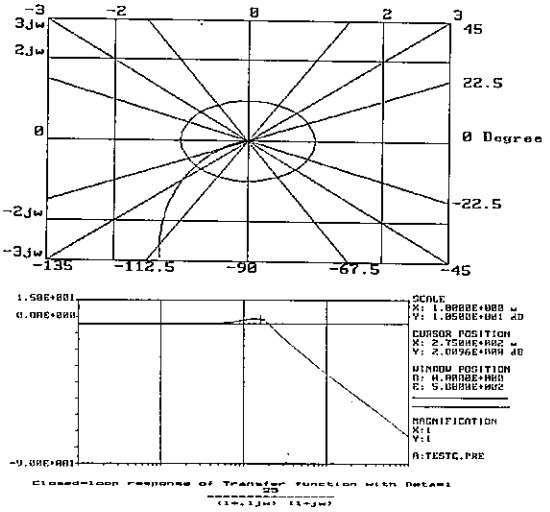


Fig.6. Nyquist Diagram and Closed Loop Responses.

7 Open and Closed Loop System Simulation

The system described above was developed for real applications but is useful in supporting teaching of digital filters both for lecture and laboratories. The core system had wider application than just for filter work. A further module was developed as a third year undergraduate project, [7], to demonstrate some aspects of simple open and closed linear systems. The user may input open loop poles and zeros as single, real, time constants and as complex pairs as required. Integrators, pure time delay, gain and feedback fraction complete the definition. Gain and phase are computed for each term and combined for system open loop gain and phase. These may be plotted either as Bode or Nyquist diagrams. Subsequently closed loop gain and phase may be calculated and plotted from the open loop files. There are restrictions on the maximum ratio of time constants, but in a teaching environment this is acceptable since it still allows useful systems to be demonstrated, as in figure 6. The closed loop frequency response file may be converted to an equivalent linear cartesian frequency file for inverse Fourier transformation to an impulse response, although again there are limitations due both to frequency range and transformation time.

8 Conclusions

The modular software system for the design, evaluation, and testing of digital filters and linear systems, performs well on a modest performance desk-top PC, provided that a maths co-processor is included. In addition to the design of digital filters, many aspects of linear systems can be demonstrated in a teaching environment by using the package in the various ways described. Such a package is useful both for undergraduate support for teaching of many DSP principles, and has also been used for teaching and support for postgraduate research projects at M.Sc level.

The speed of resonse is acceptable for most design excercises, but in a teaching situation it is helpful to pre-process some demonstrations. Later PCs and work stations have considerably enhanced performance. Most also have good colour graphics facilities. This work has shown that, in line with other CAD areas, the design, evaluation, testing, teaching and demonstration of digital filter and linear systems can be achieved through desk-top PCs and work stations at reasonable cost with considerable consequential benefits.

REFERENCES

1. Parks, T.W. and McClellan, J.H. 'A program for the design of linear phase finite impulse response digital filters.' IEEE Trans. on Audio and Electroacoustics. AU-20, No.4, pp.280-288, 1972.
2. Rabiner, L.R. 'Approximate design relationships for low pass finite impulse response digital filters', IEEE Trans. on Audio and Electroacoustics, Oct. 1973.
3. Goh, B.H. 'FIR digital filter design on a ACT Sirius 1 microcomputer', M.Sc. Thesis, University of Manchester. 1984.
4. Fletcher, R. and Powell, M.J.D. 'A rapidly convergent descent method for minimisation.' Computer Journal. Vol.6, No.2, pp.163-168, 1963.
5. Gough, S.M., 'The Design, Simulation and Evaluation of Infinite Impulse Response (IIR) Filters on a Microcomputer', M.Sc. Thesis, University of Manchester, 1985.
6. Zobel, R.N. and Tang, P.S. 'A High Performance Multi-Channel Decimating Finite Impulse Response Digital Filter System for Microprocessor Based Data Acquisition', Proc. ISCAS 85, Kyoto, Japan, 1985.
7. Cheng, K.W.W. 'Time and Frequency Analysis Software.' Report, Department of Computer Science, University of Manchester, 1985.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

GENERAL PROCESSOR APPLICATION ; CAD TOOL FOR FILTER DESIGN

Gonzalo Lucioni

Lehrstuhl für Nachrichtentechnik
Ruhr-Universität Bochum
D-4630 Bochum, West Germany

Abstract

A CAD tool is presented for the top-down design of digital filters, starting from specs and ending with the assembler program of general purpose digital signal processors (TMS 320, NEC 7720, etc.). The generation method uses macro libraries and is always matched to the actual signal processor architecture. The method will be ilustrated by an example.

I. Introduction

The recent progress in VLSI-technology makes possible the realization of real time digital networks with single chip, software programmable digital signal processors (DSP's).

The flexibility grow, accuracy and reduction in cost, size and power requirements, together with a wide application field are some of the features of DSP's techniques /1/.

These DSP's have generally a highly-parallel multi-instruction assembler set, which makes an errorless program development difficult, specially for sophisticated digital signal processing algorithms (e.g. wave digital filters (WDF's), Bergland FFT, adaptive filtering).

In this paper we present a method by means of an automatic program generation of assembler description, for DSP's as the TMS 320, NEC 7720, etc. Starting from given specifications the program generates an optimized, run-able (fit) algorithm for the actual DSP, including the operations (saturation as necesary, appropiate truncation) to manage all finite arithmetic effects.

II. General aspects

Before we actually begin with the method description, it is important to point out some aspects related to the compatibility between digital networks (described by their respective signal flow graphs) and DSP's (given by their hardware-constrained processing capabilities).

First it is well known that there exist usually several variants of the flow graph description of a digital network, satisfying a given specification set. This fact gives an optimization margin for our realization purposes. Further it is clear that the constrains, under which the network has to be realized are given by the DSP capabilities. The constrain-degree depends upon the effectiveness with which the DSP handles with the components of a real realizable digital network /2/, namely the DSP capabilities for

- handling with arithmetic operations ; it depends upon the arithmetic function(s) that the DSP performs /3/
- handling with delays ; we distinguish between seriall & parallel delay realizations or a combination thereof.(See fig. 1)
- handling with shimming delays ; since all the arithmetic operations are realized by a single arithmetic unit (though a one port arithmetic unit/4/) , there are shimming delays required for equalizing delays in different paths of the signal flow diagram
- managing finite arithmetic effects ; (depending on the binary arithmetic used).

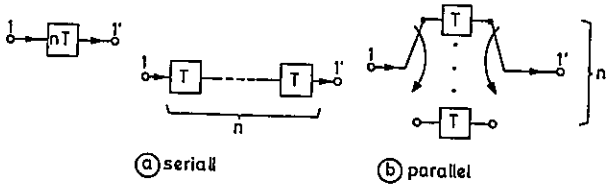


Fig.1

In addition, these capabilities are mutually engaged, due to piplineability used to increase troughput rate in the DSP's.
So we state roughly, that e.g. in order to fulfil the sampling rate requirement of a digital network we must profit by the processing capabilities of the DSP with the (for this purpose) best suited signal flow description of the network.
Clearly is the sampling rate not the only one criterion for optimizing the DSP network algorithms (though mostly the crucial point), so that other options and versions for the algorithm-realization could be important as well (see Sec.IV).

III. The algorithm generation method

Due to the aforementioned discussion we propose herefor a method based on a predesigned MACRO-set, which is optimized for the actual DSP.
The macros correspond to algorithm building blocks /5/, which are already specified in a library and may be interconnected by a pair of nodes (input and output node), thus a one port or more. The macros (e.g. in lattice WDF's elemental reactance cells) offer among other things following advantages, they

- ensure the split off the DSP specific attributes from the CAD-program, increasing so its flexibility,
- reduce the optimization amount (by reducing networks into subnetworks),
- allow modularity be effectively applied,
- avoid pipline conflicts,
- allow a differentiate optimization criterion.

The latter implies the need sometimes to make a compromise between different criterions for fulfilling the design statements.
The main disadvantage is the fact, that the macro independence from other macro structures is only guaranteed if (and only if) all its inputs are buffered (and in consequence known) at the moment where the corresponding flow path is processed. It follows, that structures not fulfilling this condition are no more independent from each other. We speak then from a sub-macro set.

IV. CAD-tool for filter design

In order to put in practice the above ideas, a WDF-synthesis program FALCON /6-8/ and an automatic program generation routine SIPRO, based on a WDF-macro set were coupled to a powerful CAD tool for the TOP-DOWN development of WD-lattice, branching and multirate filters; starting from the filter specifications and ending with the DSP-filter algorithm.
SIPRO features commercially available DSP options like the TMS 320, NEC 7720 and offers other services like canonic delay realizations (the degree of the filter is equal to the RAM locations used to implement the digital network), realizations with minimum instruction cycle amount (effectively the fastest solution), realizations with minimal I/O-delay (a constant delay, that is independent from the order of the filter and corresponds to the time which is necessary for carrying out the required computer step cycles to generate the output samples), and RAM-coefficient realizations for adjustable filtering.
It must be stressed, that the assembler program developed using the macro set is as effective as a carefully optimized handmade program.

V. Proposed macro structure

As is well known lattice, branching and multirate WDF's can be implemented with branches based on allpass sections using only two-port adaptors. This results in highly modular structures, which require a low expense on macro's, reducing program amount.
Depending on the capabilities offered by commercially available DSP's we use herefor mostly the two-port adaptor configuration shown in Fig. 2. It is clear, since the arithmetic function performed by the DSP's in discussion combine usually one multiplication with one addition (scalar product).

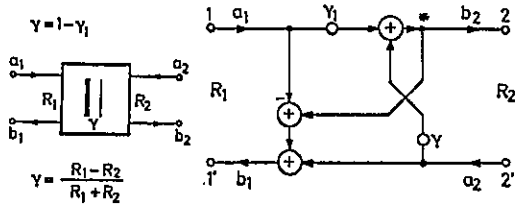


Fig. 2

We require here $R_1 >= R_2$ to avoid overflow at *, but this is no restriction since for $R_1 < R_2$ one can reduce the case to that for $R_1 >= R_2$ by simply inverting the roles of ports 1 and 2 /9/.

VI. Example

In order to demonstrate the concept of CAD for WDF based on a macro library, one numerical example will be given. The implementation will be carried out for the TMS 320 processor.

The lattice WDF-realization of a fourteenth order symmetrical bandpass filter, which could be compared with a filter /see 10 filter Nr.51/ will be considered. The design parameters for the analysed filter are:

passband:	12.3 - 15.7 kHz
max. passband att.:	.25 dB
stopband:	-11.7 16.3- kHz
min. stopband att.:	60 dB
sampling freq.:	56 kHz

The measured frequency response of the bandpass filter in Fig. 4 is shown in Fig. 3. It agrees with the theoretical one. Since we have measured directly on the output pins without using any reconstruction filter, the sinx/x distortion has appeared in the response.

The sampling frequency here is 40 kHz due to the system used for measure. In addition 12 bit ADC, DAC were used, so that the wobble signal of the spectrum analyser was amplified in order to become a reasonable SNR for the AD-,DA-converters.

Nearly optimal L_{∞}-scaling is used /7/. In order to avoid overflow at * in fig. 2 (due to the asymmetry of two's complement) a passive rather than lossless filter realization is used; (see Fig. 5). Automatic overflow correction-option is used /9/. Two's complement truncation were used, no parasitic oscillation were observed.

The total program steps are 90, including the terminating branch operation. By droping this and using interrupt branch the step size reduce to 89, which allowe a sampling rate of 56 kHz.

The modular structure can be recognized in the corresponding assembler program below.

The program correspond to the runable one implemented with the TMS 320 system developed in our institute. The assembler varies from a standard one only insignificantly.

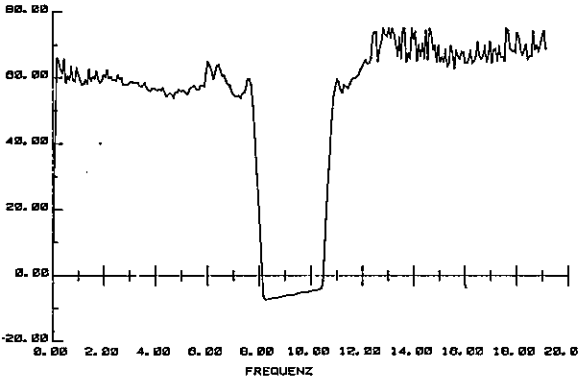


Fig. 3

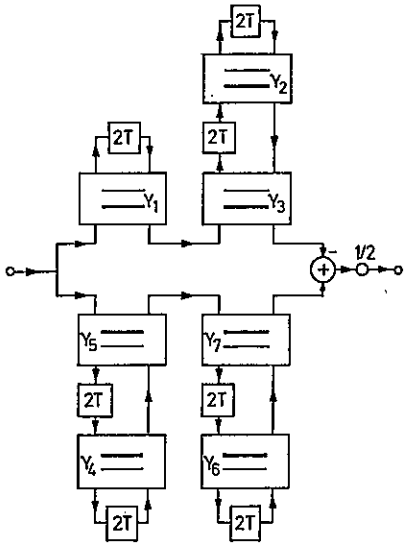


Fig. 4

Finally it is interesting to remark, that further improvements can be done in the example by avoiding seriall delay processing and using parallel delay block processing. It can be realized by reducing the filter network with delay ratio 2:1 (fig.4) in two periodically switched filter networks with a delay ratio of 1:1. This follows from a generalization of fig.1,1b, which is closely related to N-path filters /11/.

Acknowledgement

The author would like to thank Dr. L. Gazsi for stimulating suggestions, Prof. A.Fettweis for his support and to all those who contributed to this paper.

VII. Literature

/1/ T.Nishitani,et.al.:A single-chip Digital Signal Processor for telecomm.App., Aug.81,IEEE J. Solid State

/2/ A.Fettweis: Realizability of digital filter networks,AEÜ Bd.30,1976,pp.90-96

/3/ S.L.Freeny: Hardware implementation of digital filters II-signal processors, Zeitdiskr.Syst.,H.Lang&Cie.AG Bern 1980 pp. 117-132

/4/ L.Gazsi: N-Port arithmetic unit, ICASSP 82 pp. 707-710.

/5/ A.Fettweis: A general theorem for signal flow networks with app., AEÜ, Bd. 25, 1971, pp.557-561.
/6/ " : Digital filter structures related to classical filter networks, AEÜ, Bd. 25, 1971, pp.78-89.
/7/ L.Gazsi: Explicit formulae for lattice wave digital filters, IEEE trans. CAS, vol.32, Jan. 85.
/8/ R.Jain,et.al.: CAD-tools for the optimized design of custom VLSI wave digital filters, Proc. IEEE 1985.
/9/ A.Fettweis: Wave digital filtes, theory and praxis, Proc. IEEE, Feb. 86.
/10/ Ch.Kao,L.F.Kurth: Method and Apparatus for interfacing digital and analog Carrier Systems, U.S. patent.
/11/ A.Fettweis,et.al.: A solution to the balancing problem in N-path filters, IEEE trans. Circuit Theory, May 71., pp. 403-405.

```
;  ********************************************************
;  *        BWDF BANDPASS 14.GRADES SYMMETRISCH           *
;  ********************************************************
       B      INIT

       B.     FILT

FILT:  IN     DAT127,PA1

       MPYK  -3495
       PAC
       LT     DAT127
       MPYK  -0600
       LTA    DAT121
       SACH   DAT124,SH4
       ZALH   DAT124
       ADDH   DAT127
       SUBH   DAT126
       DMOV   DAT125
       DMOV   DAT124
       SACH   DAT124

       MPYK  -3855
       PAC
       LT     DAT123
       MPYK  -0240
       LTA    DAT124
       SACH   DAT119,SH4
       ZALH   DAT119
       ADDH   DAT123
       SUBH   DAT121
       DMOV   DAT120
       DMOV   DAT119
       SACH   DAT119
       MPYK  -0482
       PAC
       LT     DAT119
       MPYK  -3613
       LTA    DAT116
       DMOV   DAT122
       SACH   DAT122,SH4
       ZALH   DAT122
       ADDH   DAT124
       SUBH   DAT119
       SACH   DAT119

       MPYK  -3971
       PAC
       LT     DAT118
       MPYK  -0124
       LTA    DAT127
       SACH   DAT114,SH4
       ZALH   DAT114
       ADDH   DAT113
       SUBH   DAT116
       DMOV   DAT115
       DMOV   DAT114
       SACH   DAT114
       MPYK  -0892
       PAC
       LT     DAT114
       MPYK  -3283
       LTA    DAT111
       DMOV   DAT117
       SACH   DAT117,SH4
       ZALH   DAT117
       ADDH   DAT127
       SUBH   DAT114
       SACH   DAT114

       MPYK  -3791
       PAC
       LT     DAT113
       MPYK  -0304
       LTA    DAT114
       SACH   DAT109,SH4
       ZALH   DAT109
       ADDH   DAT113
       SUBH   DAT111
       DMOV   DAT110
       DMOV   DAT109
       SACH   DAT109
       MPYK  -0144
       PAC
       LT     DAT109
       MPYK  -3951
       LTA    DAT126
       DMOV   DAT112
       SACH   DAT112,SH4
       ZALH   DAT112
       ADDH   DAT114
       SUBH   DAT109
       SACH   DAT109

       LAC    DAT109,SH15
       SUB    DAT119,SH15
       SACH   DAT127
       OUT    DAT127,PA0

;
WAIT:  EINT
       B      WAIT

;
INIT:  ZAC
       LDPK   0
       SOVM
       B      WAIT

       END
```

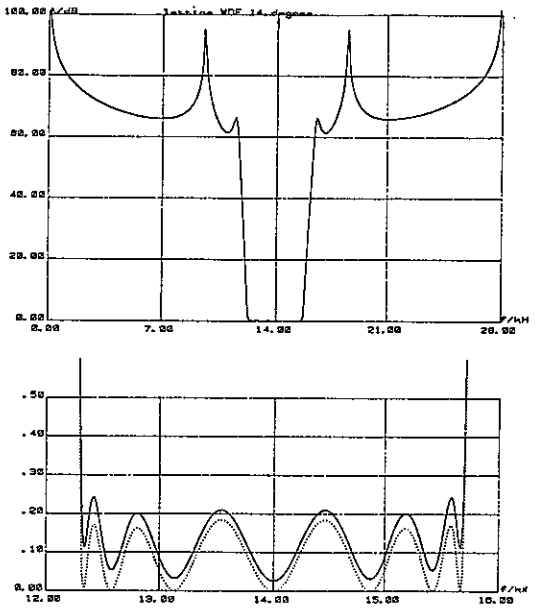


Fig. 5

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

PARALLEL PROCESSOR FOR REAL-TIME CALCULATION OF INNER PRODUCTS

Jan A van Alstê, Arjan J Mulder

Biomedical engineering division, Dept. of E.E., Twente University of Technology, P.O. Box 217, 7500 AE Enschede, The Netherlands.

Inner products of vectors consist of the sum of the products of corresponding elements of two arrays having equal length. They are the time-consuming part of many digital signal processing operations, such as convolution and correlation.

Because of their extensive use in signal processing inner products cause a heavy burden to normal microprocessors. To relieve the processor from a considerable number of inner products in real-time applications, a special purpose processor, called VIPER-II, was designed as a parallel processor. VIPER-II's program and data is stored in two random- access memory banks that are alternately accessible for the VIPER-II arithmetic unit or normal microprocessor. A control-status register and interrupt facilities are provided.

Besides real-time signal processing and control, VIPER-II is also suitable for the processing of large amounts of data.

1. INTRODUCTION

When microcomputers are involved in real-time signal processing or control functions, computing time is often a well-known limitation. Much effort has been put into the reduction of the number of time consuming operations such as multiplications [1]. This sometimes led to considerable concessions being made with regard to the performance of the algorithms concerned. Convolution and correlation operations are extensively used in digital signal processing for digital filtering, spectral analysis, pattern recognition etc [2]. Therefore we studied these operations in order to isolate their computer time-consuming part.

The convolution sum, used to calculate the output signal y(k) of a non-recursive digital filter with an impulse response of N elements, can be described by equation (1), where u(k) is the input signal and h(i) the impulse response.

$$y(k) = \sum_{i=1}^{N-1} h(i)\, u(k-i) \qquad (1)$$

The discrete Fourier transform $X(j\omega)$ of a sampled signal x(k) can be derived using equation (2):

$$X(j\omega) = \sum_{i=o}^{N-1} x(i) \cos(\omega i/N) - j \sum_{i=o}^{N-1} x(i) \sin(\omega i/N) \qquad (2)$$

The correlation coefficient r can be used to recognize patterns described by a template k(i) in an one-dimensional signal y(k) as shown in equation (3). It is expected that k and y will average zero during the observed interval.

$$r = \Big\{ \sum_{i=o}^{N-1} k(i).y(i) \Big\} \cdot \Big\{ \sum_{i=o}^{N-1} k(i)^2 \cdot \sum_{i=o}^{N-1} y(i)^2 \Big\}^{-1/2} \qquad (3)$$

The computer time-consuming part of the equations (1-3) consists of inner products of two vectors as described in equation (4).

$$\text{inner product } U.V = \sum_{i=o}^{N-1} u(i).v(i) \qquad (4)$$

where U = u(0), u(1), ..., u(N-1)
and V = v(0), v(1), ..., v(N-1)

One or both vectors U and V often represent part of a time-dependent signal. These signal epochs can be described as in (5), where T is the sample interval.

$$U(t) = u(t), u(t-T), \ldots, u(t-(N-1)T) \qquad (5)$$

In the case of a constant vector length N, the oldest element
u(t-(N- 1)T) is removed after every sample interval T and a new element u(t+T) is added, in order to follow the latest signal developments.

In such cases it is convenient to store vectors as presented in equation (5) in so-called circular buffers, where in a linear array a moving pointer indicates the logical begin of the vector. In this way the data manipulation can be restricted to one vector element every sample interval, i.e. exchanging the oldest element by the newest.

The inner product is computer time consuming mainly because of the relatively slow execution of the integer multiply instruction. This is for instance 25 microseconds for the LSI 11/23 processor. The data manipulations needed for the inner product calculations also take a considerably amount of time.

In order to overcome this bottle-neck for digital filtering special hardware is proposed in order to perform the convolution operation in case of a finite impulse response [3]. We developed a special purpose processor for the fast calculations of inner products. The processor is partly based on experience with a predecessor [4] having less elaborate features. The apparatus described here is called VIPER-II like its predecessor. VIPER stands for vector inner product equipment for real-time. It operates in parallel to a DEC LSI-processor and is connected to the Q-bus by its own build-in interface. Using VIPER-II, inner products are no longer a load for the processor so the processor stays available for other data and control manipulations.

The basic system configuration of VIPER-II is shown in figure 1.

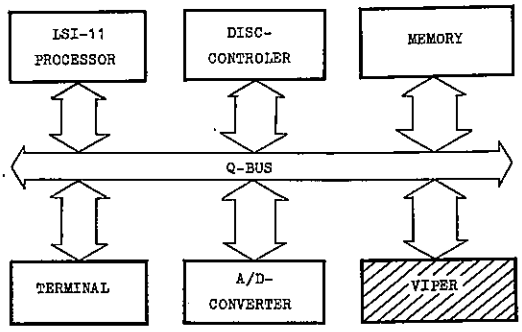


Figure 1. System configuration.

VIPER-II calculates a number of inner products of vectors consisting of 16-bit integer-valued arrays of which the length may vary from 64 to 512 elements. The vectors may be loaded or changed any time between calculations which makes VIPER-II suitable for the implementations of, for instance, adaptive filters or correlators. Its use of circular buffers minimizes the data exchange. VIPER-II is realized as a plug-in quad Q-bus module and is therefore equipped with a specially designed Q-bus interface. The data exchange is controlled by means of a control/ status register and interrupt facilities.

2. PRINCIPLE OF OPERATION

VIPER-II is realized as a special purpose processor that operates on the Q-bus parallel to the LSI-11 processor. It contains two separate random access memory (RAM) banks of 4096x16 bits which are switched antiparallel as shown in figure 2.

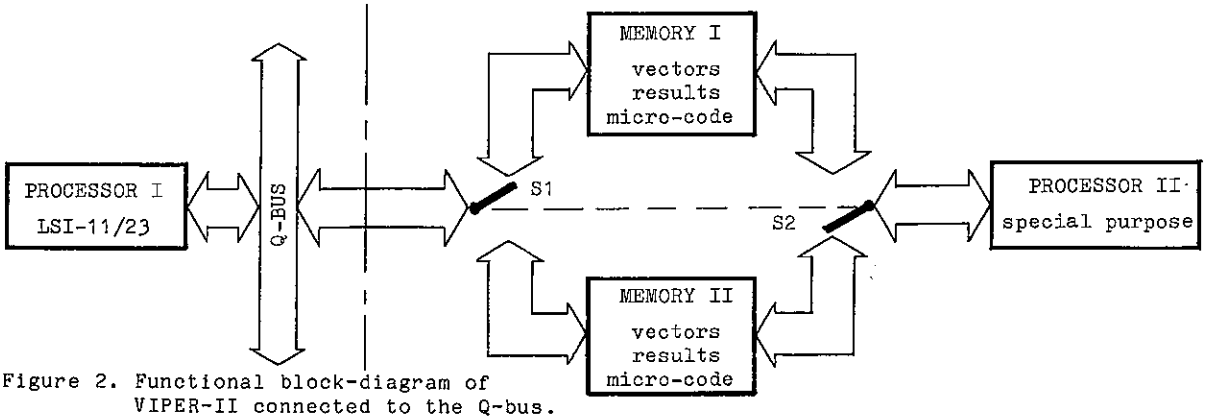


Figure 2. Functional block-diagram of VIPER-II connected to the Q-bus.

One of the RAM banks is connected to VIPER-II's processor and the other is accessible for the LSI-11 processor like normal computer RAM. After calculation of the inner products in one RAM bank, this RAM bank is switched to the Q-bus and the other bank to the VIPER-II processor.

Each RAM bank has to contain a coherent set of vectors and a control microcode to match. The inner product results are also stored in this bank. The inner product processor consists of a fast multiplier/accumulator integrated circuit with control and vector element address generation logic. The switches, consisting of the multiplexers S1 and S2, are controlled anti-parallel. These multiplexers are controlled by the LSI-11 processor by means of the control/status register. This register is also used to actually start the inner product calculations.

The elements of a vector have to be stored at successive addresses in the RAM bank. The microcode provides the highest addresses of the two vector memory fields involved in a specific inner product. Address generation logic constructs the actual vector element addresses needed from these data.

For example, the address of the ith element a(i) of a vector with length N is composed in the following way:

$$a(i) = v + (n+i) \bmod N$$

where

v, the vector starting address is the lowest address of the RAM space used for the storage of the vector concerned and is directly obtained from the microcode.

n, the newest element index indicates where the most recent element of the vector is stored, relative to v. This index is used when a vector is arranged as a circular buffer.

mod N, represents the mathematical modulo N operation.

3. CIRCUIT DESCRIPTION

The block diagram of figure 3 represents the total apparatus.

The multiplier/accumulator comprises a single large scale integrated circuit, the ADSP 1010 KD which is manufactured by Analog Devices [5] This multiplier/accumulator

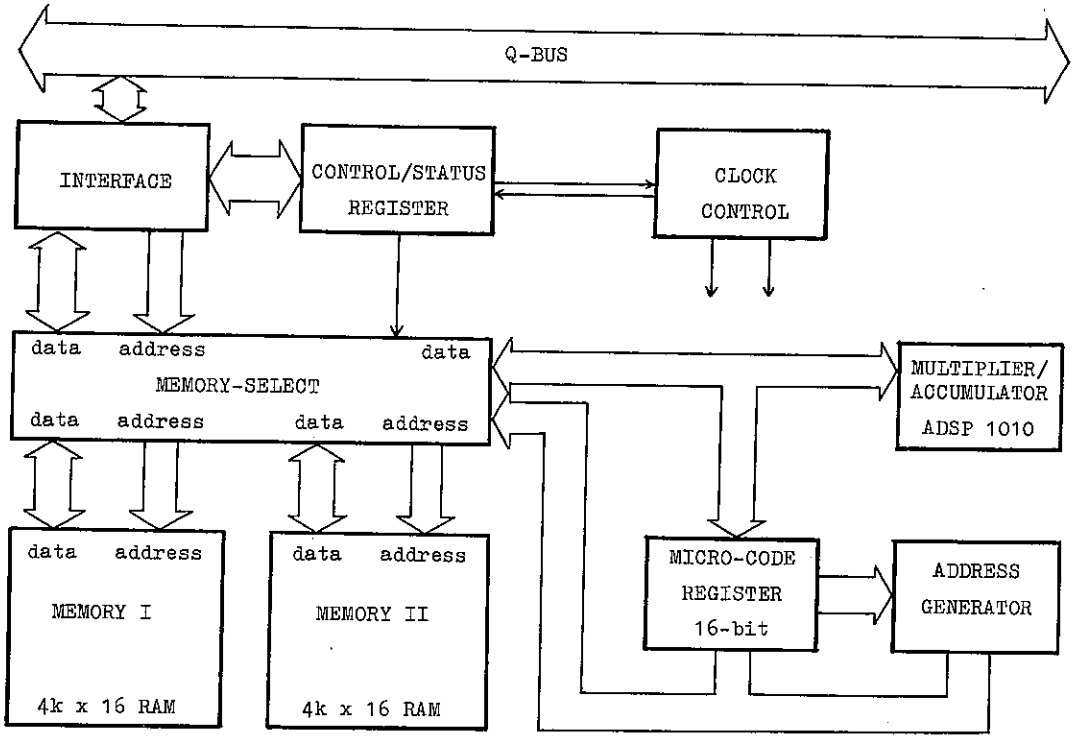


Figure 3. Simplified block-diagram of VIPER-II.

obtains its data from and stores its results in the 4096x16-bit RAM to which it is connected via a data bus by means of the memory select unit. The microcode necessary for the control of the calculations is obtained from the same RAM bank. The actual microcode word, describing an inner product is stored in the microcode register and used as input to the address generator.

A microcode word specifies the base address of the actual vectors, the vector length and whether each of the vectors is stored as a circular buffer or not.

Besides the addresses of the vector elements, the address generator also provides the addresses where the 32-bit product results are to be stored. A special stop code in the microcode program indicates when the last inner product has been calculated.

The control/status register specifies the communication between the Q-bus and VIPER-II. It contains, as shown in figure 4, a start bit initiating the calculations, a data ready bit, indicating that the products are all calculated, a memory select bit and an interrupt enable bit.

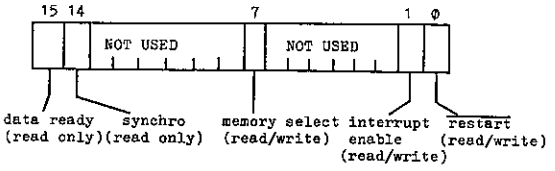


Figure 4. Control/status register.

The interface provides the signals necessary for the operations of VIPER-II as a Q-bus module. It controls the data exchange with both memory banks and the control/status register. The interface also contains the interrupt generation logic.

The clock control provides the signals for the synchronization and timing of the various hardware components.

4. CHARACTERISTICS

The characteristics of VIPER-II are summarized as follows. VIPER-II is a plug-in quad-slot Q-bus module. It comprises a special purpose parallel processor for inner vector products, which is micro-programmable. Data and program are stored in two alternatingly accessible RAM banks. The vectors may consist of 64, 128, 256 or 512 x 16-bit integer elements. The product results are represented as 32-bit integers. In one calculation run, a maximum of 61 products of vectors having a length of 64 elements to 7 products of vectors having a length of 512 elements, can be calculated. There may also be a free choice combination of other vector lengths, only restricted by memory space and a maximum of 63 products for each memory bank.

The communication between VIPER-II and the Q-bus is arranged as normal memory access to the selected memory bank. The bank selection and calculation control functions is performed by means of a control status register. Interrupts can be generated on completion of the actual calculations. The inner product calculation time is determined by the number of multiplications, which means by the vector length. One multiplication/accumulation cycle takes 500 ns.

5. APPLICATIONS

VIPER-II has worked satisfactorily as a parallel processor in a PDP 11/23 computer system for over a year. It has been found to be very useful in real-time signal processing and real-time control applications, especially when impulse responses or templates have to be adaptive. But its applicability is not restricted to these functions. Other possible applications are: digital filtering, recursive or non-recursive as for example linear phase filters, adaptive filters and matched filters; continuous frequency analysis of sampled signals, calculations of signal power; cross- and auto correlation functions; statistical manipulations; matrix operations; etcetera.

6. REFERENCES

[1] Alsté, J.A. van, Schilder T.S., Removal of baseline wander and powerline interference from the ECG by an efficient FIR filter with a reduced number of taps. IEEE Trans. on Biomed. Engin. Vol. BME 32, 1052-1061, 1985.

[2] Jones N.B., Digital signal processing, IEE control engineering series 22, Peter Peregrinus Ltd. UK, 1982.

[3] Kolb H.J., Digitales FIR-filter für die Messwertverarbeitung. Elektronik 3, 85-88, 1983.

[4] Alsté J.A. van, Luursema E.D., VIPER: a powerful tool for the real-time calculation of inner products for biomedical signal processing. Med. & Biol. Eng. and Comput., 23, 74-76, 1985.

[5] Analog Devices. ADSP 1010, 16x16 bit CMOS Multiplier/Accumulator. Datasheet from Analog Devices, Norwood, Massachusets, U.S.A.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REALIZATION OF HYBRID FINITE IMPULSE RESPONSE FILTERS USING SEMICONDUCTOR LIGHT-EMITTING DIODES AND PHOTODIODES

Peter Laws

Physikalisches Institut
Abteilung Angewandte Optik
Universität Erlangen-Nürnberg
Erwin-Rommel-Str. 1
D-8520 Erlangen, Federal Republic of Germany

A method of realizing nonrecursive filter structures is proposed. This method is based on a fast electro-optical multiplier which executes multiplications by means of light-adding and light-weighting surfaces of semiconductor light-emitting diodes and photodiodes.

1. INTRODUCTION

On-line filtering of broadband signals strongly depends on the speed of multiplication and accumulation procedures. Using digital 1D Finite Impulse Response (FIR) filter structures as a basis, a study has been made to investigate to what extent semiconductor light-emitting diodes (LED) and photodiodes can replace the hardware-multipliers/accumulators of such FIR filter structures. One of the results of this study is a proposal for an electro-optical multiplier, which functions on the basis of special LED- and photodiode arrays.

2. THE ELECTRO-OPTICAL MAGNITUDE MULTIPLIER

On-line filtering of a quantized 1D signal s_q (bandwidth f_c) by means of a digital 1D FIR filter specified by N quantized filtercoefficients $h_{q\mu}$ ($\mu=0,1,\ldots,N-1$) and operating at a sampling frequency

$$f_a = 1/T_a \geqslant 2f_c \tag{1}$$

means that N products $h_q s_q$ per period T_a must be executed. That is, if the number N of filtercoefficients and the cut-off frequency f_c are high, the time T_m for a single multiplication/accumulation must be relatively low.

The following description explains a hybrid (electro-optical) multiplier which is fast because of its parallel processing structure.

We assume that the signal portion s_q, as well as the filter coefficient h_q, has been binary coded by means of the sign-magnitude code and both are available as an electronic parallel M-bit word (M = wordlength):

$$s_b = s_{M-1}\ s_{M-2}\ \cdots\ s_m\ \cdots\ s_1\ s_0 \tag{2}$$

$$h_b = h_{M-1}\ h_{M-2}\ \cdots\ h_n\ \cdots\ h_1\ h_0 \tag{3}$$

Then the most significant bits s_{M-1} and h_{M-1} represent the sign of s_q and h_q respectively, corresponding to

$$s_{M-1} = \begin{cases} 1 \text{ if } s_q > 0 \\ 0 \text{ if } s_q \leqslant 0 \end{cases} \tag{4}$$

$$h_{M-1} = \begin{cases} 1 \text{ if } s_q > 0 \\ 0 \text{ if } h_q \leqslant 0 \end{cases} \tag{5}$$

whereas the correspondence between magnitude and remaining bits is given by

$$|s_q| = \sum_{m=0}^{M-2} s_m\, 2^m \tag{6}$$

$$|h_q| = \sum_{n=0}^{M-2} h_n\, 2^n \tag{7}$$

From Eq. (6) and Eq. (7) we see that the product-magnitude of $h_q s_q$ can be described by

$$|h_q s_q| = \left(\sum_{n=0}^{M-2} h_n 2^n\right)\left(\sum_{m=0}^{M-2} s_m 2^m\right) \tag{8}$$

Multiplying formally on both sides of Eq. (8) by the constants A_o and M_e yields

$$P = M_e A_o |h_q s_q| = \sum_{n=0}^{M-2} \sum_{m=0}^{M-2} (h_n s_m M_e 2^{n+m} A_o) \tag{9}$$

Interpreting A_o as the "smallest area" (unit: m^2) of a semiconductor light-emitting diode (LED) radiating at a "specific radiation" M_e (unit: W/m^2) the quantity P in Eq. (9) corresponds to the total amount of radiation power radiated by all partial areas

$$A_{nm} = 2^{n+m} A_o \tag{10}$$

of single separated LEDs, where the single LED partial area A_{nm} radiates the partial radiation power

$$P_{nm} = \begin{cases} M_e 2^{n+m} A_o & \text{if } h_n s_m = 1 \\ 0 & \text{if } h_n s_m = 0 \end{cases} \qquad (11)$$

Following this interpretation we can design an array of LEDs and electronic AND-gates, which realizes Eq. (9) and whose optical output P is proportional to the product-magnitude $|h_q s_q|$.

A corresponding magnitude multiplier is shown for h_b = 1101 and s_b = 0110 in Fig. 1.

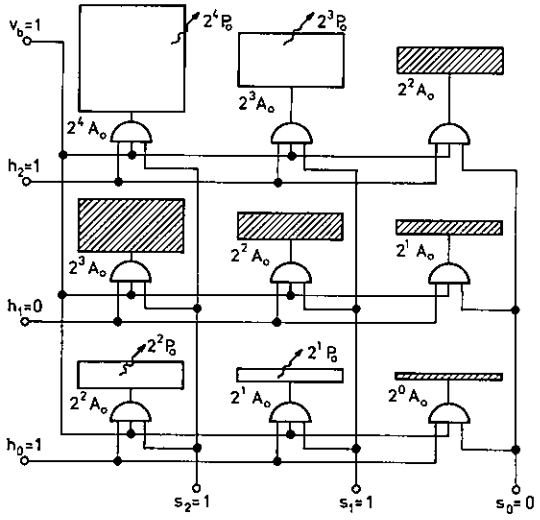


Fig. 1 Magnitude multiplier
shadowed areas: inactive LEDs,
$P_o = P_{oo} = M_e A_o$

As it can be seen from Fig. 1 every single LED partial area A_{nm} is controlled by an associated electronic logic AND-gate, which switches the LED on or off, if $h_n s_m$ equals 1 or 0, respectively.

Assuming that the two electronic logic signals h_{bB} and s_{bB} associated with the magnitudes $|h_q|$ and $|s_q|$ are input simultaneously, the product magnitude quantity P is output as fast as the gate-LED combinations can switch on, that is independent from the wordlength M.

Fig. 1 also illustrates that the radiation power P is only available at the output of the magnitude multiplier (its symbol is depicted in Fig. 2) if there is a logic 1 at the v_b-input.
A logic 0 for v_b disables the complete magnitude multiplier.

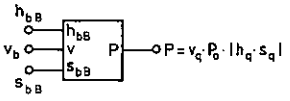


Fig. 2 Symbol of the magnitude multiplier

The sign-controlling input v_b is used for the construction of the product-magnitude switch PBS (Produkt Betrag Schalter), which is shown in Fig. 3.

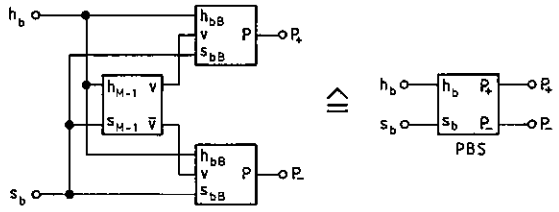


Fig. 3 Product magnitude switch PBS

The PBS consists of an electronic logic sign controller and two magnitude multipliers. According to

$$\overline{v}_b = h_{M-1} \oplus s_{M-1} \qquad (12)$$

the sign controller converts the sign bits h_{M-1} and s_{M-1} into an EXCLUSIVE OR signal $\overline{v}_b$ and a corresponding equivalence signal v_b. Thus, depending on the resulting product sign the upper magnitude multiplier radiates

$$P_+ = (1-\overline{v}_b) M_e A_o |h_q s_q| \qquad (13)$$

or the lower magnitude multiplier outputs

$$P_- = \overline{v}_b M_e A_o |h_q s_q| \qquad (14)$$

3. THE ELECTRO-OPTICAL MULTIPLIER

The fast conversion of the electronic M-parallel-bit signals h_b and s_b (to be multiplied) into a product-proportional voltage signal u_a is performed by means of the hybrid (electro-optical, logic-analogue) circuit shown in Fig. 4.
This circuit contains one product-magnitude switch PBS, two semiconductor photodiodes D1 and D2, which collect and convert the radiation power P_+ and P_-, and an operational amplifier OP, which, together with the feedback resistor R_F, acts as a transimpedance amplifier.

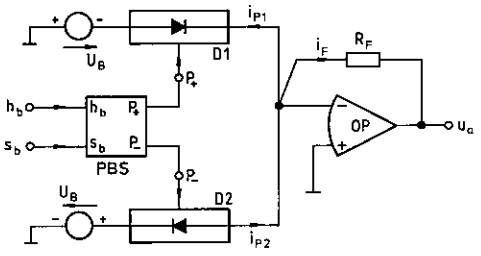


Fig. 4 Electro-optical multiplier

Assuming that the reverse-biased semiconductor photodiodes D1 and D2, covering the corresponding PBS (see Fig. 5), convert the radiation power P_+ and P_- into photocurrents, which are proportional to the product-magnitudes

$$i_{P1} = SP_+ \tag{15}$$

$$i_{P2} = SP_- \tag{16}$$

S = photodiode sensitivity, unit: A/W

and considering Eqs. (13), (14), (15) and (16) we obtain the sum current

$$i_F = i_{P2} - i_{P1} = SM_eA_o(2\overline{v}_b - 1)|h_q s_q| \tag{17}$$

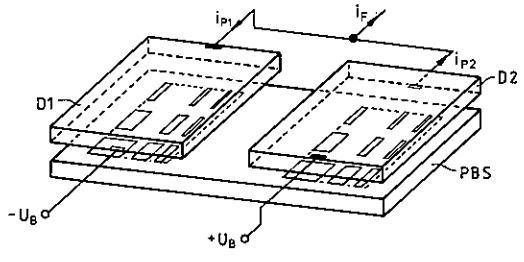


Fig. 5 Photodiodes coupled to the PBS

Since the output voltage of the transimpedance amplifier can be described by

$$u_{as} = -R_F i_F \tag{18}$$

and because

$$\text{sign}(h_q s_q) = 1 - 2\overline{v}_b \tag{19}$$

we conclude from Eqs. (17), (18) and (19) that

$$u_{as} = R_F SM_e A_o h_q s_q \tag{20}$$

This result shows that the ouput signal u_{as} is product-proportional.

4. THE HYBRID FIR FILTER STRUCTURE

The hybrid FIR Filter structure is depicted in Fig. 6.

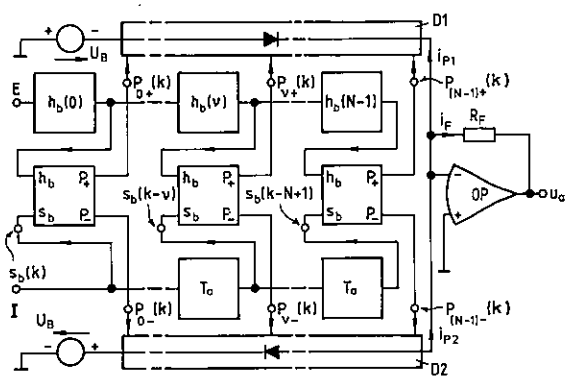


Fig. 6 Hybrid FIR filter

The structure contains N electronic logic latches (storing the N filter coefficients), N product-magnitude switches PBSs (whose P_+- and P_--ports are coupled to the photodiodes D1 and D2, respectively) and (N-1) electronic logic shift registers (specified by the shift period T_a).
The electronic logic input signal $s_b(k)$ to be filtered is entered on port I.at clock rate T_a, shifts through the shift registers and appears after convolution at the ouput port as

$$u_{as}(t) = R_F SM_e A_o \sum_{k=0}^{\infty} y_q(k)\,\text{rect}\left(\frac{t}{T_a} - \frac{1}{2} - k\right) \tag{21}$$

where

$$y_q(k) = \sum_{\mu=0}^{N-1} h_q(\mu)s_q(k-\mu) \tag{22}$$

describes the discrete convolution process. As can be seen from Fig. 6 the photodiodes execute simultaneously N products and N accumulations. Thus the speed of filtering is limited by the switching time of the product-magnitude switches rather than by the number of filter coefficients.

Eq. (21) describes the ideal case, i.e., the noiseless case. In reality the transimpedance amplifier outputs

$$u_a(t) = u_{as}(t) + u_{an}(t) \tag{23}$$

where $u_{an}(t)$ is the sum of noise portions generated by photon-carrier conversion, photodiode dark currents, resistance R_F and by internal noise sources of the operational amplifier.

One possibility for specifying the noise properties of the above hybrid FIR filter is to define the signal/noise ratio

$$S_a/N_a = u^2_{asmin}/u^2_{anmax} \qquad (24)$$

where

$$u_{asmin} = R_F S M_e A_o \qquad (25)$$

is the smallest amplitude step of the quantized output $u_{as}(t)$. However, u_{anmax} depends upon various filter-, LED-, photodiode- and amplifier parameters and can be calculated for given filter coefficients if the input signal $s_b(k)$ is assumed to show always its largest value (for detail see [1]). That is, if

$$s_b(k) = \begin{cases} 0\ 0\ 0 \cdots 0\ 0\ 0 & \text{if } k < 0 \\ 1\ 1\ 1 \cdots 1\ 1\ 1 & \text{if } k \geq 0 \end{cases} \qquad (26)$$

5. COMPUTER SIMULATION AND RESULTS

The physical dimension of a FIR filter described above may be very small, if the filter is realized as a monolithic or hybrid integrated circuit.
This can be derived from results obtained by computer-aided filter design and simulation considering electrical and/or optical characteristics of commercially available LEDs, photodiodes and broadband operational amplifiers.

Requiring, for example, a discrete short-time averaging filter specified by

- sampling rate f_a = 10 MHz
- number of filter coefficients N = 20
- wordlength M = 6
- filter coefficients $h_q(\mu) = 2^{M-1}-1$ for $\mu = 1,2,\ldots(N-1)$
- S_a/N_a = 10

and considering the characteristics of

- an IR-LED AEG V234P
- Si-PIN-photodiode RCA C30808
- operational amplifier BB OPA605

leads to

- smallest LED Area : $A_o = 65.7\ (\mu m)^2$
- smallest area of covering photodiode: $A_D = 1.26\ (mm)^2$
- substrate area (including all product-magnitude switches, shift registers and operational amplifier): $A_S = (4.77\ mm)^2$
- smallest amplitude step: $u_{asmin} = 326\ \mu V$

REFERENCE

[1] Laws, P., Nichtrekursive Filter mit lichtaddierenden und lichtwichtenden Oberflächen, Habilitationsschrift, Universität Duisburg, 1983

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ARRAY PROCESSOR FOR 2-D DISCRETE COSINE TRANSFORMS

M. Afghahi, S. Matsumura*, J. Pencz, B. Sikström
U. Sjöström, L. Wanhammar

Linköping University
Department of Electrical Engineering
S-581 83 Linköping, Sweden

* Kanazawa Institute of Technology
Kanazawa, Japan

ABSTRACT

In this paper we discuss the design and architecture of an array processor for the 2-dimensional discrete cosine transform, DCT. The necessary matrix transposition in the calculation of a 2-dimensional transform by first computing the transform of the rows and then the transform of the intermediate columns, is included in the processor. The implementation technique is also applicable to other discrete transforms. The processor is aimed for use in high speed applications, e.g., transform coding of TV images in real time. The processing elements are based on distributed arithmetic.

1. INTRODUCTION

Various types of 2-dimensional discrete cosine transforms (DCT's) are important tools in the field of digital signal processing, e.g. speech and image coding. In this paper an architecture that facilities VLSI implementations of different kinds of 2-dimensional discrete cosine transforms [2, 3] is presented. The application for the 2-D DCT is in transform coding of TV images in real time. Transforms of 8x8 and 16x16 points are of special interest. An array processor for 2-D discrete cosine transforms of 8x8 points is considered, but other types of 2-D transforms can be implemented as well using the proposed method.

The processing elements are based on distributed arithmetic [1]. Special techniques are exploited to reduce the size of the ROM's, which makes it feasible to implement a 16x16 point 2-D DCT processor onto one chip using the same scheme. Hitherto, a test chip containing only the processing elements for an 8 point symmetric DCT have been implemented [2, 3].

In the architecture, advantage is taken of the fact that all processing elements operate on the same set of data, which simplifies the interconnection network.

2. 2-D DISCRETE COSINE TRANSFORMS

Several discrete cosine transforms has been presented in the literature, i.e., odd, even and symmetric versions [2, 3]. The proposed implementation is intended for use in systems for transform coding of images. In such applications it is required that a large DC-component in an image do not contribute with an error in higher frequency components. Therefore, only the even and the symmetric DCT's are to be considered further. Furthermore, the proposed implementation method exploit symmetries in the base-functions of the DCT's. The new version of the symmetric DCT (SDCT-2) presented in [2] is therefore used. Moreover, the forward and the inverse transform are identical for the SDCT-2. This is of great importance since only one chip has to be designed. One minor drawback is that the transform is only "near orthogonal".

Other 2-D discrete transforms can also be implemented using the presented technique.

The 2-dimensional SDCT-2 transform is defined according to:

$$y(p,q)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} c_i c_j\, x(i,j)\cos\left(\frac{ip\pi}{N-1}\right)\cos\left(\frac{jq\pi}{N-1}\right)$$

$$p, q = 0, \ldots, N-1 \quad (1)$$

where c_k are the weighting coefficients

$$c_k = \begin{cases} \frac{1}{2} & , \; k=0 \text{ and } k=N-1 \\ 1 & , \; \text{otherwise} \end{cases}$$

$x(i,j)$ is the input array of data (NxN):

$$\begin{bmatrix} x(0,0) & x(0,1) & \cdots & x(0,N-1) \\ x(1,0) & \cdot & & \cdot \\ \cdot & & \cdot & \cdot \\ \cdot & & & \cdot \\ x(N-1,0) & & \cdots & x(N-1,N-1) \end{bmatrix}$$

and $y(p,q)$ is the output array of data (NxN):

$$\begin{bmatrix} y(0,0) & y(0,1) & \cdots & y(0,N-1) \\ y(1,0) & \cdot & & \cdot \\ \cdot & & \cdot & \cdot \\ \cdot & & & \cdot \\ y(N-1,0) & & \cdots & y(N-1,N-1) \end{bmatrix}$$

The 2-dimensional cosine transform can be separated into 1-dimensional transforms by first computing the intermediate data $w(i,q)$, according to Eq. (2), and then compute the output data $y(p,q)$, according to Eq. (3), respectively.

$$w(i,q)=\sum_{j=0}^{N-1} c_j\, x(i,j)\cos\left(\frac{jq\pi}{N-1}\right)$$

$$i, q = 0, \ldots, N-1 \quad (2)$$

and

$$y(p,q)=\sum_{i=0}^{N-1} c_i\, w(i,q)\cos\left(\frac{ip\pi}{N-1}\right)$$

$$p, q = 0, \ldots, N-1 \quad (3)$$

For each value of i, Eq. (2) represent an N point 1-D transform, i.e., from row i of the matrix of input data, the i:th row of the matrix of intermediate data are computed.

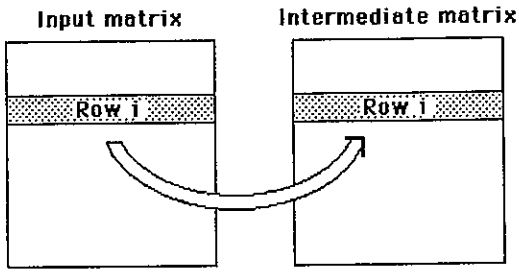


Fig. 1. Computation of one row in the matrix of intermediate data.

In the same way Eq (3) represent an N point 1-D transform of the q:th column of the matrix of intermediate data to the q:th column of the matrix of output data.

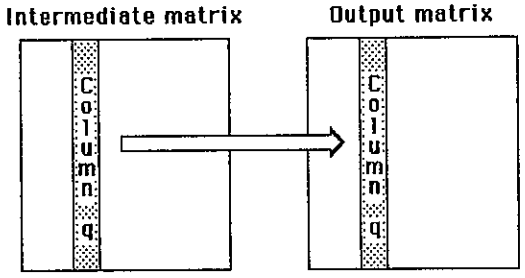


Fig. 2. Computation of one column in the matrix of output data.

Thus, a 2-dimensional NxN point cosine transform can be computed using 2N 1-D transforms, each of N points.

3. THE PROCESSING ELEMENTS

The processing elements (PE's) are based on distributed arithmetic [1], which is an efficient method for computing inner products. The main parts in one PE are a shift-accumulator and a ROM containing all possible sums of the coefficients in the inner product.

Consider the inner product

$$W=\sum_{i=0}^{N-1} a_i x_i \quad (4)$$

The coefficients a_i corresponds to the weighted cosine factors, and x_i are the input values. The values are represented in two's complement with a word length W_d.

$$x_i = -u_{i0} + \sum_{r=1}^{W_d-1} u_{ir}\, 2^{-r} \qquad (5)$$

where u_{ir} is the r-th bit in the variable x_i. Thus, Eq.(4) can be rewritten using Eq.(5):

$$w = -F_0(u_{00},\ldots,u_{N-1,0}) + \sum_{r=1}^{W_d-1} 2^{-r} F_r(u_{0r},\ldots,u_{N-1,r})$$

$$F_r(u_{0r},\ldots,u_{N-1,r}) = \sum_{i=0}^{N-1} a_i u_{ir} \qquad (6)$$

Eq.(6) is a partial sum of coefficients depending on the corresponding data bits. The 2^N possible values of $F_r(u_{0r}, \ldots, u_{N-1,r})$ are stored in the ROM, which is addressed by the binary variables $(u_{0r}, \ldots, u_{N-1,r})$. Thus, the inner product is obtained by successive accumulation of the appropriate partial sums of the coefficients. Therefore, only one shift-accumulator and a ROM look-up table, as shown in Fig. 3, are required to compute one inner product.

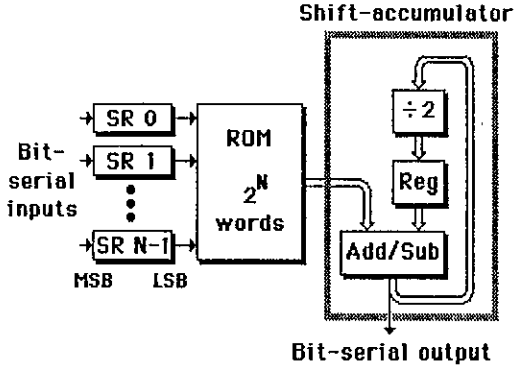


Fig. 3. Distributed arithmetic.

For N = 8, the ROM contain 256 words, which is not negligible from an chip area point of view. The number of terms in the inner products, and the number of words in the ROM, can be reduced by utilizing the symmetry in the base-functions of the transform. The input data in the inner products are added (subtracted) pairwise according to Eq.(7). Hence, the number of remaining terms is halved, and the number of words in the ROM are reduced to only 16.

$$\begin{aligned} w &= a_0x_0 + a_1x_1 + a_2x_2 + a_3x_3 \pm \\ &\pm a_3x_4 \pm a_2x_5 \pm a_1x_6 \pm a_0x_7 \\ &= a_0\,(x_0 \pm x_7) + a_1\,(x_1 \pm x_6) + \\ &+ a_2\,(x_2 \pm x_5) + a_3\,(x_3 \pm x_4) \end{aligned} \qquad (7)$$

4. IMPLEMENTATION OF A 1-D 8 POINT DCT

By using several PE's, a high speed 1-D DCT circuit can be implemented. An 8 point DCT requires 8 PE's, 4 adders and 4 subtractors, as shown in Fig. 4.

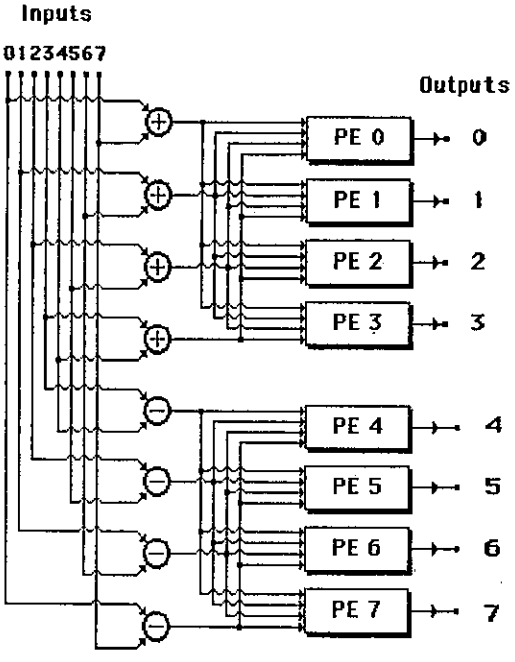


Fig. 4. Block diagram for an 8 point 1-D DCT.

5. IMPLEMENTATION OF A 2-D 8x8 POINT DCT

The 2-dimensional 8x8 point DCT is partitioned into 16 identical 1-dimensional 8 point DCT's. The circuitry for implementing the 8 point DCT is used to compute the 1-D transforms. In addition, a RAM for storage of intermediate data together with parallel/serial and serial/parallel converting SR's are needed. A block diagram for the 2-D DCT is shown in Fig. 5.

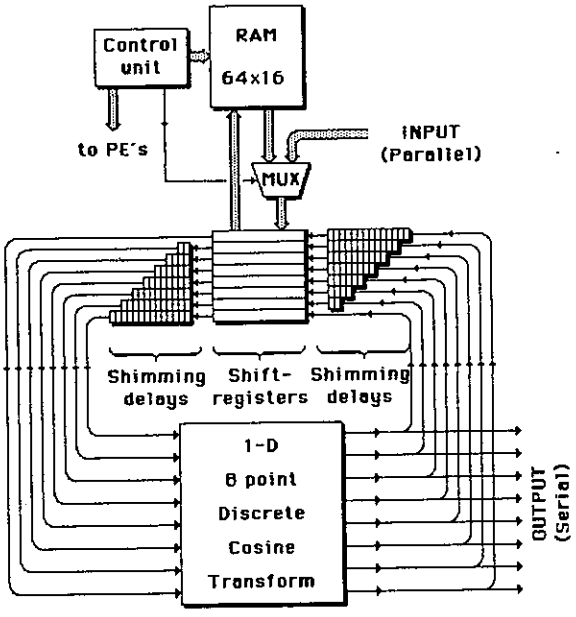


Fig. 5. Block diagram for an 8x8 point 2-D DCT.

In 16 consecutive clock cycles data, with the word length W_d=16 bits, are written in parallel into the RAM from the shift-registers (SR) and subsequently loaded in parallel from the RAM into the SR. The data loaded to the SR are either input data (**x**), which are assumed to be available in the proper time instances at the input port, or intermediate data (**w**) stored in the RAM. The left set of SR's, so-called shimming delays, with increasing lengths ensures that the bit-serial input data arrives to the PE's word-synchronized. This loading procedure takes 16 clock cycles. During the next 16 clock cycles, the computations of the inner products takes place, and new data for the next 1-D transform can be loaded into the SR's. Finally, after the next 16 clock cycles, the output data from the PE's are loaded into the SR's and subsequently written into the RAM. This completes the activities associated to one 1-D transform. Thus, the computation of one 1-D transform takes $3W_d$ clock cycles. However, a new set of computations for one 1-D transform can start every W_d clock cycle, due to the pipelining in three stages.

It is obvious that the proper data must be available in the RAM or of the input port when a new set of computations start. This is the case in all time instances except for the first set of computations of output data, **y(p,0)** in Eq.(3), depending on the fact that intermediate data from the previous set of computations is still into the two last pipeline stages. Thus, $2W_d$ extra clock cycles must be added. Therefore, $18W_d$ clock cycles are needed for the computation of one 2-D transform.

In order to avoid overflow in the adders and subtractors, a guard bit is used in the data words. The effective data word length is therefore 15 bits.

6. CONCLUSION

It has been shown that high performance 2-D cosine transforms can be efficiently implemented using a multiprocessor architecture based on distributed arithmetic. Using the same scheme, 16x16 point transforms can be implemented with the utilization of 16 PE's. However, in that case, several RAM's must be used in order to obtain balance between the communication channels and the PE's, i.e., to continuously support the PE's with input data.

The most important aspect of this approach, besides the high performance in speed, is the the regular and modular circuitry, which simplifies the design effort.

Moreover, this also decreases the probability of design errors.

ACKNOWLEDGMENT

This work has been supported by STU, the Swedish National Board for Technical Development.

REFERENCES

[1] Kronander T., Matsumura S., Sikström B., Sjöström U., Wanhammar L.: VLSI Implementation of the Discrete Cosine Transform, Proc. Nordic Symp. in Computers and Communications, Tampere, Finland, June 13-16, 1984.

[2] Matsumura S., Sikström B., Sjöström U., Wanhammar L.: LSI Implementation of an 8 Point Discrete Cosine Transform, Proc. Intern. Conf. on Computers, Systems and Signal Processing, Bangalore, India, Dec. 10-12, 1985.

[3] Matsumura S.: Discrete Cosine Transforms – Theory and LSI Implementation, Thesis No. 43, LiU-Tek Lic 1985:08, Linköping University, Sweden, May 7, 1985.

[4] Dinha F., Sikström B., Sjöström U., Wanhammar L.: A Multi-Processor Approach to Implement Digital Filters, Proc. Nordic Symp. in Computers and Communications, Tampere, Finland, June 13-16, 1984.

[5] Dinha F., Sikström B., Sjöström U., Wanhammar L.: LSI Implementation of Digital Filters – A Multi-Processor Approach, Proc. Intern. Conf. on Com- puters, Systems and Signal Processing, Bangalore, India, Dec. 10-12, 1985.

[6] Wanhammar L.: On Algorithms and Architecture Suitable for Digital Signal Processing, EUSIPCO-86, The Hague, The Netherlands, Sept. 2-5, 1986.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SOLVING SETS OF LINEAR EQUATIONS FOR REAL TIME SIGNAL PROCESSING

Kishan Jainandunsing and Ed F.A. Deprettere

Delft University of Technology
Department of Electrical Engineering
Mekelweg 4
2628 CD Delft, The Netherlands

ABSTRACT

In this paper we present a novel approach towards the problem of solving sets of linear equations, Ax=b, as they appear in many digital signal processing problems. This approach avoids a back substitution step or an orthogonal transformation, after the factorization step, as is the case for the conventional direct methods of QR or LQ factorization. In fact, the novel algorithm enables one to calculate the solution x forwardly from the factorization of the matrix A, using orthogonal or J-orthogonal transformations. It can be combined with the (generalized) Schur algorithm, which does the Cholesky factorization of the matrix A efficiently in case A is positive definite, symmetric and Toeplitz or close to Toeplitz. In case A is a general (non singular) matrix, the complete equations solver appears to be an orthogonal equivalent of Faddeeva's.

1. INTRODUCTION

A well known numerical procedure to solve systems of linear equations of the form :

$$Ax = b, \quad A = [a_{ij}]_{i,j=1,\ldots,n}. \tag{1}$$

is the direct method of decomposing the nxn matrix A into an orthogonal and an upper- or lowertriangular factor. The solution vector x is then recovered by a back substitution procedure or an orthogonal transformation. In short :

Uppertriangularization :

$$A = QR \rightarrow Rx = Q^t b = \underline{b}. \tag{2.a}$$
$$x = R^{-1}\underline{b}. \tag{2.b}$$

Lowertriangularization :

$$A = LQ \rightarrow Qx = \underline{x} = L^{-1}b. \tag{3.a}$$
$$x = Q^t\underline{x}. \tag{3.b}$$

where Q is an nxn orthogonal matrix (i.e., $QQ^t = Q^tQ = I$), R is an nxn uppertriangular matrix, L is an nxn lowertriangular matrix and ^{t} denotes transposition.

When addressing the problem of mapping the algorithms in equations (2) and (3) on an array of processor elements (PE's), one is faced with a storage problem. In the uppertriangularization algorithm one has to store the uppertriangular factor prior to the back substitution. This means $O(\frac{1}{2}n^2)$ storage. In the lowertriangularization case one has to provide storage for the matrix Q, which is again $O(\frac{1}{2}n^2)$. If n is very large, this means that virtual memory (disk or tape units) will be required in combination with high bandwidth busses in order to off load and down load the elements of R or Q.

The novel algorithm, which is presented in section 2, does not suffer from this storage problem. All operations on data can be done locally in the PE's of the array. Moreover, all operations are of the same type; either Givens rotations or Householder reflections. There are no excessive storage requirements for a PE. All storage requirements can be kept independent of the size of the matrix A. One drawback is however, that the throughput, measured as the rate at which problems Ax=b are solved, is half of the conventional algorithms in (2) and (3). But, with additional hardware, this drawback can be eliminated.

In (2.b) and (3.a) there are divisions involved with the diagonal entries of the uppertriangular matrix R and the lowertriangular matrix L. In case the systems of equations (1) is nearly singular, some of these diagonal entries may become (almost) zero within the finite machine precision and the divisions may cause severe numerical problems. The novel algorithm is robust in regard of this aspect.

It is also possible to exploit special structure in the coefficient matrix A and to have a solver whose efficiency is tuned to the special structure. In signal processing the matrix A is often of the Toeplitz or close to Toeplitz type [1,2]. In these cases one can combine the novel algorithm with the (generalized) Schur algorithm [3,4,5], which exploits the Toeplitz or close to Toeplitz structure of the matrix A and obtain an overall optimized solver, which exhibits the numerical robustness of both algorithms. This is the topic of section 3.

2. THE NOVEL EQUATIONS SOLVER

In this section we solve the storage problem with a forward procedure, however, be it at the cost of a decreased throughput (for a minimal hardware reali-

zation). Looking back at equation (1), we can rewrite it as follows :

$$[x^t \;\; 1] \begin{bmatrix} A^t \\ -b^t \end{bmatrix} = 0. \tag{4}$$

Equation (4) can be embedded in a bigger, yet simpler, problem. This simpler problem is the QR factorization of the (n+1)xn matrix $[A \;\; -b]^t$. The embedding is as follows.

Let $\pm(1 + x^t x)^{-\frac{1}{2}}[x^t \;\; 1]$ be the last row of an orthogonal matrix U, where the remainder of U is chosen such that

$$U \begin{bmatrix} A^t \\ -b^t \end{bmatrix} = \begin{bmatrix} R \\ 0_n^t \end{bmatrix}. \tag{5}$$

I.e., U is the orthogonal factor in the QR factorization of the augmented matrix $[A \;\; -b]^t$. Writing U as

$$U = \begin{bmatrix} U_{11} & u_{12} \\ u_{21}^t & u_{22} \end{bmatrix}, \tag{6}$$

we derive the following relationship from (5) and (6) :

$$u_{21}^t = u_{22} x^t. \tag{7}$$

And from the fact that

$$\pm(1 + x^t x)^{-\frac{1}{2}}[x^t \;\; 1] = [u_{21}^t \;\; u_{22}],$$

we conclude that

$$u_{22} = \pm(1 + x^t x)^{-\frac{1}{2}}, \tag{8}$$

which is the common scaling factor for the elements of the solution vector x.

Conclusion : it is sufficient to compute the orthogonal factor of the augmented matrix $[A \;\; -b]^t$ in order to find the required solution of the equation $Ax = b$.

The complete algorithm is summarized in equation (9) :

$$U\underline{A} = U \begin{bmatrix} A^t & I_n & 0_n \\ -b^t & 0_n^t & 1 \end{bmatrix} = \begin{bmatrix} R & U_{11} & u_{12} \\ 0_n^t & u_{22}x^t & u_{22} \end{bmatrix}. \tag{9}$$

The (n+1)x(n+1) identity in the (n+1)x(2n+1) matrix $\underline{A}$, extracts the last row of U. Notice that this row could also be obtained by premultiplying U with the row vector $[0_n^t \;\; 1]$. However, this is equivalent to a backwards procedure and would require storage for the matrix U. Hence, in the scheme in (9) there is twice as much data involved in the processing, than was strictly needed. In (9) the scaling factor u_{22} is produced last. Substituting for the identity the matrix with 1's on the i(n-i)th positions and zero's elsewhere, would reverse this situation, creating a completely forward procedure.

Despite the larger data set that needs to be processed in (9), the novel algorithm has some important implementation advantages. Its data flow pattern is completely regular and all elementary operations are the same (except for the unscaling), namely Givens rotations or Householder reflections. Hence, it is a good candidate for a VLSI implementation on a simple array of PE's. In fact, equation (9) is an ordinary QR factorization of the (n+1)x(2n+1) matrix $\underline{A}$. For this matrix factorization problem there are various systolic implementations described in the literature [6, 7]. One of these, due to Gentleman and Kung [6], is the triangular array shown in figure 1.

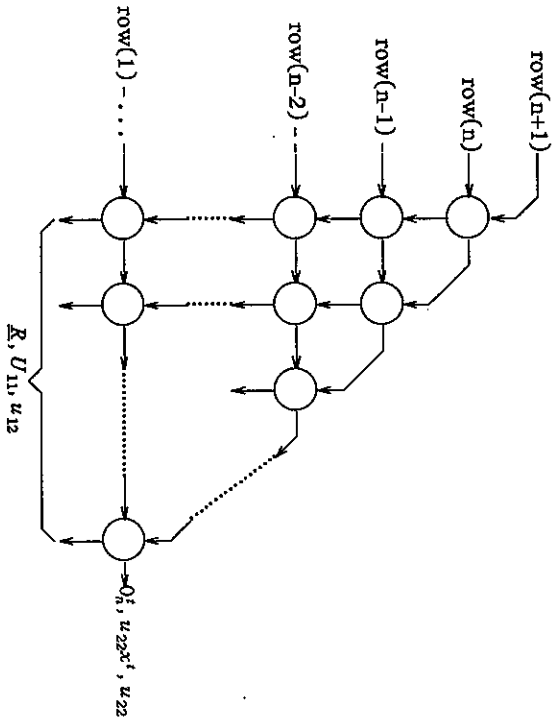


Figure 1. Triangular systolic QR factorization array.

This array is an array of Givens rotors

$$G(\theta) = \begin{bmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{bmatrix}.$$

At the left of the array the (n+1) rows of the matrix $\underline{A}$ are inputted. Each column of rotors computes once an orthogonal matrix such that the uppertriangular form $[R^t \;\; 0_n]^t$ is computed from the augmented matrix $[A \;\; -b]^t$. An angle θ is stored locally at a PE once it has been computed. This storage has not been shown in figure 1. The throughput of the algorithm in equation (9) is half of that of the conventional ones in (2) and (3). This is because the (n+1)x(n+1) identity has to be processed additionally. A solution to this problem is to duplicate the array in figure 1 and have the identity processed by the duplicated one. The PE's in this array receive the θ-parameter from the PE's which compute the uppertriangularization. In case the PE's are implemented as pipelined CORDICs [8], such a hardware configuration of twin arrays is a very elegant and highly regular machine that may achieve high throughputs. There is no complicated data flow, controller and excessive storage needed as is the case for systolic implementations of the conventional algorithms in (2) and (3).

There is an interesting connection between the algorithm in equation (9) and Faddeeva's algorithm for solving systems of linear equations [9]. In fact, our algorithm is an orthogonal equivalent of Faddeeva's, which uses Gaussian elimination. If we would omit the last column of the augmented matrix $\underline{A}$ and transpose the remainder, we get exactly the matrix in Faddeeva's case :

$$\begin{bmatrix} A & -b \\ I_n & 0_n \end{bmatrix}.$$

The difference with Faddeeva's is that we get a scaled solution. Because of the similarity with Faddeeva's we shall refer to the algorithm in (9) as *orthogonal Faddeeva.*

3. THE (CLOSE TO) TOEPLITZ CASE

In digital signal processing an often encountered problem is to solve a symmetric positive definite system of linear equations. Most frequently the coefficient matrix has a Toeplitz structure or one that is *close* to Toeplitz [10, 11, 5]. These matrices are said to have a *distance* of α to Toeplitz [12]. I.e,

$$\alpha = rank\{A - ZAZ^t\}, \tag{10}$$

where Z is the lower shift matrix :

$$Z = \begin{bmatrix} 0 & & O \\ 1 & 0 & \\ & 1 & . \\ & & . \\ O & & 1\ 0 \end{bmatrix}$$

and A is an nxn matrix. The quantity α lies in the range $1 \leqslant \alpha \leqslant n$, depending on the structure of the matrix A. For a Toeplitz matrix we have $\alpha = 2$.

The Cholesky factorization of an nxn symmetric positive definite Toeplitz matrix $A = LL^t$, with L lowertriangular, can be efficiently done by the Schur algorithm [3] at the expense of $O(2n^2)$ elementary operations, instead of the usual $O(n^3)$. The Cholesky factorization of an nxn symmetric positive definite matrix A, which has a distance of α to Toeplitz, can be done in $O(\alpha n^2)$ elementary operations with the *generalized* Schur algorithm [5]. In the generalized Schur algorithm the Cholesky factor is computed by a sequence of $(\alpha - 1)n$ (Givens) orthogonal and J-orthogonal rotations

$$\Theta(\phi_i) = \begin{bmatrix} cosh(\phi_i) & sinh(\phi_i) \\ sinh(\phi_i) & cosh(\phi_i) \end{bmatrix}.$$

The scheme is depicted in figure 2. Each box represents an orthogonal or J-orthogonal rotor and each box marked with "D" represents a delay.

Now, reconsideration of the orthogonal Faddeeva algorithm of the previous section, shows that we may decompose the QR factorization of the matrix $\underline{A}$ into 2 parts. The first part computes the factorization of the matrix A^t and the second part performs the zeroing of the vector $-b^t$. For a matrix which does not posses any structure, the factorization is just an ordinary QR factorization. But for matrices which are symmetric, positive definite and Toeplitz or *close to* Toeplitz, we can perform the Cholesky factorization by means of the Schur or generalized Schur algorithm. This algorithm does not only compute the Cholesky factor L^t, but also the inverse L^{-1} as is shown in figure 2. Hence, we can combine orthogonal Faddeeva with Schur. This is done by computing the QR factorization of the augmented matrix

$$\begin{bmatrix} L^t & L^{-1} \\ -b^t & 0_n^t \end{bmatrix}. \tag{11}$$

Substituting this matrix for $\underline{A}$ in equation (9) leads to the result :

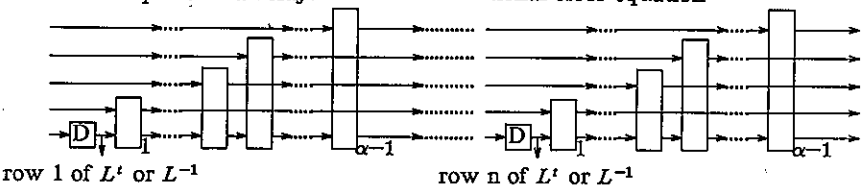
$$u_{21}^t L^{-1} = u_{22} x^t. \tag{12}$$

By adding a column $[0_n^t\ \ 1]^t$ to the matrix in (11), we can again obtain the scaling constant u_{22} from the orthogonal factor U. Orthogonal Faddeeva performed on the matrix in (11), results in a linear cascade of orthogonal rotations, because of the uppertriangular structure of L^t. The combination of the Cholesky factorization array in figure 2 and the linear cascade of orthogonal rotations is shown in figure 3.
The row $[-b^t\ \ 0_n^t\ \ 1]$ is the input to the lower input. At first sight this solver may seem completely different from the one in figure 1. But after rearranging equation (9) a little :

$$U\begin{bmatrix} -b^t & 0_n^t & 1 \\ A^t & I_n & 0_n \end{bmatrix} = \begin{bmatrix} R & u_{22}x^t & u_{22} \\ 0_n^t & U_{11} & u_{12} \end{bmatrix}. \tag{13}$$

the similarity is apparant. Now row (1) has become $[-b^t\ \ 0_n^t\ \ 1]$ in figure 1. Hence, in both solvers one can distinguish a part that computes the factorization of the coefficient matrix A and a part that computes the scaling constant and the scaled solution (the last row for the array in figure 1). The last part is identical in both arrays. It is only the factorization part that is different, because of the exploitation of structure in the coefficient matrix A.

In [10] a concurrent, multi systems of equations solver was described, very much alike the one in figure 3. However, in [10] the computation of the solution vectors was not derived via equation (11) and (12), but via a Levinson type of recursion. This was possible, because of the availability of an additional error equation



row 1 of L^t or L^{-1} row n of L^t or L^{-1}

Figure 2. Cholesky factorization of a matrix with a *close to* Toeplitz structure.

$$\epsilon^2 = \epsilon_0^2 - x^t b,$$

besides the equation Ax=b. Combining both equations :

$$[1 \; x^t]\begin{bmatrix} \epsilon_0^2 & -b^t \\ -b & A^t = LL^t \end{bmatrix} = [\epsilon^2 \; 0_n^t]. \qquad (14)$$

allows one to perform a Levinson type of recursion to be performed on equation (14), by means of J-orthogonal rotations or orthogonal rotations, depending on the positivity of the matrix in equation (14). However, the analysis given above showed that we can always use the minimum set of information (the coefficient matrix A and the known right hand side vector b) to find the solution vector, using always orthogonal rotations.

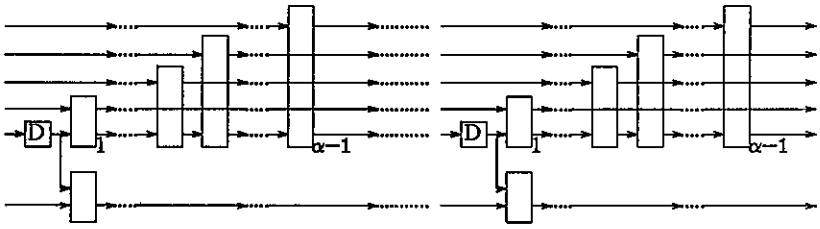


Figure 3. Orthogonal Faddeeva for a symmetric, positive definite *close to* Toeplitz system of equations.

CONCLUDING REMARKS

In this paper we have presented a novel algorithm for solving systems of linear equations. Its numerical properties are superior to that of the conventional LQ and QR method, because *only* (J-) orthogonal operations are involved. Even when the systems of equations appears to be singular, the algorithm will not break down, although the calculated solution will be probably a nonsense one. Possible numerical instability may occur when the norm of the solution vector appears to be very large. In that case the scaling constant U_{22} in equation (8) may become almost zero within the finite precision of the machine and the division by u_{22} may blow up.

Orthogonal Faddeeva has a higly regular data flow, which is strictly local and the computations are the same throughout the algorithm. This is clearly shown in figure 1. Such a nicely regular and repetetive architecture is a good candidate for a VLSI implementation of the algorithm, provided that the design of the processing elements (rotors) is feasible in VLSI technology. The regularity is preserved when the systems of equations are Toeplitz or close to Toeplitz and the algorithm is combined with the Schur or generalized Schur algorithm. If pipelined CORDICs [12] are used as processing elements, the (systolic/wavefront) array in figure 1 may achieve very high throughputs.

References

1. B.Porat, B.Friedlander, and M.Morf, "Square Root Covariance Ladder Algorithms," *IEEE Tran. Aut. Contr.* AC-27(4)(August 1982).
2. B.Friedlander, M.Morf, T.Kailath, and L.Ljung, "Extended Levinson and Chandrasekhar-type Equations for a General Discrete-time Linear Estimation Problem," *IEEE Tran. Aut. Contr.* 23 pp. 653-59 (1978).
3. P.M.Dewilde, A.Viera, and T.Kailath, "On a Generalized Szego-Levinson Realization Algorithm for Optimal Linear Predictors Based on a Network Synthesis Approach," *IEEE Tran. Circuits and Systems* CAS-25 pp. 663-675 (Sept. 1978).
4. P.Dewilde and H.Dym, "Schur Recursions, Error Formulas and Convergence of Rational Estimators for Stationary Stochastic Sequences," *IEEE Tran. Inf. Th.* IT-27(4)(July 1981).
5. H.Lev-Ari and T.Kailath, "Lattice Filter Parameterization and Modeling of Nonstationary Processes," *IEEE Trans. Inform. Th.* 30(1)(Jan. 1984).
6. M.W.Gentleman and H.T.Kung, "Matrix Triangularization by Systolic Arrays," *SPIE, Real-Time Signal Processing IV* 298 pp. 19-26 (1981).
7. S.Y.Kung, "On Supercomputing with Systolic/Wavefront Array Processors," *IEEE Proceedings* 72(7)(July 1984).
8. E.F.A.Deprettere , J.Bu, and F. de Lange, "On the Optimization of the Pipelined Silicon CORDIC Algorithm," *proceedings of this conference*, (September 1986).
9. V.N.Faddeeva, "," pp. 90-99 in *Computational Methods of Linear Algebra*, Dover Publications, Inc., New York (1959).
10. K.Jainandunsing and E.F.A.Deprettere, "Design and VLSI Implementation of a Concurrent Solver for N Coupled Least-Squares Fitting Problems," *IEEE Journal on selected areas in communications* SAC-4(1) pp. 39-48 (January 1986).
11. E.Deprettere and P.Kroon, "Regular Excitation Reduction for Effective and Efficient LP-Coding of Speech," *Proc. ICASSP-85*, (March 1985).
12. T.Kailath, S.Y.Kung, and M.Morf, "Displacement Ranks of Matrices and Linear Equations," *Journal of Math. Anal. and Appl.* 68(2)(April 1979).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

IMPLEMENTATION OF INTRAFRAME DCT CODEC FOR COLOUR TV SIGNALS

K. FAZEKAS, member EURASIP

Budapest Technical University, Department of Microwave Telecommunication
H-1111 Budapest, Goldmann György tér 3.

Abstract. - In this paper is shown an experimental real time intraframe DCT codec system for colour TV signals at a rate of 34,368 Mbit/sec. The component coding is used, the colour difference signals are line-sequentially transmitted and the transmission is extended to the horizontal blanking intervals. The main parameters of system correspond to the 4:2:2 standard by CCIR Recommendation 601. The 2D-DCT coding is applied by using the inverse trigonometric function. To the hardware implementation of this form only adders and table look-ups are needed. Because of non-stationary behaviour of colour TV signals only an adaptive procedure gives acceptable picture quality. The adaptation strategy is based on some activity measures. One of the bit assignment tables is switched on for each activity category. The size of transform matrix is 8x8. Computer simulations of the energy concentration of the transform coefficients make the base of the intraframe coder design. The image signals were modeled as a Markov process.

1. INTRODUCTION

In the last fifteen years long distance digital transmission and coding of television signals were the main issues in the field of digital television technique. The third or the fourth level of PCM hierarchy are used to the digital transmission of colour image signals over microwave links. The data rate of the third level of the digital hierarchy is regarded as particularly ecomical for the international exchange of television programs.
Since digital transmission involves increased bandwidth. Varions data compression techniques have been proposed and implemented. The coding procedure presented here is one of these.

The pricipal goal in the design of an image--coding system is to reduce the transmission rate requirements of the image source subject to some image quality constraint. A number of orthogonal transform coding system a two-dimensional (2D-) unitary transform is taken over an entire image or repeatedly over subsections called blocks. Fourier, sine, socine, Hadamard, Haar, Slant and Karhunen-Loeve transforms have been extensively utilized for image coding. The transformation produces an array of relatively decorralated, energy-compacted transform coefficients that can be efficiently quantized and coded for transmission. The Karhunen-Loeve (K-L) is the most efficient in terms of mean--square error (mse) but requires N^3 arithmatic operations for an NxN image and no fast algorithmic procedure seems possible. It is, however, taken as a reference for compressional performance, although other transforms are preferred in practical applications. The sequency transforms all offer fast transformations requiring $2N^2 \mathrm{ld} N$ additions/subtractions or fewer without the complex value arithmetic and storage penalty of Fourier transform. The Hadamard, slant and Haar transforms have all been used for this purpose.

The best energy compaction property of the cosine transform (DCT) may be seen from the comparative mse performances of the various transformations as a function of block size (Fig.1.). The two factors looked for in image

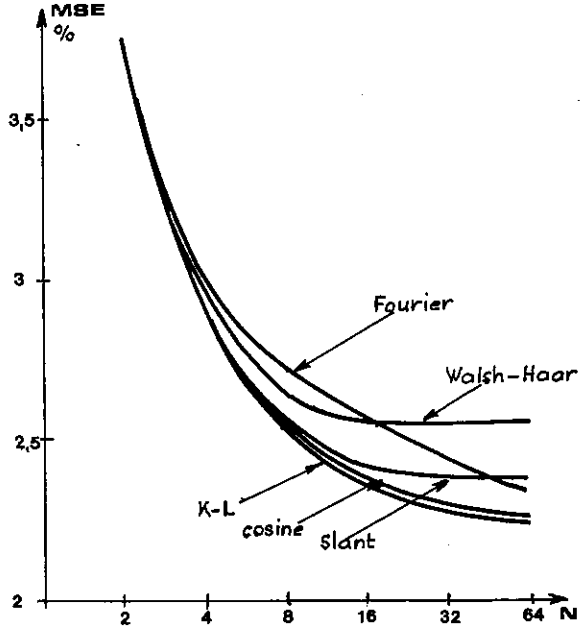


Fig.1.

coding are compressional efficiency and ease of computation. The 2D-DCT transform also provides a good approach to these two factors. In this paper is shown an experimental real time intra-frame DCT codec system for colour TV signals (SECAM/PAL) at a rate of 34,368 Mbit/sec. The component coding is used, the colour difference signals are line-sequentially transmitted and the transmission is extended to the horizontal blanking intervals. The main parameters of system correspond to the 4:2:2 standard by CCIR Recommendation 601. The 2D-DCT coding is applied by using the inverse trigonometric function. To the hardware implementation of this form only adders and table look-ups are needed. The Fig.2. shows the block diagram of the complete digital part of coder of the codec system. Due to the very high speed requirement of transform coding a parallel coding structure is applied. The 2D-DCT transform encoding is adaptive. The adaptation strategy is based on activity measures. One of the bit assignment tables is switched on for each activity category. The quantizer used here is Max's optimal quantization scheme with probability density of the transform sample modeled by Gaussian densities. An error correction unit is also applied in the system.

2. THE 2D-DCT CODING UNIT

Here is used the adaptiv block transform image coding Scheme employing the 2D-DCT. The images are subdivided into two-dimensional blocks of equal size and the 2D-DCT is applied separately and independently on each block. The resulting transform coefficients are then quantized and coded for transmission, generally under the control of an appropriate bit allocation

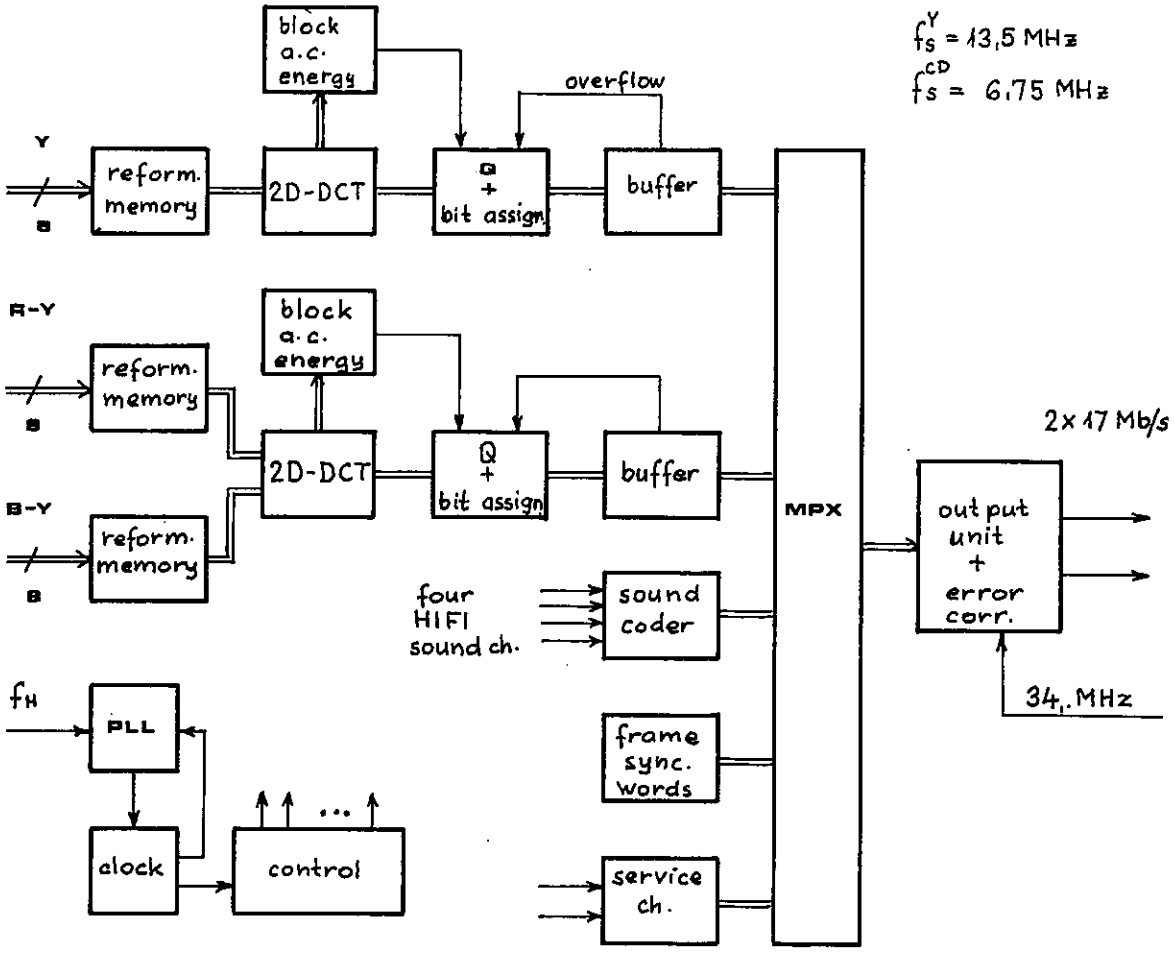


Fig.2.

algorithm. One approach in adaptive 2D-DCT coding techniques is to divide the transform blocks into a finite number of categories, or classes, according to their "activity index" which represents the amount of "activity" or "detail" in each block. Bits are then distributed among classes according to their level of activity with more bits assigned to high activity classes and fewer bits to classes with low activity.

The 2-D forward DCT is given by

$$F(k,j)=\frac{2}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f(m,n)\cos\frac{\pi k(2m+1)}{2N}\cos\frac{\pi j(2n+1)}{2N}$$

$$k,j = 0, 1, 2, \ldots, N-1$$

and similarly the 2D inverse discrete cosine transform (IDCT) is given by

$$f(m,n)=\frac{2}{N}\sum_{k=0}^{N-1}\sum_{j=0}^{N-1} F(k,j)\cos\frac{\pi k(2m+1)}{2N}\cos\frac{\pi j(2n+1)}{2N}$$

$$m,n = 0, 1, \ldots, N-1$$

where the block size is NxN. If the image data are real and $|f(m,n)| \leq 1$, one can use the inverse trigonometric function, namely

$$f^*(m,n) = \arcsin f(m,n),$$

$$-\frac{\pi}{2} \leq f^*(m,n) \leq +\frac{\pi}{2}$$

then

$$f(m,n) = \sin f^*(m,n).$$

In this case the 2D-DCT can be written as

$$F(k,j)=\frac{2}{N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\sin f^*(m,n)\cos\frac{\pi k(2m+1)}{2N}\cos\frac{\pi j(2n+1)}{2N}$$

This form of equation can be rearranged using the known trigonometric identity, namely

$$F(k,j)=\frac{1}{2N}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\left\{\sin\left[f^*(m,n)+\frac{\pi k(2m+1)}{2N}+\frac{\pi j(2n+1)}{2N}\right]\right.$$

$$-\sin\left[-f^*(m,n)+\frac{\pi k(2m+1)}{2N}+\frac{\pi j(2n+1)}{2N}\right]$$

$$+\sin\left[f^*(m,n)-\frac{\pi k(2m+1)}{2N}+\frac{\pi j(2n+1)}{2N}\right]$$

$$\left.-\sin\left[f^*(m,n)+\frac{\pi k(2m+1)}{2N}+\frac{\pi j(2n+1)}{2N}\right]\right\}$$

To the hardware implementation of the latter form only additions and table look-ups are needed. The constant values $\frac{\pi k(2m+1)}{2N}$ and $\frac{\pi j(2n+1)}{2N}$ are stored in PROM-s (or ROM-s) are used to obtain sine and arcsine values in the appropriate range. An additional ALU is used to sum terms of the given transform component.

In our case the value of N is chosen 8 to achieve moderate hardware complexity and acceptable picture quality. An adaptive 2D-DCT unit

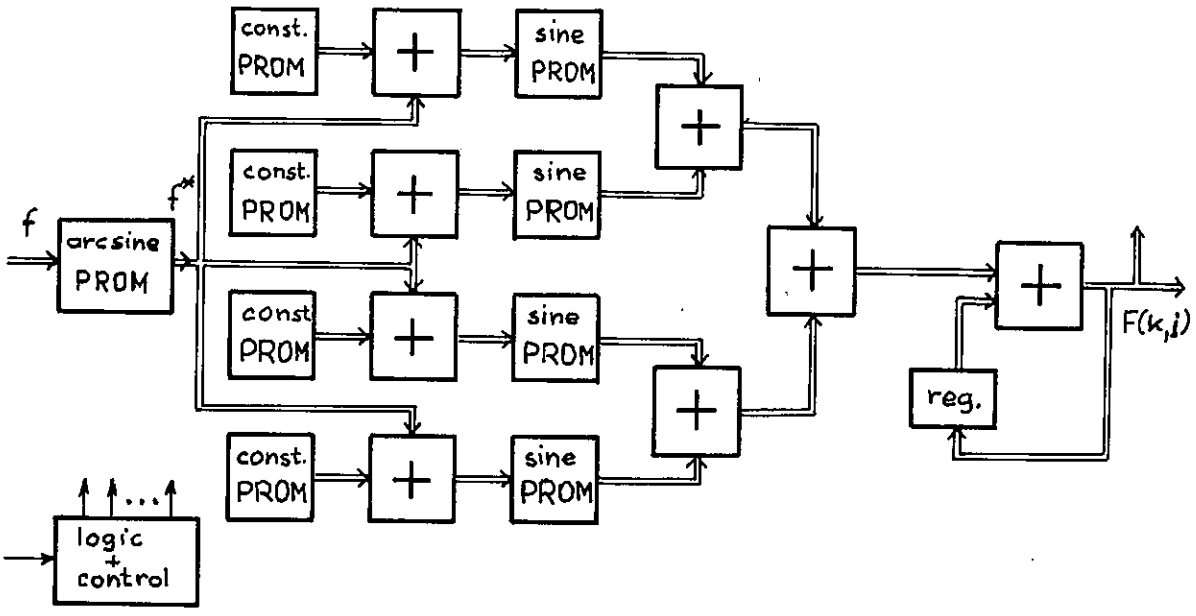


Fig.3.

have been developed to exploit the typically nonstationary nature of real-world image sources. The a.c. energy of each block has been used as the "activity index" for class assignment. An activity index is a measure of the amount of "detail" associated with each block. A fairly general activity measure, computed for the (m,n) 'th block, can be expressed in the form

$$A(m,n) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} a(k,j) F^2_{m,n}(k,j) \ ,$$

$$m,n = 1, 2, \ldots, N$$

where a(k,j) is an appropriataly defined weighting function. For example, choosing

$$a(k,j) = \begin{cases} 0 & \text{if} \quad k = j = o \\ 1 & \text{otherwise} \end{cases}$$

results in the so-called a.c. energy. Although this measure is not sensitive enough to the relatively small, yet significant, highfrequency components, here it has been used to simplify the hardware implementation. The overall block diagram of the adaptive 2D-DCT unit is shown in Fig.3.

After the 2D-DCT is performed the transform blocks are classified into four groups according to the defined measure of activity. The normalized transform samples within each class are then non uniformly quantized and adaptively coded. Naturally an overhead information is needed by the receiver to decode the received data.

Computer simulations were made to determine the distributions of the two-dimensional DCT coefficients and the typical bit allocation matrices. By the simulations the image signals were modeled as a Markov process. The Fig.4. shows the four bit allocation matrices for the Y signal.

At the receiver, the received data are decoded, and an inverse cosine transform (2D-IDCT) is performed to reconstruct the image.

3. HARDWARE IMPLEMENTATION

Parallel signal processing is applied in the 2D-DCT units of system, therefore the memory requirements are somewhat larger. Because of the very high speed requirements the elements of the ECL 10000 series are applied in the most critical parts of the units (e.g. MC 10181 type ALU; MCM 10149 type PROM; etc.). In the remaining parts of the units the S-TTL-JC-s are applied.

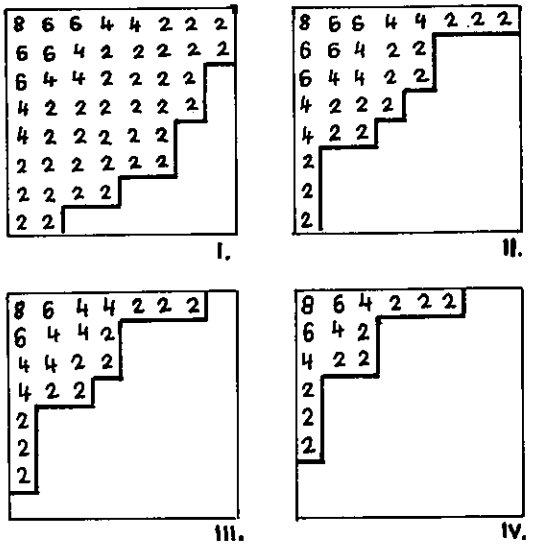


Fig.4.

4. CONCLUSIONS

The adaptive 2D-DCT coding technique described herein is applicable for real time realization. The scheme eliminates time-consuming multiplications, the DCT being accomplished with only additions and table look-ups. This procedure requires only minimal overhead information yet gives acceptable picture quality for a rate of 1,5 bits per pixel. This adaptive transform coding technique tends to propagate channel errors and some form of bit protection is desirable. Only the overhead part of the data requires error correction at channel bit error rates of about 10^{-4}.

ACKNOWLEDGMENT

The author wish to express thanks to Mr. Gy. Szondy for his useful assistance in the fulfillment of computer simulations.

REFERENCES

1 Chen, W., and Smith, C.H.,
Adaptive Coding of Monochrome and Colour Images
IEEE Trans. on Commun., Vol. COM-25, No.11, Nov. 1977 pp. 1285-1292.

2 Pratt, W.K.,
Image Transmission Techniques
Academic Press, 1979.

3 Ngan, K.N.,
Adaptive transform coding of video signals
IEE PROC.. Vol.129, Pt.F, No.1. February 1982 pp. 28-40.

4 Reiniger, R.C., and Gibson, J.D.
Distribution of the Two-Dimensional DCT Coefficients for Images
IEEE Trans. on Commun., Vol. COM-31, No.6, June 1983 pp. 835-839.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

REALIZATION OF A NONCOHERENT CPM-DEMODULATOR WITH DIGITAL SIGNAL PROCESSING

Michael Aldinger, Hans-Peter Kuchenbecker

AEG Aktiengesellschaft, Research Center Ulm
Sedanstrasse 10
7900 Ulm FR Germany

The algorithm of a noncoherent CPM (continuous phase modulation) demodulator based on correlation detection with decision feedback is described. Results based on computer simulation and measurements on hardware implementation using digital signal processors are reported. The method is applied to digital speech transmission on a narrow band mobile radio system.

1. INTRODUCTION

Digital signal transmission is generally much more disturbed by a mobile radio channel than by a transmission line. A compensation for this effect by increasing the bandwidth or the transmitted power can be achieved only to a limited extent due to existing restrictions of the radio administrations. Therefore, advanced modulation and demodulation schemes must be applied which make efficient use of the available bandwidth [1],[2]. The price to be paid for this increase of performance is the complexity of signal processing necessary especially on the receiver side. But it will be possible to fulfill these demands in the near future due to the expected progress in large scale integration.

The economic use of bandwidth is obtained by suitable coding and application of partial response signalling which introduces intersymbol interference in a controlled way. For detection of a symbol the receiver has to consider the history of the received signal over several previous symbols, and the decision should be based on the highest probability.

This can be carried out in an optimal way by the Viterbi algorithm, but the realization of such a demodulation scheme generally fails because of the great expenses of necessary computing capacity. Osborne and Luntz [3] have derived the optimum coherent receiver for CPFSK, assuming an observation interval of a few symbols of the CPFSK waveform and producing an optimum decision of one symbol. This receiver is of very complex type. The aim was to find a version with reduced complexity and with noncoherent mode of operation to save the great amount of processing necessary for phase-coherent synchronization. In this paper a suboptimal algorithm is investigated, and the relevant results of a hardware implementation are reported.

The transmitted signal originates from a wellknown modified MSK modulation using raised cosine of length 3 as a basic pulse (3RC). We tried to find out an acceptable compromise between the performance of the system and its realizability on conventional signal processor chips. The analysis of the system was carried out by computer simulation employing fixed point numbers. This tool enabled us to develop a structure of the algorithm suitable for an appropriate implementation on digital signal processors. Furthermore the computing capacity necessary for real time signal processing could be estimated.

The data rate is 10 kbit/s using a channel bandwidth of 25 kHz, and the method is applied to digital speech transmission on a narrow band mobile radio system.

2. PRE-PROCESSING OF THE RECEIVED SIGNAL

In the demodulator the digital processing of the received waveform is based on the complex envelope of the signal. For that purpose, the signal spectrum has to be shifted to the baseband, which is carried out in several steps. The received analog signal is converted by conventional means from carrier frequency to an intermediate frequency (IF) of 455 kHz, which is customary in radio equipment. The antialiasing filter (Fig.1) provides a bandlimited signal around the IF carrier. The analog-to-digital conversion is carried out with a sampling rate of 260 kHz. This undersampling of the IF signal causes periodically continuing spectra in the frequency domain, but no overlap occurs due to proper choice of frequencies.

Now the aim is to transform the spectrum next to the origin of the frequency axis to an IF of 0 Hz. This is done by digital complex mixing of the signal. Disturbing parts of the spectrum are eliminated by the succeeding digital filter. Then the sampling rate may be

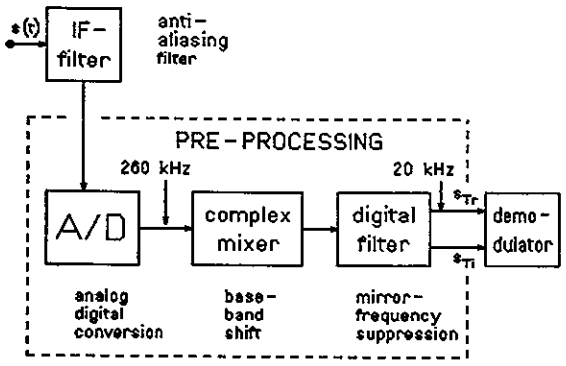


Fig. 1 Pre-processing of received signals

reduced to 20 kHz according to the effective signal bandwidth. The above mentioned procedures are shown in the frequency domain in Fig.2.

Under ideal conditions, the undersampling operation does not produce any error, for there is no aliasing between the periodically continued spectra. If there are differences between the carrier frequencies of the transmitter and the receiver, aliasing can occur. But the degradation can be kept small by applying a relatively coarse carrier frequency control in the receiver.

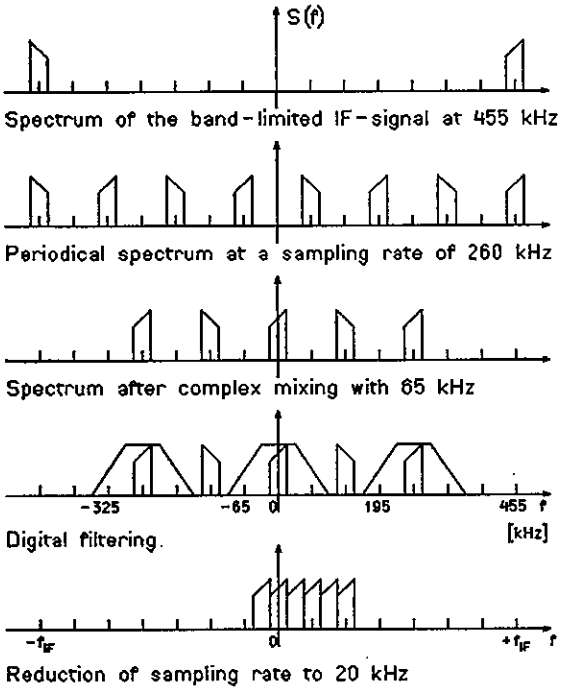


Fig. 2 Signal preprocessing in the receiver

3. STRUCTURE OF THE DEMODULATOR

As shown in [4] the signal in a partial response scheme depends on L-1 prehistory symbols, where L is the duration in symbol intervals of the basic pulse. The signal is observed over N symbol intervals, and the receiver makes a decision on the Lth symbol.

In an optimal demodulator design with L=3 and N=3 there are m = 2**(L+N-1)=32 complex correlators matched to 32 possible signal forms, which can appear in a 3 symbol observation window by superposition of shifted 3RC partial response signals. In the investigated suboptimal case the number of correlators is reduced to eight, and the references are gained by equal weight averaging of 4 original transmitter waveforms in each case.

Fig.3 shows a block diagram of the suboptimal incoherent demodulator. The receiver computes the scalar product of the preprocessed signal samples with the reference samples over a time duration of three symbols.

The arrangement of the correlators is such that, in each case, two corresponding correlators are combined where the two respective references differ only in the Lth position. The outputs D1 to D4 are the differences between two corresponding complex correlators in each case. If the sum of the four outputs yields a positive number, the device decides for a +1 as transmitted data symbol, and if it is negative it decides for a -1.

To get better results in the case when the signal is disturbed by random noise, it is tested if the absolute value of D is greater than a specified threshold V. If the result is "yes", the decision is made in the above mentioned way. If the result is "no", the decision is made with respect to the combination of two foregoing data symbols a_{n-1} and a_{n-2} which selects only the most probable correlator branch for decision on the transmitted data symbol.

4. CLOCK RECOVERY

Furthermore an algorithm for control of the receiver clock rate was implemented on the signal processor. The method is simple and needs only a small amount of processing, since the criterion for the control of the clock phase is gained from values already calculated by the demodulation algorithm. The correction of the clock rate is carried out in the pre-processing circuit in connection with the reduction of the sampling rate. By an adequate choice of the samples of the highly sampled (260 kHz) digital signal, a delay or acceleration process of the clock rate is enforced, respectively.

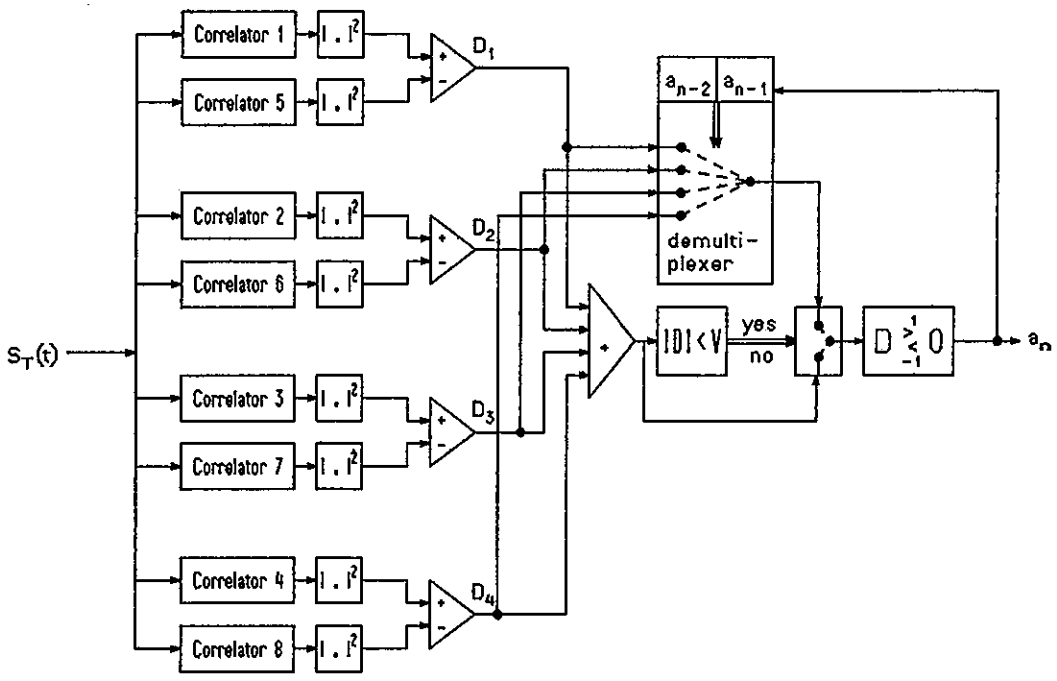


Fig. 3 Noncoherent demodulator for CPM-signals

In principle the zero crossings of the received signal are used to get a criterion for the clock control; more precisely it is the output signal of the correlation network D(t), which is closely related to the received signal. Analyzing four consecutive samples of D(t) in every symbol interval a fictive zero crossing can be estimated, provided that such a zero crossing exists within the respective interval. The position of this point relative to the transitions of the receiver clock indicates whether a correction of clock phase is necessary or not.

Three measures are taken against noise and jitter:

- To suppress zero crossings coming from noise, a threshold G is adjusted, and it is checked if the absolute value of the first D-sample X1 is greater than G. Only in this case a utilisation of the considered zero crossing is made. If X1 is greater than zero, the following two D-samples X2 and X3 are added to a value S which is set to zero at the begin of a control decision interval. If X1 is less than zero, the two values are subtracted from S.

- To avoid rapid fluctuations of the clock control, the average over several symbol durations is taken (e.g. 60 symbols), which define a control decision interval.

- To get further protection against unintentional fluctuations of the clock control signal, a hysteresis effect is buildt into the procedure. After an adjusted number of symbol intervals it is checked whether the absolut value of S is greater than a second threshold H. Depending on the outcome of the conditions of this test a clock control is initiated or not.

To correct a fixed frequency shift between transmitter and receiver clock, the D-signal is considered over several (for example 4) control intervals of the above mentioned type. If the total sums S in each time interval are of the same sign, a fixed frequency shift is detected, and the algorithm automatically corrects this fixed frequency shift.

Extensive simulations indicate a sufficient clock recovery, even if severe clock fluctuations and disturbances of the received signal are present.

5. RESULTS

The realization of the digital section of the modem was accomplished using signal processors NEC 7720. The performance of the suboptimal demodulator is represented by means of the average bit error rate as a function of the signal-to-noise ratio for different operating conditions including Rayleigh fading. The required data were gained for one thing by simulation of the whole transmission system on a digital computer, and for another by measurement on the hardware implementation.

At the date of printing, the hardware implementation of the clock recovery was not yet completed, thus the measurements of bit error rate are without clock recovery. The simulation programs for bit error rate estimation however are equipped with clock recovery and show satisfying performance. The results are compared with other demodulation schemes well known from the literature and are shown in Fig.4.

The four curves on the left hand side of the diagram are outcomes of a stationary channel with additive white gaussian noise. "PSK" designates the theoretical optimum, i.e. PSK with unlimited bandwidth and ideal coherent detection. The three curves marked "CPM" are results from the algorithm presented. A loss of 5-6 db against PSK can be stated, where about 3 db are due to noncoherent detection, about 2 db to the suboptimal implementation, and about 1 db to the clock recovery process.

The five curves on the right hand side of Fig. 4 include frequency- flat Rayleigh fading with a Doppler frequency of 25 Hz. "BMSK" is bandlimited MSK with ideal coherent detection and, in some sense, a lower limit for this situation. The three curves marked "CPM" are again outcomes from the algorithm presented in this paper. "TFM" represents a measurement in [5], where coherent demodulation was used. The loss of "CPM" against "BMSK" is remarkably less than in the stationary case (only 2-3 db). "TFM" shows the typical asymptotic behaviour of realized coherent detectors in Rayleigh fading, where the carrier synchronization fails during deep fades.

As a conclusion it can be stated, that the noncoherent CPM demodulator presented, shows satisfying performance under realistic conditions, and it can be realized by relatively simple means based on digital signal processing.

6. ACKNOWLEDGEMENT

The authors are indebted to Mr. Pehnack for implementing the algorithm on signal processors and to Mr. Bronner for hardware testing under realistic conditions and to both for many stimulating discussions on the subject.

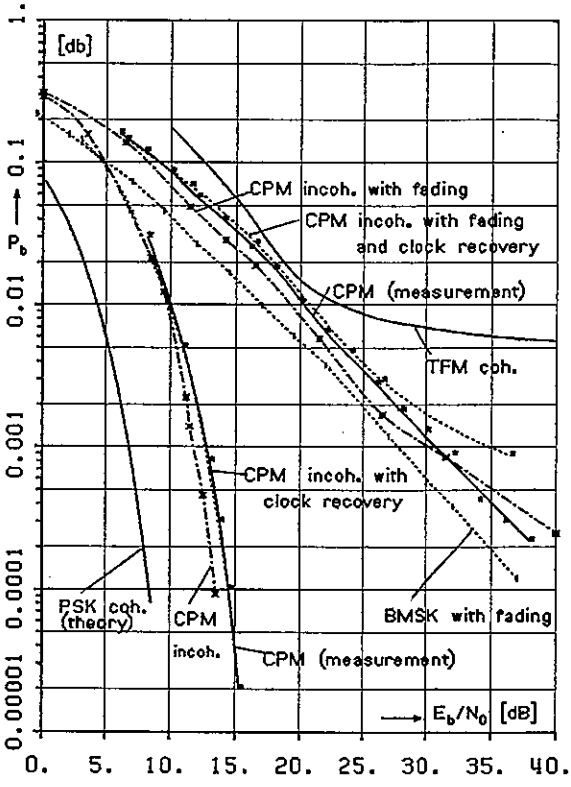


Fig. 4 Bit error probability P_b of several receiver schemes

BMSK: bandlimited MSK with ideal coherent detection.
CPM : 3RC-CPM with noncoherent detection.
TFM : Tamed frequency modulation with coherent detection (measurement) [5].

This work was partly supported by the Bundesministerium fuer Forschung und Technologie.

REFERENCES

[1] Svensson, A. and Sundberg, C.-E., Serial MSK-Type Detection of Partial Response Continuous Phase Modulation, in IEEE Trans. Commun., vol. COM-33, no.1, pp 44-52, Jan. 1985.
[2] Svensson, A. and Sundberg, C.-E., Performance of generalized AMF Receivers for Continuous Modulation, in IEE Proceedings, vol.132, Pt. F, no.7, Dec.1985.
[3] Osborne, W.P. and Luntz, M.B., Coherent and Noncoherent Detection of CPFSK, in IEEE Trans. Commun., vol. COM-22, no. 8, pp 1023-1036, Aug. 1974.
[4] Aulin, T., Rydberg, N. and Sundberg, C.-E. Continuous Phase Modulation - Part II, in IEEE Trans. Commun., vol.COM-23, no. 3, pp 210- 235, March 1981.
[5] Wellens, U., Transmission Performance of Tamed Frequency Modulation in UHF Mobile Telephone Systems, in ICC'83, B8.2.1-5, pp 549-553.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A NEW SECOND ORDER COSTAS DPLL CONFIGURATION

Aad C. Spek and Richard C. den Dulk

Delft University of Technology, Department of Electrical Engineering, Pulse and Digital Electronics Research Lab, Mekelweg 4, P.O. Box 5031, 2600 GA Delft, The Netherlands

A new second-order all digital phase-lock loop is described that can be applied in a BPSK carrier synchronizer and data demodulator. The proposed loop configuration allows the lock range to be designed independently of the noise bandwidth , unlike common (cascaded first order) loop configurations. The proposed loop can give information in a numerical format on the phase and frequency estimate of the input signal. Moreover, the configuration is ideally suitable for integration in silicon.

1. INTRODUCTION

Phase-lock loops play an important role in communication systems. With the development of large-scale integrated circuitry, there is a trend toward digital phase-lock loops (DPLL's). Some advantages of the DPLL's are their entirely digital design, programmable bandwidth and center frequency, and an accuracy that is not affected by temperature and supply voltage variations.

The errors associated with digital systems, such as quantization, round off and overflow are minor compared to the advantages.

Lindsey and Chie [1] categorize the different DPLL implementations into four classes, based on the mechanization of the phase detector (PD):

1) Lag/Lead DPLL in which the PD determines at each cycle whether the input leads or lags behind the Digital Controlled Oscillator (DCO);
2) Zero Crossing DPLL in which the loop tries to sample at the zero crossings of the incoming signal;
3) Nyquist rate DPLL in which the input is sampled at the Nyquist rate; and
4) Duty Cycle DPLL in which the phase detector delivers a duty cycle proportional to the phase error. Possible phase detectors are the EXOR and the Set Reset Flip Flop.

This article concentrates on the duty cycle DPLL. Advantages of this type of DPLL are its linearity, its relatively easy implementation and its resemblance to the analog implementation of a PLL.

2. FIRST-ORDER DUTY CYCLE DPLL

A first-order duty cycle DPLL can be implemented as is depicted in Fig.1 with a commercially available integrated circuit such as 74LS297 OR 74HC/HCT297. The DCO consists of a modulo-K counter and an Increment/Decrement (I/D) circuit [2] and is controlled by the output of a duty cycle phase detector. The K-counter consists of an up-counter and a down-counter with respectively carry and borrow outputs. The Up/Down input controls which part (up or down) of the K-counter is in operation at a particular instant. The frequency of the carry pulses minus the frequency of the borrow pulses is proportional to the duty cycle from the phase detector. The I/D circuit produces an output frequency equal to one half I/D clock frequency when no carry or borrow is in process (Fig.2 waveform b). Carry pulses will be added to half the clock frequency (waveform c), while borrow pulses are subtracted from half the clock frequency (waveform d). The I/D output frequency varies linearly with the input duty cycle.

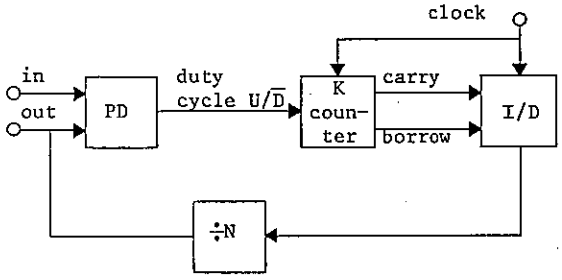


Fig.1 First-order DPLL

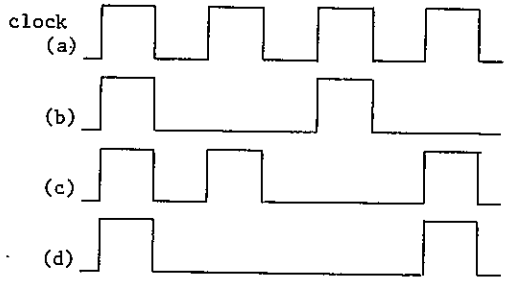


Fig.2 I/D circuit waveforms

Since the output transitions of the DPLL are synchronous with the clock used, the output phase is quantized. The ratio of the clock frequency and the output frequency, fixed by the divide by N-counter behind the DCO, determines the phase resolution.

The behavior of the first order DPLL is equivalent to a first-order analog PLL with a linear PD characteristic. The noise bandwidth is determined by the loop gain, while the lock range is determined by both the maximum phase error and the loop gain. Lock range and noise bandwidth can therefore not be chosen independently. Since the first order DPLL proposed by Iritani [3] uses an edge triggered PD with a linear range modulo $N \cdot \pi (N>>2)$ the noise bandwidth can be chosen regardless of the lock range. However, a disadvantage of a PLL employing an edge triggered PD is its degraded noise performance.

In order to create independent loop parameters as in the analog case, a higher order loop is needed.

3. HIGHER ORDER DPLL's

Digital loop filters in this category cannot be implemented easily. It is difficult to realize a digital integrating filter using duty cycles as input and output signals. It is therefore customary to create higher order loops by cascading first-order loops. In the literature all the possible ways of cascading first order loops to create a second-order DPLL are shown [3]. By cascading two first-order DPLL's and controlling the free running frequency of one DCO by the other phase error a second-order DPLL is constructed.

Although a perfect second-order loop can be created, the lock range is still determined by a first-order loop, and as a consequence the noise bandwidth and lock range can not be chosen independently.

4. THE NOVEL SECOND ORDER DPLL

The proposed perfect second-order loop consists of only one loop. As is known a loop filter can be realized by adding a proportional and integrating path [4]. In our configuration (see Fig.3) the proportional path consists, as in the first-order configuration, of Fig.1, of the modulo-K counter and the I/D circuit. The integrating path is formed by the modulo-K counter, Up/Down Counter (UDC) and the rate multiplier (RM). A rate multiplier is a digital circuit that multiplies an incoming frequency by a digital number $K=P/Q$ (P,Q integer, $K \leq 1$) [5]. The integral and proportional path are added in the I/D circuit after they have been converted to frequencies.

The operation can be explained as follows: Assumé a step function behind the phase detector (the control duty cycle changes from 50 to 100 percent). At a 100 % duty cycle the K-counter counts up only, generating carry pulses. These pulses have two disjoint effects:
-The carry pulses are added in the I/D circuit to half the fraction (P/2Q) of the clock frequency, generating a DCO frequency step (proportional path of the loop filter).
-The Up/Down counter counts up every carry pulse, so the rate multiplier will generate a frequency ramp (integrating path of the loop filter).

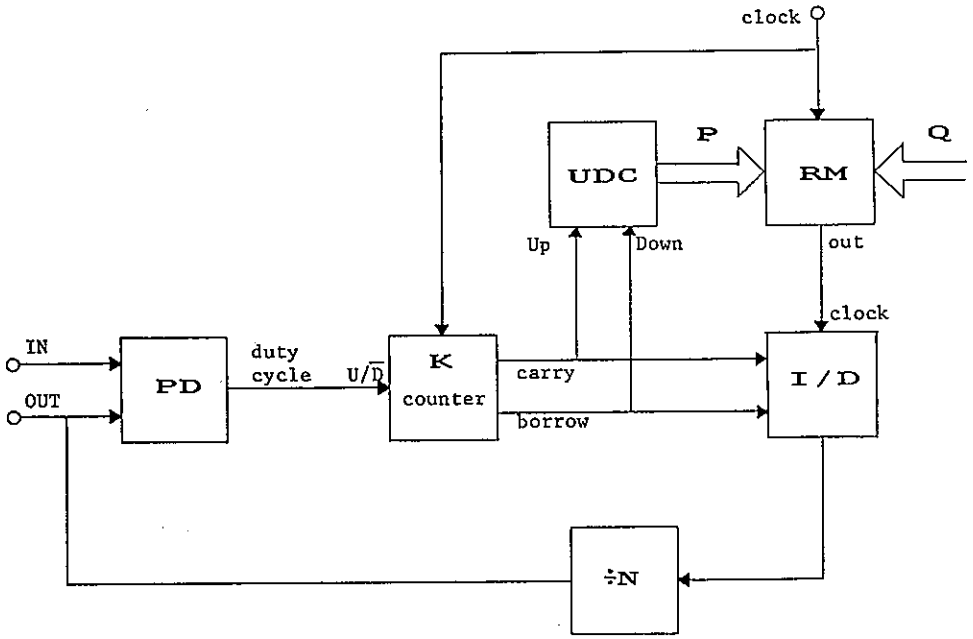


Fig.3 Proposed second-order DPLL

The behavior of the proposed second-order loop is equivalent to a perfect second-order analog PLL. The lock range theoretically lies between Fclock/2N and zero. The lock range can be limited by using an Up/Down counter with limitations. The noise bandwidth can be chosen by varying the proportional and integral gain. The lock range and noise bandwidth can be chosen fully independently, unlike the cascaded first-order configurations.

4.1 Acquisition of second-order duty cycle DPLL's

The acquisition behavior is quite similar to that of a perfect second order analog PLL. For small initial frequency errors the loop will lock within one phase transient (phase acquisition). This phase aquisition range is determined by the proportional loop gain.

For larger initial frequency errors the loop can also reach phase lock after skipping some cycles (frequency acquisition). The frequency difference results in a beat note behind the PD. The controlled oscillator will be phase modulated by this beat note. In the analog case this results in a pull-in voltage and in the digital case a pull-in duty cycle. The pull-in range is limited by the phase quantization: The DCO phase modulation must exceed this quantization to deliver the pull-in duty cycle.

4.2 Phase jitter of duty cycle DPLL's

The DCO output signal will show phase jitter, even if the input signal is noise free. The zero crossings of the DCO are always synchronous with the clock transitions. For an optimally spaced output signal the time jitter amounts to + or - 1/2 clock period, while the phase jitter is determined by the ratio of the clock frequency and the output frequency.

The output signal of a first order DPLL can be optimally spaced. However the output signal of a second order DPLL is more badly spaced since there are always two independent quantizations.

For our DPLL this results in a time jitter of + or - 1 clock period if the loop is locked on to the upper lock range limit (F-upper = Fclock/N). For lower input frequencies the time jitter will increase since the I/D clock frequency is reduced, so that the quantizations in the proportional path are magnified (see Fig.3). To minimize this additional jitter the I/D clock frequency has to be chosen as high as possible: This implies a divide by N-counter as large as possible. At F-upper/2 the time jitter already amounts to + or - 1.5 clock period.

5. NEW COSTAS DPLL APPLICATIONS

DPLL's can only handle input frequencies an order of magnitude lower than the clock frequency used. However, if bandwidth and carrier/sample frequency requirements can be fulfilled [6] input frequencies much higher can also be applied, because the digital equivalent of the S&H circuit mentioned in [6] can be used as a "harmonic mixer". However, the common baseband Nyquist applications are FSK detection, transition tracking clock synchronizers [7], motor speed control, noise filtering, tone recognition, frequency synchronization and multiplication.

The proposed loop is ideally suitable for a Costas loop carrier synchronizer and data demodulator for BPSK detection because in lock no static phase error exists. The fundamental difference between a Costas PLL and a conventional PLL in the analog case is the PD characteristic. This characteristic is periodic, recurring every 180 degrees, thereby wiping out the effect of the BPSK modulation. By multiplying the outputs of two quadrature phase detectors a 180 degree periodic PD characteristic is created (see Fig 4).

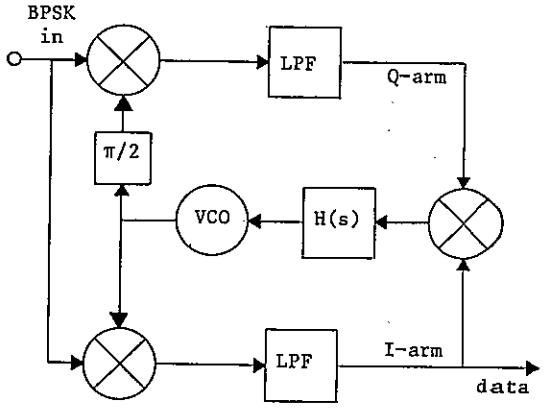


Fig.4 Analog Costas implementation

A simpler Costas implementation uses a hard limiter in the I-arm [8]. The limiter signal only decides whether the loop gain of the loop formed by Q-arm PD the loopfilter and the VCO has to be reversed. This configuration has been implemented digitally (see Fig.5). The loop gain reversals are realized by inverting the

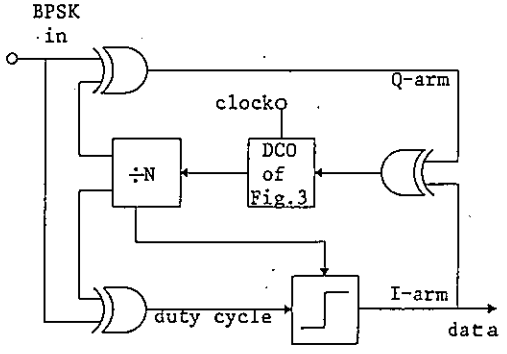


Fig.5 Digital Costas implementation

Q-arm duty cycle in the third EXOR. Since arm filters using duty cycles as input and output signals can hardly be implemented, the input noise bandwidth has to be determined by a band-pass filter.

Special attention must be paid to the I-arm hard limiter: This limiter has to determine whether the I-arm duty cycle is greater or less than 50 %, so a duty cycle hard limiter is required. This limiter can be implemented by (see Fig.6):

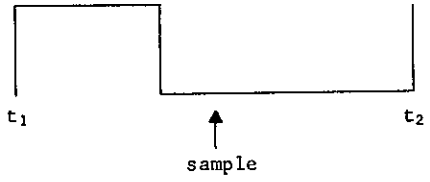


Fig.6 output signal of I-arm phase detector

-Integrate and dump. The interval t_1-t_2 is known since this is half a output period. During this interval the duty cycle can be integrated in an Up/Down counter. At t_2 a decision is made and the counter will be reset for the next cycle. A fast Up/Down counter is needed.
-Assume the duty cycle has only one transition during t_1-t_2 (see Fig.6). As long as this assumption is valid the duty cycle can be hard limited by sampling in the middle of the interval t_1-t_2. This Costas implementation uses only two EXORs and a D type flip-flop in addition to our proposed normal second-order DPLL.

6. CONCLUSION

With the configuration described the lock range and noise bandwidth of all-digital phase-locked loops can be designed independently. The analysis of the loop has shown that the acquisition is equivalent to analog PLL's and the residual phase jitter is inherent to the quantizations of time which exist in every digital mechanization of a PLL. A perfect second-order Costas loop can be implemented easily.

REFERENCES

[1] Lindsey W.C. and Chie C.M., A survey of digital phase locked loops, Proc. of the IEEE, vol. 69 no.4 April 1981, pp 410-431.
[2] Data sheet PC74HC/HCT297, Philips Elcoma Division, IC06N, pp.529-536, Jan.1986 and Data sheet SN54/74LS297 , The TTL Data Book, Texas Instruments , Ch.7 pp.449-454 , 83/84.
[3] Iritani T., Linear digital phase-locked loops using integrators in a pulse frequency-modulation system, IEE Proc, vol. 129, Pt.f, no. 5, Oct 1982, pp 352-358.
[4] Gardner F.M., Phaselock Techniques , (New York, Wiley, 1979)
[5] Den Dulk R.C. and Stuyt J.J., A versatile CMOS Rate Multiplier/Variable Divider, IEEE Journal on Solid State circuits, vol SC-18, no.3, June 1983, pp.267-272
[6] Den Dulk R.C., Well-defined sub-Nyquist sampling-frequency range limits, this volume
[7] Troha D.G. and Gallia J.D., Aufbau und arbeitsweise digitaler PLL schaltungen, Elektronik, vol.33, no.15, July 1984, pp 63-64.
[8] Cahn C.R., Improving frequency acquisition of a Costas loop, Proc of the IEEE, vol.69, no.4, April 1981, pp.410-431

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ARCHITECTURE FOR OPTIMAL 2-D SIGNAL PROCESSING*

A.P.J. Engbersen

IBM Zurich Research Laboratory, 8803 Rüschlikon, Switzerland

In the past years, several signal-processor architectures have been proposed, which constitute solutions for the computation of one-dimensional convolution algorithms. The principle of these architectures lies in its combination of I/O, arithmetic and multiplication in one machine cycle. In combination with the internal data-path structure, this allows a result of an n-point convolution in n + 1 steps to be delivered into the data memory of the processor. Here, the principle of this architecture is generalized for 2-D convolutions of digital signals. Problems involving this kind of convolution occur regularly in image-processing applications, and are also known under the name of window operations. The architecture proposed claims to attain a maximum cost-performance figure, by using only N + M registers, one hardware multiplier and one arithmetic logical unit (ALU). Notwithstanding this rather low set of resources, in every instruction cycle one product of a 2-D convolution sequence is delivered. Examination of the structure reveals its simplicity, which allows a potential hardware implementation to have very fast instruction cycles, typically less than 100 nanoseconds. A program written for the architecture, which implements a 3 × 3-window convolution:

$$\text{result} = h1 \times x1 + h2 \times x2 + \cdots + h9 \times x9$$

uses exactly ten instructions to obtain the result desired. Proof will be given that ten instructions is the theoretical lower limit to compute this equation with a single processor system.

1. INTRODUCTION

Digital signal processors (DSP's) combine the compactness and consistency of digital circuitry with the flexibility of programmable devices. Because of the real-time nature of the application environment, DSP architectures are optimized with respect to speed [1]. Normally, this requires efficient implementations of ALU, multiplier, registers and internal busses.

With the advent of low-cost gate array technology, special tailored architectures become closer than ever before to being an efficient solution to a problem. At the same time, however, this means that there is a need for carefully balanced and optimized architectures to achieve an optimum cost/performance ratio.

*This work was performed when the author was a Pre-Doc at the IBM Zurich Research Laboratory from 1980 through 1983, and is partly taken from his Thesis.

2. 2D CONVOLUTIONS

A two-dimensional convolution in the time domain can be represented as:

$$y(m,n) = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} u(k,l) \times h(m-k, n-l) \ ,$$

which represents filtering a signal u(m,n) with a finite impulse response (FIR) filter given by impulse response h(m,n). When we assume that h(m,n) has only non-zero values for some (m,n) tuples, then the operation is generally known under the name "neighborhood filtering". Kruse [2] has shown that all FIR filters for 2D filtering can be reduced to a sequence of so-called 3 × 3 filters:

$$h(m,n) \text{ not zero ONLY for } -1 <= m <= +1, \text{ and } -1 <= n <= +1.$$

Operations with these 3 × 3 filters are very common in image-processing applications.

3. LOWER LIMIT ON 3 BY 3 FILTERING

A 3 × 3 filter has nine coefficients, which for simplicity will be called h1 through h9. The evaluation of one new output element y(m,n), being the result of filtering the input data u(m,n) with h(m,n), requires nine multiply-add operations:

$$
\begin{aligned}
y(m,n) = \\
&u(m-1,n-1)\times h1 + u(m-1,n)\times h2 + u(m-1,n+1)\times h3 + \\
&u(m,n-1)\times h4 + u(m,n)\times h5 + u(m,n+1)\times h6 + \\
&u(m+1\times n-1)\times h7 + u(m+1,n)\times h8 + u(m+1,n+1)\times h9.
\end{aligned}
$$

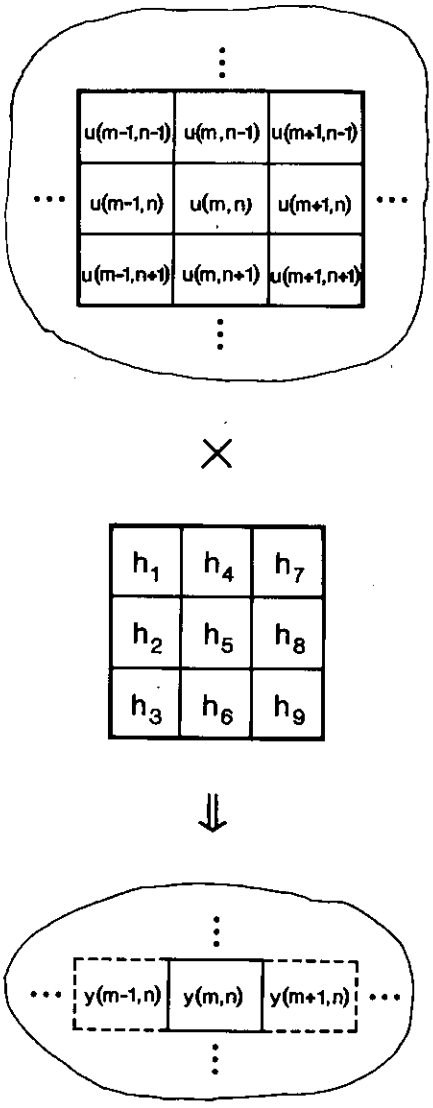


Figure 1. Spatial organization of the filter operation.

Figure 1 shows the spatial organization of these coefficients. Clearly, if a single processor has to carry out these nine multiply-add operations, at least nine cycles are required. Only by adding more multiply-add units to the processing unit can a higher throughput be achieved. However, this will no longer be regarded as a single-processor unit, based upon increased complexity.

At the end of the nine multiply-add operations, the result of filtering one point is available. This result has to be stored into the processor memory. Finally, the processor has to initialize itself for starting the process for the next point. All this data movement will normally require a tenth cycle.

4. ARCHITECTURE

Assume that the DSP architecture under consideration has separate busses to access data and instruction memory as is usual in these architectures [3]. When we further concentrate on the data memory access mechanism and the configuration of the components such as ALU, accumulators, multiplier and registers, it can be stated that an optimal architecture for a certain application is obtained whenever the data memory access mechanism and the computation part can be kept busy as much as possible throughout the process.

Accessing data outside the actual processing unit is a slow and costly operation, as opposed to accessing data in internal registers: per cycle, the memory access mechanism will transfer one data element between processing unit and memory, while the processing unit can execute a complete arithmetic operation on the contents of internal registers. This means that once a data element has been brought into the processing unit, it has to be used for as many cycles as possible. In particular, referring to Fig. 1, once an input element, say u(m,n-1) has been fetched, it is used in three multiply-add operations, one to compute y(m-1,n), one to compute y(m,n), and one to compute y(m + 1,n). This requires that filter coefficients h4, h1, and h7 are also used. These three filter coefficients can reside in internal registers to the processing unit.

However, when three arithmetic operations are carried out, and only one memory access operation, the memory access mechanism is only used 33%. This indicates that too much internal storage has been used or was available for the operation. In fact, the two free memory cycles could fetch two of the three filter coefficients into one internal register, instead of using three registers.

The three multiply-add operations are partial results for the computation of three output elements, which requires three internal accumulator registers to keep the results from interfering with one another.

Figure 2 shows an architecture developed according to the above observations. The following features will facilitate understanding of the operation of the architecture.

1. Memory data can be loaded into registers W1, W2, W3, and D.
2. W1 through W3 are the so-called weight-coefficient registers, and can be selected to feed the multiplier.
3. The multiplier will always start multiplication of the values in the selected weight and D registers, without any special instruction to initialize the operation. At the proper time, selecting the M-register as ALU input, the result of a multiplication can be obtained.
4. The ALU can combine (add) the contents of a selected accumulator A1, A2 or A3 with M, and store the result of this operation in either an accumulator, or in memory in one cycle.
5. Instruction fetch pipeline will not be flushed if a branch is taken.

Assuming that the multiplier has a pipeline-delay of one instruction cycle, the following program implements a 3×3 time domain convolution:

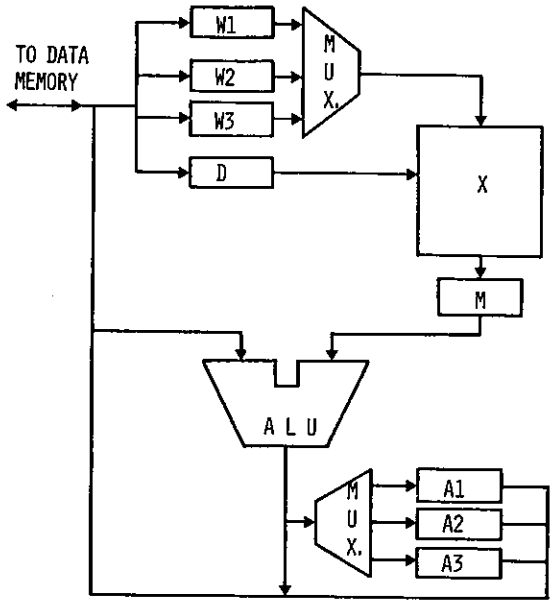


Figure 2. Principle of proposed architecture.

'Lxx <addr>' means load register xx from address <addr>

'Ax' designates an accumulator

'STORE' <addr> means store the output of the ALU to <addr>

'out' stands for an output address (not further specified).

'I' designates an index register.

Assume that an initiation phase has loaded W1 with h1, W2 with h2 and W3 with h3. The last two arithmetic operations of the program shown below, can be specified together with the first two instructions when a proper initiation is provided.

```
LD        data+1+I
LW1 W1    h4
LW1 W1    h7          A1:=M          *u1×h1
LD  W1    data+2+I    A2:=A2+M       *u1×h4
LW2 W2    h5          A3:=A3+M       *u1×h7
LW2 W2    h8          A1:=A1+M       *u2×h2
LD  W2    data+3+I    A2:=A2+M       *u2×h5
LW3 W3    h6          A3:=A3+M       *u2×h8
LW3 W3    h9          A1:=A1+M       *u3×h3
LD  W3    data+4+I    A2:=A2+M       *u3×h6
STORE     out         A3:=A3+M       *u3×h9
LW1 W1    h4          A3:=A2         *u3×h9
LW1 W1    h1          A3:=A3+M       *u4×h7
LD  W1    data+5+I    A1:=A1+M       *u4×h4
LW2 W2    h5          A2:=M          *u4×h1
LW2 W2    h2          A3:=A3+M       *u5×h8
LD  W2    data+6+I    A1:=A1+M       *u5×h5
LW3 W3    h6          A2:=A2+M       *u5×h2
STORE     out         A3:=A3+M       *u6×h9
BRA       top         I:=I+6         *incr. index reg.
LW3 W3    h3          A1:=A1+M       *u6×h6
                      A3:=A1         *u6×h6
                      A2:=A2+M       *u6×h3
```

This algorithm uses an average of 10.5 instructions to produce one result for the 3×3 neighborhood convolution. The average use of the I/O part is 20/21*100% = 95%, that of the multiplier is 18/21*100% = 86%. Note that updating the index register and switching the accumulators is done without interfering with the convolution process.

5. CONCLUSION

A new, simple architecture has been presented for time domain FIR filtering. The architecture achieves optimal performance thanks to the perfect balance of I/O operations, arithmetic operations and resources such as ALU, registers, etc.

Based on the simple interconnection of these resources, it is expected that VLSI implementation, for instance, in gate array technology, will achieve cycle times less than 100 nanoseconds. Filtering one million elements with a 3 × 3 filter then becomes feasible within one second.

Finally, the architecture presented is extendable to accommodate N × M impulse response filter matrices at the expense of only N weight coefficient and M accumulator registers, as opposed to the N × M registers required in classical architectures to achieve maximum throughput.

REFERENCES

[1] Ungerboeck, G., Maiwald, D., Käser, H.P., and Chevillat, P.R. The SP16 Signal Processor, in: Proceedings ICASSP 84, (IEEE, 1984) pp. 16.2.1-16.2.4.

[2] Kruse, B., IEEE Trans. Comput., Vol. C-22 (Dec. 1973) pp. 1075-1087.

[3] Magar, S.S., Caudel, E.R., and Leigh A.W., Digital Signal Processors, in: L. Winner (ed.), Proceedings ISSCC 82, (IEEE, 1982) pp. 32-33.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A FRAME-BASED MULTI-RULE NETWORK SYSTEM STRUCTURE FOR SIGNAL PROCESSING

X. LI, P. MORIZET, P. GAILLARD

UA 817, Dépt. Génie Informatique, Université de Compiègne
60206 Compiègne cedex, France

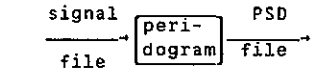
The system to be presented is composed of two major components : a modular signal processing software system and an expert system having global database plus distributed expert system structure. The modular signal processing software system provides a numerical computation support. The global database allows a structural representation of the objects in signal processing environment. It defines an internal vocabulary and provides a support of information sharing for expert systems. The expert systems are separated or distributed, and are proposed to be constructed around signal processing modules. Thus, problem solving in such a system becomes distributed problem solving and is accomplished by the process of the problem decomposition, kernel sub-problem solving and result synthesis phases.

1. INTRODUCTION

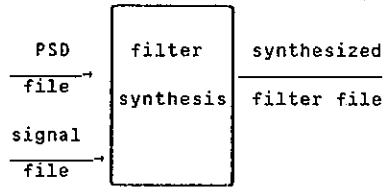
Knowledge-based systems have been built in many areas. In signal processing, one can cites for instance the HEARSAY II speech understanding system [1], the HASP/SIAP in application of underwater acoustics [2], the low level image segmentation system [3] and others [5], [4].

In signal processing, many software systems, CRISAL for example [15], have been developed and commercialized. It is modular, easily extendible, and with data file manipulation ability. Generally, problem in signal processing can be resolved within this kind of system by simply the application of several modules. Experiences in building the spectral analysis expert system [4] show that it is possible to construct a general knowledge based signal processing system by developing expert systems around modules.

Application of a module such as the periodogram to a signal file requires the determination of a set of parameters and will produce a result file of type of power spectral density (Fig. 1a). A file of the type of power spectral density contains its proper qualifying attributes. If the signal is to be filtered, then both the signal file and the result file will be used in filter synthesis. Application of a module requires, in fact, a precondition which can be represented by "input" files, and may produce some "output" files (Fig. 1b). If an expert system is to be built around a module, only the knowledge associated with the module and that with the input files is required, and the knowledge associated with the output files is derived as a result.



a) application of a module



b) "input" & "output" files of a module

Fig. 1 : Model of a module

2. ENVIRONMENT AND DATABASE

The signal processing environment consists of three kinds of objects : the signal files, the result files and the modules. This environment can be easily represented by a database having frame-like structure [6], [7]. Fig. 2 is example of module objects. They are hierarchically classified by the a-kind-of relation.

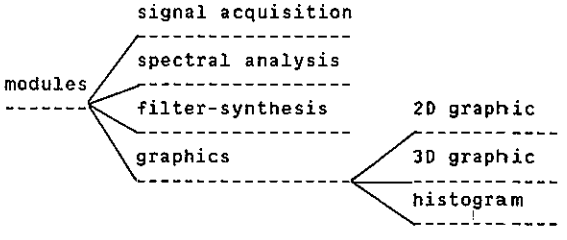
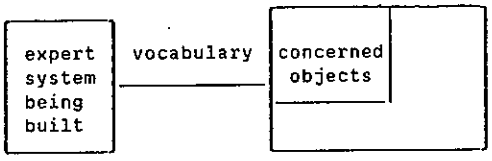


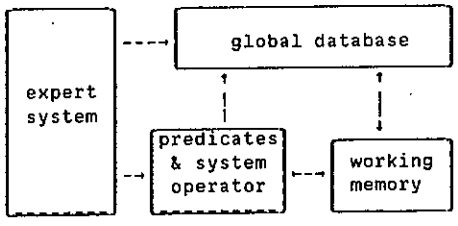
Fig. 2 : Example of module objects

Frame representation language (FRL) provides facilities for describing object attributes. A frame representing a signal, for example, might include descriptions of signal length, sampling frequency, mean, etc. These facilities allow frames to include partial descriptions of attributes values, and help preserve an uniform internal vocabulary. The a-kind-of relation in FRL enables descriptive information to be shared among similar frames via inheritance. Other relations like is input-of and is-computed-by, etc, can specify the relation signal module and result module.

Fig. 3 shows the relation between the database and the rule-based expert system. In the state of knowledge acquisition, the expert system employs the vocabulary defined by the concerned objects to form the rule conditions and action, where simple logic predicate language and some system operators may be used. The predicates must reflect the relationships represented in the FRL. At the beginning of a consultation of an expert system, all the available data of the concerned objects are transfered to the working memory as data of expert system. At the end of the consultation, results derived will be normally returned through the working memory to the database and available for other consultations. Thus, the database provides, in fact, a support of information sharing for cooperating expert systems.



a) knowledge acquisition & internal vocabulary



b) state of consultation

Fig. 3 : Relation between database and expert system

3. DISTRIBUTED PROBLEM SOLVING

A distributed expert system differs from other systems in that the knowledge sources constituting its expert systems are distributed and no one expert system has sufficient knowledge sources to solve an entire problem. Distributed problem solving may be considered as a process of three phases [11], [12], [13] : the problem decomposition which proceeds until the kernel sub-problems are generated ; the sub-problem solving which necessites the cooperation among the expert systems and the result synthesis which integrates sub-problem results to achieve a solution to the overall problem. The last phase may be trivial and often depends on the ways in which the overall problem is decomposed.

It should be noted that our system is not physically-separated, a common database is used so that the information is naturally shared among expert systems. So, the rest of this section will be focused on three following issues : development of an individual expert system, generation of an initial solution plan, and dynamic planning.

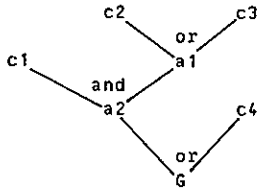
3.1 Developing expert system around module

An individual expert system has a module object in the database, a knowledge base, and other components. The module object contains in fact the meta-knowledge about the kinds of problems the expert system is interested in solving. It contains also the information about the necessary input files and potential output files. These concerned objects form the vocabulary of the knowledge base. The knowledge base may contain several knowledge sources. If the knowledge base is represented by an AND/OR network and the rule interpreter works in the goal-directed fashion [4], [8], [9], [10], all the final goal of the network to be reached may correspond to the solutions of a kernel sub-problem.

Advantages of constructing individual expert system around module are in that the expert system is also modular. It is easier to be managed and to be developed by the designer of the module. Working as a kerne sub-problem solver, it may contribute to the solution of the overall problem. Thus, the problem decomposition and dynamic planning can be done on the sub-problem level.

3.2 Initial solution plan

Problem solving plans like a set of production rules can also be represented by the AND/OR network (Fig. 4) without formal difference, where the goal G corresponds to the overall problem, the leaves C to the kernel sub-problems and nodes A to the sub-problems. Every leaf may be associated with a cost. So, if a path from leaves to the goal represents a solution of the overall problem, the cost of a path can be easily computed.



Initial solution plans of many trivial problems can be a priori inserted to a knowledge base of an expert system. Problems arise when the solution plan of an user's special problem is unknown. If the user can form the solution plan of his problem, the system can determine

1)whether the problem is solvable, this is done by examining the availability of expert systems required in a path ;

2) whether the causality is respected, this is done by examining the input output file relations among modules.

It is prefered that all the solution plans would be inserted to a single knowledge base. In this case, the problem is solved by simply "calling" other expert systems for kernel sub-problem solving. If expert systems "call" each other, the control will no longer centralized, but also distributed. In most cases, we have a mixture of more contralized and less distributed control.

3.3 Dynamic planning and sub-problem solving

There is many solutions to the problem of dynamic planning [13]. The HEARSAY II speech understanding system employs a data-driven control mechanism [1]. The goal-directed mechanism is employed here. After the solving of a sub-problem, the state of knowledge represented by the available objects in the database is taken into account. The solving process is directed along the most promising path to the final goal. A backtracking mechanism is implemented and can be used at the end of a kernel sub-problem solving. This mechanism is often desired if a consultation is to be repeated.

4. TESTING AND MANAGING

A distributed expert system is characterized by the completeness and exactness of the knowledge sources constituting its expert systems. For the system having global database plus distributed expert system structure, reasonable objects classification and appropriate distribution of knowledge sources among expert systems are of primary importance. Definition of internal vocabulary and development of individual expert system may be done by the designer of modules. In a distributed system, the solution of a sub-problem depends not only on the availability of results of its descendant sub-problems but also on their qualities. So, in addition to the test of the logic of expert systems, the system should be tested under simulated driven environment in order to see the change of states of knowledge.

Developing and testing an expert system of large size is a difficult and time consuming task. Constructing expert systems around modules allows them to be modular and of relatively smaller size. As a result, most all of the managing tasks are done on the level of individual object and expert system.

5. CONCLUSIONS

A distributed expert system has been presented at the level of system structure. This system is designed for signal processing applications and has a CRISAL system as support of numerical computation. The basic idea of constructing "modular" expert systems around modules is based on the experiences accumulated in building a spectral analysis expert system.

The feasibility of such a structure was studed and the program implemented in the VAX/LISP environment [14]. This program is being implemented in Pascal language in order to have the possibilities of implementation on other machines having no LISP environment. This work is being completed, current results appear to be interesting.

References

1) Erman,L.D., Hayes-Roth,F., Lesser,V.D. and Reddy,R.D., "The HEARSAYII speech understanding system : integrating knowledge to resolve uncertainty", ACM Computing Surveys, june 1975.

2) Nii,H.P., Anton,J.J., Feigenbaum,E.A. and Rockmore,A.J., "SignaltoSymbol transformation : HASP/SIAP case study", The AI MAGAZINE, Spring 1982.

3) Nazif,A.M. and Levine,M.D., "Low level image segmentation : an expert system", IEEE Trans PAMI, sept. 1984.

4) Li,X., Morizet-Mahoudeaux,P., Trigano, P. and Gaillard,P., "A spectral analysis expert system", IASTED International Symposium Applied Signal Processing and Digital Filtering, june, Paris, frame

5) Kopec,G.E., "The integrated signal processing system ISP", Proc. ICASSP, pp. 8.1.1-8.1.4, 1984.

6) Minsky,M., "A framework for representing knowledge", In the Psychology of Computer Vision, P.Winston, Ed. McGrawHill, pp. 221-277, New York, 1975.

7) Fikes,R. and Kehlu,T., "The role of frame-based representation in reasoning", Communication of the ACM, sept. 1985.

8) Hayes-Roth,F., "Rule-based system", Communication of the ACM, sept. 1985.

9) Lauriere,J.L., "Représentation et utilisation des connaissances : 1 et 2", TSI, 1982.

10) Fontaine,D. and Le Beux,P., "An expert system for rubella consultation", MEDINFO84, pp. 529-532, North-Holland, 1983.

11) Fox,M.S., "An organization view of distributed system", IEEE Trans SMC, vol. 11, N°. 1, jan. 1981.

12) HayesRoth,B., "A blackboard architecture for control", AI, vol. 26,N°. 3, july, 1985.

13) Yang,J.D., Huhns,M.N. and Stephens,L.M., "An architecture for control and communications in distributed artificial intelligence systems", IEEE Trans SMC, may/june 1985.

14) Steele, Guy, Common LISP : the language, 1984.

15) UTC-CRISTAD Society, "CRISAL, v. 0.1, Guide de l'utilisateur", 1983.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A SYSTEM FOR REAL-TIME PROCESSING OF DATA AT 45 MEGA-SAMPLES/SECOND AND BEYOND

Louis Schirm IV

DSP Systems Corp.
1081 N. Shepard St. Mail/stn E
Anaheim, CA USA 92806

ABSTRACT

The GOPS™ Matrix Processor is a special purpose, high speed vector processing machine, applicable to classical one-dimensional signal processing tasks such as transversal filtering, auto or cross correlations, adaptive equalizers, line enhancers or cancellers, etc. Each input array can be either limited or infinite length. (If both are infinite arrays, then the output is a limited burst of answers after a trigger pulse, the number of answers being a function of the number of taps in the machine. Since the heart of the machine is purely digital, the system inputs and outputs may be either analog (via converters) or digital. The arithmetic unit cards are configured so that there is great flexibility in the system architecture to accomodate very high bandwidth (up to 50 MHz analog bandwidth) as well as large matrix size.

The GOPS Matrix Processor is a special purpose, high speed vector processing machine applicable to classical one-dimensional signal processing tasks, such as FIR filtering, auto or cross correlations, adaptive equalizers, line enhancers or cancellers, etc. One matrix may be limited in size to multiples of 24, or less by filling zeros. This set of data is stored in the coefficient memory (C-memory) prior to beginning the calculations, and can be updated while operations are going on without changing other words or reseting anything. If this array is intended to have indeterminate length, such as in large correlations, then a trigger pulse activates a limited burst of output correlation scores and resets the sums to start over again.

The other matrix *can be, but doesn't have to be,* infinite - ie., coming from a real-time source. Since the heart of this special processor is all digital, the large matrix can come from an ADC, a tape drive, a large data buffer, or any other source of high speed continuous digital data.

The arithmetic units (AU) are configurable (within certain physical limits) by the backplane wiring. Each module can do 24 simultaneous multiplications and double-precision additions on each cycle of a (maximum) 15MHz clock. One output may be delivered for each input in a continuous filter mode, or enabled in a burst in a large correlation mode.

These modules can be cascaded horizontally to process larger vector lengths, and can be stacked vertically by means of multiplexing to achieve greater speed. A 1x3 matrix of AU cards can do a 72 tap transversal filter at 15MHz. A 3x3 matrix will do the same size at 45MHz. Note that the 3x3 matrix mentioned here is derived from the speed of the front end of the system (45MHz) versus the processing speed of the AU cards (15MHz), and from the desired tap length (72) versus the length on each AU board (24). These boards are not set up to do 3x3 kernal matrix operations on images.

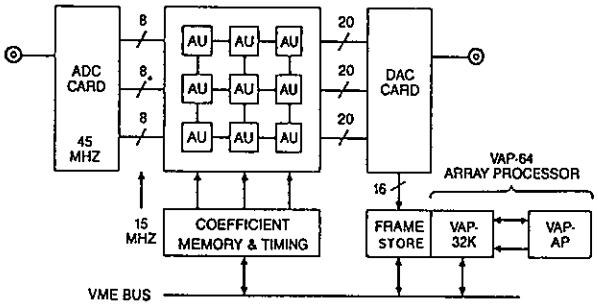
The input data and C-memory data busses to the AUs are all 16 bits two's complement. The initial output from the arithmetic unit(s) (AU) is 19 bits wide with a built-in divide by 2, so there are four bits of headroom above the input precision for the summation of products, to allow word growth without overflow. In certain configurations of multiple AUs, the card outputs will also be summed, so an external 20 bit adder is provided on each AU card for use in the system as necessary.

Final arithmetic results are typically passed to the Reformater and DAC module (DAC). It performs programmable scaling and saturation checking and produces a 16 bit high-speed digital output of the results of the matrix calculations. It also contains a second stage of programmable ranging for the 8-bit digital-to-analog converter and its high-speed analog output.

In most cases, this super-speed processor will be doing filtering, adaptive equalizing, or other continuous real-time tasks. In order to digitally monitor it's progress, a high speed frame store (FS) is also provided, which can capture up to 32K words of high-speed data. Once the data has been sampled, this memory reverts to a dual-port RAM, where one port runs at a syncronous 5MHz and is intended for an array processor such as DSP's VAP family. The other port is asyncronous and available on the VMEbus as normal memory.

The overall system resides in one VMEbus rackmount chassis. All boards conform to the VMEbus form factor and power pin layout. However, not all cards require or accept the VMEbus electrical specifications, since clearly 15 or 45MHz busses are beyond that bus's capabilities. Although several slots of a standard rack might not be used and have standard VMEbus connections, the majority of the backplane wiring is custom for each application configuration, since the AUs and other cards communicate with each other through many separate, short busses. The system can be delivered installed in a tabletop 19 inch rack with fans and power supplies.

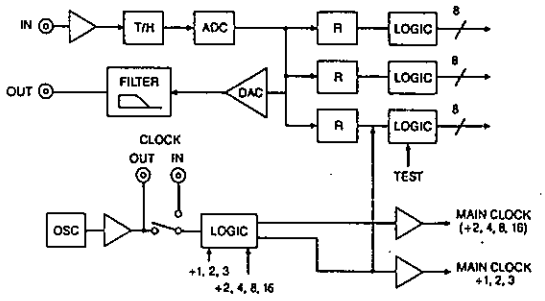
GOPS™ System Diagram



ANALOG TO DIGITAL CONVERTER

The Analog-to-Digital Converter module is the front end analog connection of the GOPS system. The actual ADC section itself is an 8 bit flash converter which can run up to 50 MHz. The input voltage range is +/- .5V, and is internally terminated to 50 ohms. A track-and-hold precedes the digitizer in order to realize 8 bits accuracy at a full 20 MHz, with minimal analog rolloff. The overall analog front end circuitry is flat with less than 3 dB of rolloff at 25 MHz. A DAC is also included, along with a 30 MHz lowpass filter, to monitor the output of the ADC. This module also contains a 90 MHz master oscillator which can be disconnected if the user wishes to utilize an external clock instead. This clock may be divided by 2,4 or 6 depending on jumper selections. It also can be further divided by 2, 4, 8, or 16 under software control.

GOPS™ AD8/50M



Since the arithmetic units currently can only run at 15 MHz, the output of the ADC is split into 3 parallel busses, each running at 15 MHz and being loaded sequentially from the 45 MHz ADC. (If the user configuration only calls for a 30 MHz sample rate, jumpers can change the output scheme to only 2 busses wide, or if 15 MHz is needed, then no demultiplexing is necessary at all.) The final output buffers are programmable logic devices that contain additional testing options. Under user software control, the ADC output can be turned off, and instead either a +.5 DC value or a -1.0 to +.992 ramp can be inserted in place of the analog data. The ADC card dissipates about 15 Watts of power and requires +/- 15V, -5.2V, and the normal +5V TTL supply.

ARITHMETIC UNITS

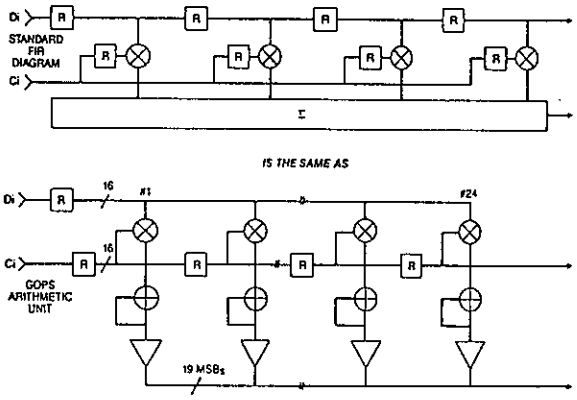
The arithmetic units may contain two sections- the Multiplier-accumulator (MAC) string, as well as an external adder useful for some configurations. Each AU has 24 high speed MACs arranged on a 16 bit parallel input data bus. The input, coefficient and output data formats are all fractional two's complement. Accumulations are performed with 35 bits accuracy, providing 4 bits of growth above the basic 31 bit double precision product. The top 19 bits of the accumulations are returned on the output bus. The output format has the sign and 3 integer bits at the MSB end, and 14 fractional bits. *Thus, there is a builtin divide by 2 in the output format, compared to the input.*

If the filter size desired is larger than 24, or if more than 24 consequtive correlation scores are desired, then more AUs are required. In the simple case of a 15 Msample/sec machine, 48 taps can be realized using 2 AUs. If a higher speed is called for than each individual AU is capable of, then the system must have paralleled AUs. In this case, each is doing a portion of the overall filter equation. At a sample rate of 45 MHz, 3 parallel paths are needed. However, because of the rolling phases of data out of the ADC, we also need 3 horizontal cards to get all of the calculations. Thus, at 45 MHz, the smallest

filter size is 72 taps, and the next size up is 144 taps.

Some AU cards may also contain an extra adder and pipeline registers. The depopulated cards are designated as an AU only. The fully loaded cards are AUAs, (arithmetic unit + adder). Each AU draws about 12 Watts of power when running at speed. The AUAs draw about 16 Watts. Future options may allow depopulating the MACs should filter lengths other than integrals of 24 be desired.

FIR Structure

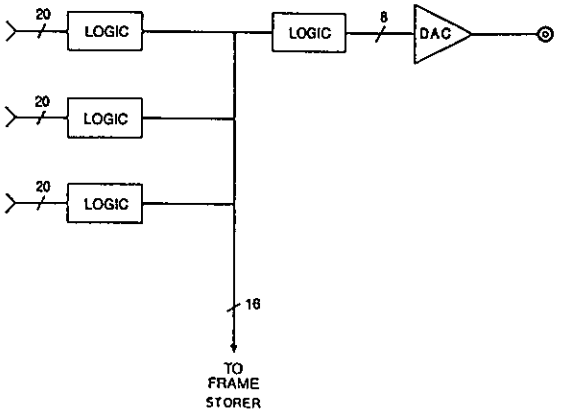


REFORMATTER AND DIGITAL TO ANALOG CONVERTER

This module formats and converts to analog the digital data from the AUs after recombining them into a single digital stream (assuming 2 or 3 wide, multiplexed processing). The data coming out of the AUs is either 19 or 20 bits wide, 5 of which may be above the fraction point. A 4 postion multiplexer allows the user to select under software control the best 16 out of the original 20 bits. If a position lower than the top (/16) is chosen, logic checks the skipped upper bits and will saturate the results to either the maximum positive or negative value, based upon the sign bit. There is also an overflow latch and LED that will set and hold if any overflow occurs, for both software and visual monitoring purposes. Another control line from the control area can reset and disable this flag if desired, however the saturation logic always remains active.

The resulting 16 bits are multiplexed according to the system configuration back to a single bus running at up to 45 MHz. This high-speed bus is buffered out to the Frame Store. It also goes to another 8 to 1 multiplexer. The user can select which 8 out of 16 bits he wishes to send to the Digital to Analog Converter (DAC).

DAC Card

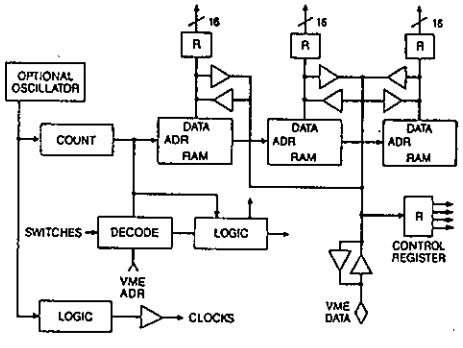


COEFFICIENT MEMORY AND CONTROL

The C-Memory card is the GOPS controller. The user can read and write coefficients to it, and write to a control register, all as normal VME memory. The P2 connector is used to send out the coefficients and control bits to the rest of the GOPS. The module contains a "dual ported" RAM for the coefficients, and a control register. It also has a continuous running counter for reading the RAM and sending coefficients to the input(s) of the AU(s). It can have an oscillator on board along with some clock logic, if there is no GOPS ADC in the system.

There are 3 parallel RAM memories, each 16 bits wide by 2K long for a maximum total of 6K coefficients, and each capable of cycling at 60 nsec. They are addressed by a looping counter whose period is set according to the frame time of the filter. If there is only one AU row, then the frame length is equal to the tap length. When there are multiple rows, the tap length is divided by the number of rows. A 72 tap, 45 MHz filter therefore has a frame length of 24.

GOPS™C-Memory and Control



The Frame Store is a dual-ported memory card that in normal operation looks just like the VAP-32K memory card within the VAP system. It also operates in a high-speed acquisition mode where dual-ported access is temporarily denied while up to 32K 16-bit digital words are loaded from an external source at up to a 50 MHz rate. However, it does have a few minor differences. Instead of having a status register, interrupt logic and various control interface logic, it has two 16 bit counters on board. One is an address counter and the other is a word counter, allowing software control of the acquisition mode.

The Frame Store is intended to work in a VAP system composed of a VAP-AP array processor and a VAP-32K Dual Port Memory board. (The Frame Store may also be configured for stand alone operation from the VMEbus.) In such a system, the Frame Store must be mapped as the lower page (both VME and VAP addresses) while the VAP-32K is mapped as the upper page. The VAP-32K card provides the VAP system with macroinstruction and status ports, and interrupt logic, in addition to the 32K words of memory. The Frame Store provides another 32K words of dual-port memory and 50 MHz digital data acquisition capability. **This allows a user to grab a high-speed block of data, and rather than then move the data to wherever it is to be processed, the data is already resident in the cache memory space of the array processor, ready for immediate computations.**

During a frame store acquisition phase, reads and writes from the VAP and VME ports are ignored. Any reads from the VAP at this time will provide indeterminate data. The array processor can be, of course, continuing to do other processing on the upper half of the memory using the other memory card while the Frame Store is active.

Some degree of software synchronization is necessary where an application allows either the VAP or VME to initiate an acquisition phase at any time. This well known "dead lock" condition can be alleviated by polling or passing permission in software using the Frame Store Busy and VAP Lock flags which are readable in the VAP's status word.

The Frame Store data acquisition inputs are 16 data lines and one clock. The clock is a single, terminated TTL line which can be up to 50 MHz. The input data bus uses the D16 to D31 lines on the P2 connector in the B row. If the user is using a standard VMEbus P2 connector on a system that is capable of long word transfers, then the user must be sure to isolate those 16 lines from the rest of the system. That too can be terminated. The A and C rows of the P2 connector are connected up similar to the VAP-32K card, if the VAP Array Processor is to be used.

[The following information is exerpted from the VAP User's Manual.]

The VAP is a compact and inexpensive fixed-point processor. It consists of a minimum of two cards for the VME bus and is expandable to a third. One of these is a 16-bit fixed-point array processor (AP) that is capable of computation rates of 10 million operations per second. The other contains 32K words of high-speed dual port RAM. The combination of these two board types allows the user to load AP macroinstruction queues and blocks of data into the VAP memory from the VME bus. Meanwhile the AP may simultaneously be executing any one of its powerful signal processing routines.

The library of VAP software routines reside in 64-bit wide microcode PROMS. These routines are initiated by the host computer merely by writing the macroinstruction word pertaining to a particular routine into any portion of VAP memory. For those routines which require additional operands (i.e. word count, transform size, data pointers), these too must accompany the macroinstruction in the required sequence. A series of macroinstructions constitutes a queue which may include signal processing routines such as FFT, FIR FILTER, COMPLEX DEMODULATION as will as program flow instructions like JUMP TO SUBROUTINE, and BLOCK ADD. Execution of a queue is initiated after writing the address pointer of the beginning of the queue. This is followed by writing to the VAPs instruction port (allocated as a separate section of VME memory) notification that a queue is ready. The user has the option of generating an interrupt once the queue has completed its run. Greater detail of the VAP's operation is given in those catagories listed under INTERFACE SPECIFICATIONS in the VAP User's Manual.

If the user wishes to modify existing, or write new, microcode, then the VAP-DS development system is available. It consists of a special extender card that plugs into the chassis in place of the VAP-AP card. The AP card is then plugged into it. Expansion cables are hookuped up to the AP, allowing the VAP-DS to turn off the AP's microcode PROMs and replace them with EEPROM writeable control store. The development system circuitry is connected to an IBM PC which contains the drivers for editing the code in easy-to-learn mnemonics, and for reading the busses of the array processor during execution of the microcode. The system also has multiple breakpoint, single step, and psuedo-run capability in order to clearly see what is going on in the VAP-AP. All of this is transparent to the user's operating system, since the VAP-DS only draws power from the bus. Thus the user does not have to bring down or modify his driver codes in any way when communicating with the VAP.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A MULTI-BAND HEARING-AID EMULATION USING REAL-TIME DIGITAL SIGNAL PROCESSING

João Carlos VENTURA, Lorenzo MORELLINI

Laboratoire d'Electromagnétisme et Acoustique
Ecole Polytechnique Fédérale de Lausanne
Ch. de Bellerive 16 1007-CH LAUSANNE

The subject of this paper is the description of a multiprocessor digital signal processing system and its utilisation in the emulation of a multichannel hearing aid.

1. Introduction

In spite of the recent breakthrough in digital techniques, hearing aids have not yet benefited from these achievements. Therefore studies are being carried out in order to perform hearing rehabilitation by means of digital signal processing by a VLSI chip. The projected functions must be tested for both, their technical realizability and rehabilitation effectiveness (experimentation on the hearing impaired). This leaded to the construction of the SENSA (Système d'Emulation Numérique de Suppléance Auditive), a TMS32010 processor array controlled by a desktop computer.

Since simple hardware functions may require complex algorithms when performed by software, the SENSA is conceived as a multiprocessor system to assure enough throughput for the handling of such algorithms.

This paper describes the design of the SENSA and its test and tool software, as well as its performance of the hearing aid functions.

2. Station configuration

The SENSA is organized according to fig. 1 block diagram. The Processor Units (PU) are controlled by a SMAKY 100, which is a Motorola's 68000 based desktop computer. The SMAKY loads the PU's program memory RAMs by DMA, through a VME interface. The interfaces interrupt controller allows the vectorization of interrupts coming from up to 8 PUs. If no interrupt routines are needed, more PUs can be added to the station, since the SMAKY does not interfere with the Input/Outputs (I/O) between PUs and analog interfaces.

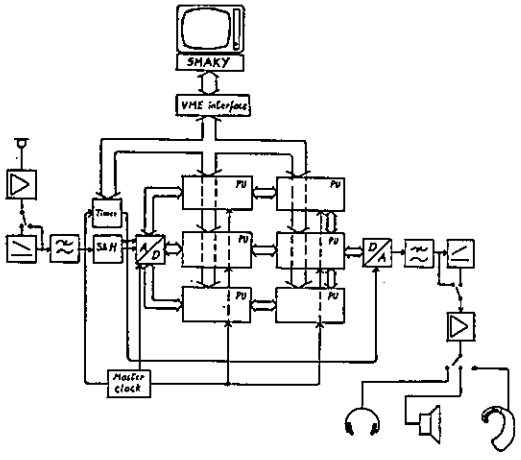


fig. 1

In its present state, the SENSA works with six PUs (this number has been determined by the hearing aid's functions' needs) disposed in a parallel/ pipeline configuration. PUs can interconnect in no matter which pattern, limited only by the number of ports available in each.

12 bit A/D and D/A converters are used for analog interfacing. A programmable timer, controlled by the SMAKY, determines the sampling period (Te) with a 100ns step precision. Sampling rates can be set up to 55kHz.

3. Processor units architecture

PUs' hardware uses the dual-memory principle developed by R. Van Kommer in the LAMI/EPFL using a TMS32010 processor. The TMS is switched to one of the two 4Kword program memory RAMs (RAM0,RAM1) while the SMAKY is switched to the other. Thus, the processors can operate while program/data loadings are occuring (fig. 2).

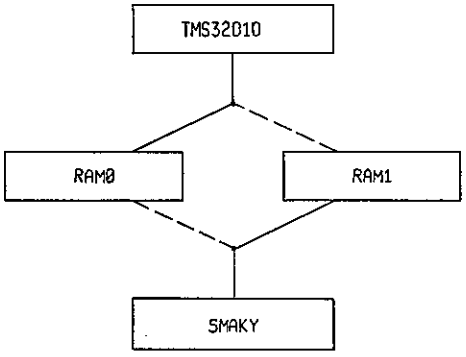


fig.2

The original PU (designed for a single coprocessor unit) was modified in the LEMA, in order to operate synchronized I/O. Four I/O ports are available while a fifth one is used for signaling that data is ready for input. Whenever a sample is outputed from a PU (or A/D converter), it sets up a flag. The following PU can test it as a bit of its fifth port and clears the flag while inputing the new sample.

4. Tool software

Modula-2 was chosen for the SENSA programming.

The determining factors leading to this choice have been its modularity, its code execution speed and the easy access it offers to a given memory location. Furthermore, a high-level language allows an easier handling of floating point numbers, especially when calculating a routinés coefficient, a very useful feature in the calculation of signal processing parameters.

The first step was the definition of software tools for program development:

- Assemblage routines, allowing the creation of an executable code for the TMS32010.
- SENSA's command routines, allowing control over the sampling rate, PU status and memory access.

These control and assemblage routines are gathered in the "COMMANDE" module, in which all TMS32010's instructions are defined.

The user can therefore define in the same procedure the dialogue for a given function and the routines for the PU that will perform it.

5. Structure of the hearing aid

Up to now, hearing aids have yielded an unsufficient effectiveness on rehabilitation. Indeed, the percentage of rejections is high. In some cases prosthesis can even be completely inadequate. A satisfactory fitting of the speech signal to the patient's residual hearing has not been achieved because it demands a control over more sound parameters than the ones presently taken into account. Not only the hearing dynamic range is often reduced (case of recruitment), but also the amount of this reduction is frequency dependant.

The right adaptation must therefore include a compression which characteristic is controllable over frequency. This can be achieved by a multichannel hearing aid as shown in fig.3 block diagram. The speech signal is splitted into 3 bandpass channels by digital filtering. Each channel is amplified and compressed to fit the patient's own residual auditive headroom in that frequency zone. The filter cutoff frequencies must be programmable in order to adapt the channels' bandwidths to the patient's audiogram.

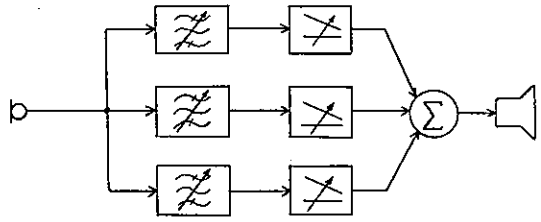


fig.3

Filter selectivity won't necessarily be uniform to all channels: if the audiogram shows a brisk change in hearing loss over frequency, the channel splitting requires a greater selectivity than if the split occurs through a smooth change. Therefore it must be possible to decrease the order of one filter to increase the order of another one. This is achieved by synthesising each filter by the association of biquad cells. Having a certain number of them available, they will be assigned to different filters according to the required selectivitiy needs.

The PU's interconnection pattern is based on the hearing aid block diagram. Each processing element (filter or compressor) is assigned to one PU. This does not mean that each function requires a whole processor's throughput but this particular architecture eases the dialogue with the user (the parameters of each function can be manipulated independently).

6. Filtering routines

For the purposes of this research, it is important to be able to implant filters with any desired response. A way of obtaining it is the use of algorithms as the Remez, which allow synthesis through the attenuation specifications. However this solution is far too complex to be implemented presently in the SENSA. On the other hand, the transposition from analog to digital filters by the bilinear transformation is easily achievable.

We have therefore created a data base file containing the parameters of several analog filters as the cascading of biquad cells. They can be:

- Low-pass Chebychef or Butterworth filters (transposable to high-pass or band-pass)
- any kind of filters, calculated formerly in a VAX computer using the Remez algorithm.

The data base file is used by the module 'FILTRES'. This program handles the dialogue for filter design, the conversion into band-pass or high-pass, displays the amplitude or phase response of the chosen filter, and converts it into its digital equivalent using the bilinear transformation. Since the parameters stored in the data base file are relative to a normalized frequency scale, the digital filter coefficients calculation takes into account the cutoff frequencies and the sampling rate.

The amplitude response of the digital filter can be displayed by means of a program that operates the FFT of a filtered white noise.

A procedure of the module "FILTRES" creates the code for the implementation of the chosen filter. It is optimised in view of a minimum time of calculation and can be considered in two parts:

1) The loading of filter coefficients into the TMS 32010 data memory.
2) The filtering loop code. This one is the repetition of a basic single biquad cell code, in wich the coefficient adresses are changed for each cell. The portions of basic code will be shortened, if allowed by the existence of coefficients equal to zero (low-pass filters). The module 'FILTRES' also allows the implantation of several filters in parallel with respect to condition Te > Calculation time.

At this stage, the module 'COMMANDé can load no matter which processor with the routine corresponding to the chosen filter.

This operation proceeds as follows:

- Sampling rate setting;
- PU enabling;
- Choice of the type of filter;
- Transposition, if needed, to high-pass or band-pass;
- Choice of cutoff frequency(ies);
- Analog to digital transposition;
- Creation of TMS 32010 code for the processing;
- Transfer of the code into the PU's program memory;
- Start PU.

Figure 4 shows a filtering function performed by the SENSA over a white noise input, plotted from a B & K spectrum analyser.

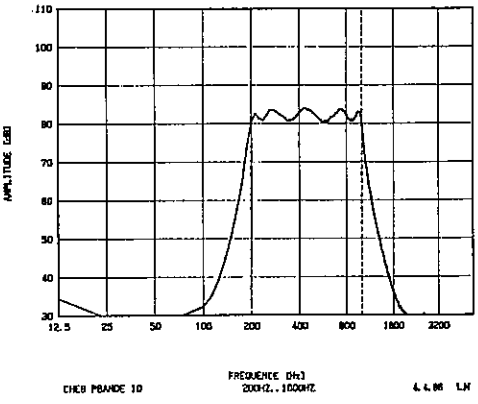


fig.4

The research concerning the compression function is oriented in order to best satisfy the following requirements:

- Full programmability of the compressor static characteristic.
- Adaptation of time constants (i.e. attack time (AT), recovery time (RT)) to speech intelligibility requirements.
- The gain control function is obtained from the input signal value, according to the static characteristic. The input signal detection must therefore represent at its best the ear's intensity perception (choice of RMS detection, peak detection or other).

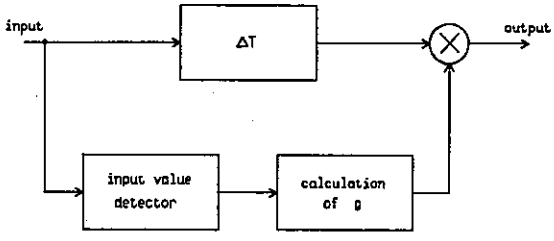


fig.5

7. Compression routines

The compression is performed by the automatic gain control (AGC) function described in fig. 5, where ΔT is a delay applied to the input signal and **g** the gain applied to the delayed signal.

- Avoid, or at least limit, overshoots.

The main parameter concerning the static characteristic of a channel compressor is its compression ratio (R). It is determined by the dynamic range loss in the channel bandpass, relative to a normal hearing audiogram. R is the slope in the compression zone of fig. 6 diagram. Furthermore, the amplification of low signals must be avoided (noise reduction and stability) and high signals must be limited, wich leads to a 3 segment diagram.

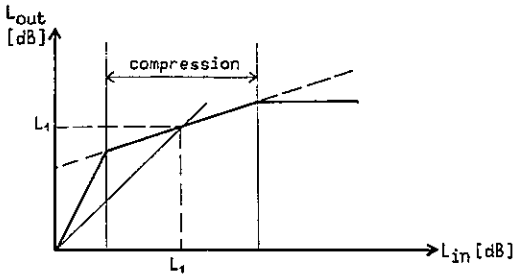


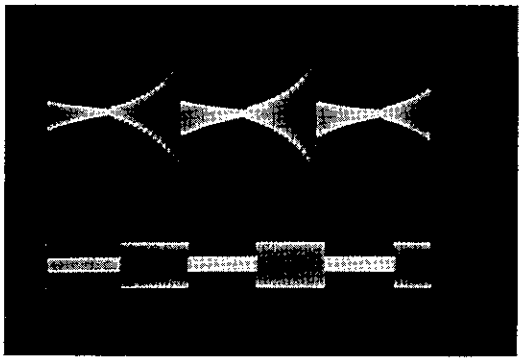
fig.6

The gain can be determined from this characteristic as a function of the input signal value. Defining the input level in the diagram as $L_{in} = 20*\log U_{in}$ and L_1 as the unity gain level, we have in the compression zone:

$$g = (U_{in}/U_1)^{1/2}$$

g is not calculated in the PU. Instead, the 12 bit value obtained by the signal value detector yields the address of the corresponding **g** in program memory. Half of the program memory (2048 words) is loaded with a lookup table of **g**, previously calculated by the SMAKY for a given static characteristic. The address is obtained simply by inverting the sign bit of the detector's output sample.

Picture 1 shows an exemple of compression with gain control by first order RMS detection. The time constant is 2.5 ms, R = 0.2 and $\Delta T = 0$.



picture 1

The present research concerns the optimisation of the input value detection in order to obtain the best adapted gain control function. More results will be presented by the time of this paper's presentation.

Acknowledgements

This research project was made possible by the financial support of the CERS (Commission d'Encouragement à la Recherche Scientifique) and the FNS (Fonds National Suisse).

References:

[1] R. Charlet de Sauvage "Surdité profonde avec recrutement: compression et filtrage multibandes" Bulletin d'Audiophonologie 6-7, Vol. 16, 1983, pp. 719-728

[2] G.W. McNally "Dynamic Range Control of Digital Audio Signals" J. Audio Eng. Soc. Vol. 32, No.5, pp. 316-327 (1984 may)

[3] E.F. Stickvoort "Digital Dynamic Range Compressor for Audio", J. Audio Eng. Soc., Vol. 34, pp. 3-9 (1986 jan./feb.)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CORDIC Realization of a DFT Processor

R. Lerch (*), J. F. Böhme (**), H. Hahn (*), B. J. Hosticka (*),
G. Schmidt (**), D. Timmermann (*) and G. Zimmer (*)

(*) Fraunhofer Institute for Microelectronic Circuits and Systems,
Bismarckstr. 69, D-4100 Duisburg 1, Federal Republic of Germany
(**) Lehrstuhl für Signaltheorie, Ruhr-Universität Bochum,
D-4630 Bochum, Federal Republic of Germany

Abstract

This contribution describes a proposed implementation of a DFT processor based on a modified Bluestein algorithm. It uses CORDIC elements to execute complex multiplications. The processor has been simulated using the digital circuit simulator HILO-2.

Bluestein Algorithm

The impulse response of an N-point sampled-data chirp filter is given by $h(n) = \exp(j\pi n^2/N)$. Assuming an input signal sequence $x(n)$, which is nonzero for $0 \leq n \leq N-1$, we obtain at the output of the chirp filter the signal

$$u(N+k) = \sum_{n=0}^{N-1} x(n) e^{j\pi(N+k-n)^2/N}$$

at a time instant N+k. This yields

$$u(N+k) = e^{j\pi k^2/N} (-1)^N \cdot \sum_{n=0}^{N-1} x(n) e^{j\pi n^2/N} \, e^{-j2\pi nk/N}.$$

It can be seen that if we premultiply an input sequence $x(n)$ by the factor $\exp(-j\pi n^2/N)$, perform the convolution in the chirp filter, and postmultiply the output by $(-1)^N \cdot \exp(-j\pi k^2/N)$, we obtain as a result the DFT of the input sequence [1], i.e.:

$$y(k) = \sum_{n=0}^{N-1} x(n) \, e^{-j2\pi nk/N}.$$

Chirp Filter

The chirp filter can be implemented using a transversal structure but the amount of digital hardware does not make this method particulary attractive. Bluestein himself proposed a recursive chirp filter for the case of $N = m^2$. As the z-transform of the chirp filter transfer function is

$$H(z) = \sum_{n=0}^{2N-1} e^{j\pi n^2/N} \, z^{-n},$$

he introduced the substitution $n = t+im$, where $0 \leq t \leq m-1$ and $0 \leq i \leq 2m-1$, and obtained a two-dimensional mapping, namely:

$$H(z) = \sum_{t=0}^{m-1} z^{-t} e^{j\pi t^2/N} \sum_{i=0}^{2m-1} (-e^{j2\pi t/m} z^{-m})^i.$$

This can be realized by a filter bank consisting of m filters in parallel. In this work we will use another implementation of the filter bank. To start with, consider the t-th filter of the filter bank that carries out the inner convolution. The partial sum at the output is defined by

$$u_t(N+k) = \sum_{n=0}^{N-1} (N+k-n) \, h_t(n)$$

at a time instant N+k, where $0 \leq k \leq m-1$. This yields for $k < t$

$$u_t(N+k)=e^{j\pi t^2/N} \star \sum_{i=0}^{m-1} e^{j2\pi ti/m}(-1)^i(N+k-im-t)$$

using the above substitution for n. If we introduce f = m-i, we can write

$$u_t(N+k)=e^{j\pi t^2/N} \star \sum_{f=1}^{m} e^{-j2\pi tf/m}(-1)^{m-f}(k+mf-t).$$

For $k \geq t$, we obtain

$$u_t(N+k)=e^{j\pi t^2/N} \star \sum_{f=0}^{m-1} e^{-j2\pi tf/m}(-1)^{m-f}(k+mf-t).$$

Similarly, at a time instant N+k+gm the convolution is given by

$$u_t(N+k+gm)=e^{j\pi t^2/N} \star \sum_{i=0}^{m+g-1} e^{j2\pi ti/m}(-1)^i(N+k+(g-i)m-t)$$

for $0 \leq g \leq m-1$. The latter expression can be rewritten for $k<t$ (but the same result is obtained for $k \geq t$) as

$$u_t(N+k+gm)=e^{j\pi t^2/N} \star \sum_{i=g}^{m+g-1} e^{j2\pi ti/m}(-1)^i(N+k+(g-i)m-t)$$

owing to the finite length of the input sequence as stated above. Thus, we conclude that

$$u_t(N+k+gm) = e^{j2\pi tg/m}(-1)^g u_t(N+k).$$

This suggests an efficient realization of the chirp filter. The input samples are multiplied by $e^{j\pi t(t-2f)/m}(-1)^{m-f}$ where f is being incremented after every m-th cycle. Also every m-th cycle the intermediate result are accumulated and saved. These sums must then be multiplied by $e^{j2\pi tg/m}(-1)^g$, where g is also being incremented every m-th cycle. Hence two complex muliplications are necessary for each individual filter that is a part of the chirp filter. If we wish to perform the DFT afer Bluestein we can merge conveniently these operations with the pre- and postmultiplications mentioned above and create a single module. The total premultiplication is then given by $\exp(j\pi[t(t-2f)/m+(m-f)-n^2/N])$ and the postmultiplication is $\exp(j\pi[2+g/m+(N+g)-k^2/N])$ for a t-th module.We shall designate the corresponding rotation angles as ϕ_1, and ϕ_2, respectively.

CORDIC Implementation

The complex multiplications can be readily executed using the CORDIC arithmetic processor [2]. This processor can perform vector operations in circular, linear, and hyperbolic coordinate systems by iterations [3]. Its hardware implementation requires only adders, shifters, registers, and comparators. For our complex multiplications we need only vector rotations in the circular system. The result of the CORDIC rotation can be described as: $X = x\cos\phi - y\sin\phi$ and $Y = x\sin\phi + y\cos\phi$, where x, y and X, Y are the coordinates before and after the rotation, respectively, and ϕ is the angle of rotation (Fig.1). Obviously, the CORDIC element can carry out a complex multiplication, if we use two data paths to represent the real and imaginary component. To adapt the CORDIC for our pre- and postmultiplication to be utilized in

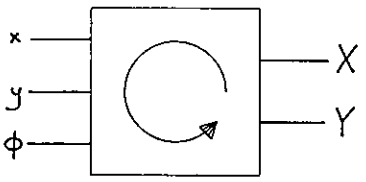


Fig. 1 CORDIC element

our DFT processor, we have to produce the required rotation angles ϕ_1 and ϕ_2. These have been derived in the previous section and their hardware generation (incl. the squared terms) is illustrated in Fig. 2. The constants are as follows: a = -1/N, b = -(1+2t/m) and c = $m+t^2/m$ for the premultiplication (ϕ_1) and a = -1/N, b = 1+2t/m and c = N for the postmultiplication (ϕ_2) in the t-th module.

Fig. 3 shows the circuit that serves as a t-th module in the DFT processor. The processor consists of m pipelined modules which are all identical. In each module the first CORDIC carries out the total premultiplication. This is followed by an accumulator to generate the intermediate sums. The accumulator must be initialized each N-th cycle and the data must be correspondingly delayed to ensure proper sequencing. The data are then added at the output of the postmultiplication stage to the output stream of other modules to yield the complex Fourier coefficients y(k).

Simulations

We have simulated the complete DFT processor using the digital circuit simulator HILO-2 [4]. The results of the simulation appear in Fig. 4. The input signal is shown in Fig. 4a and it represents a frequency mixture

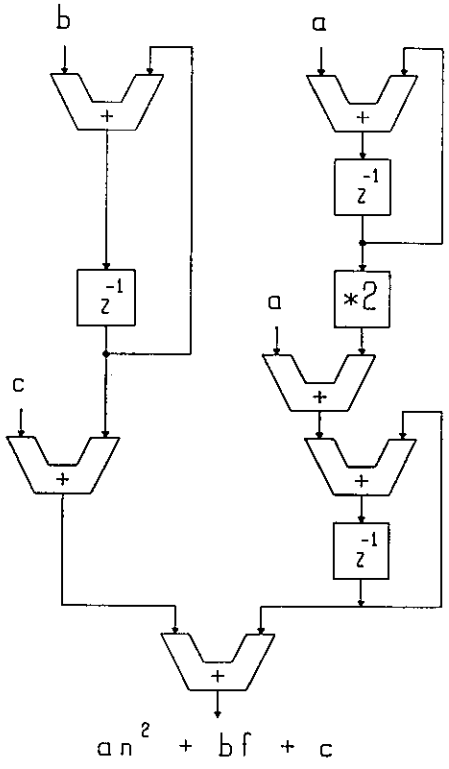


Fig. 2 Generator of rotation angles

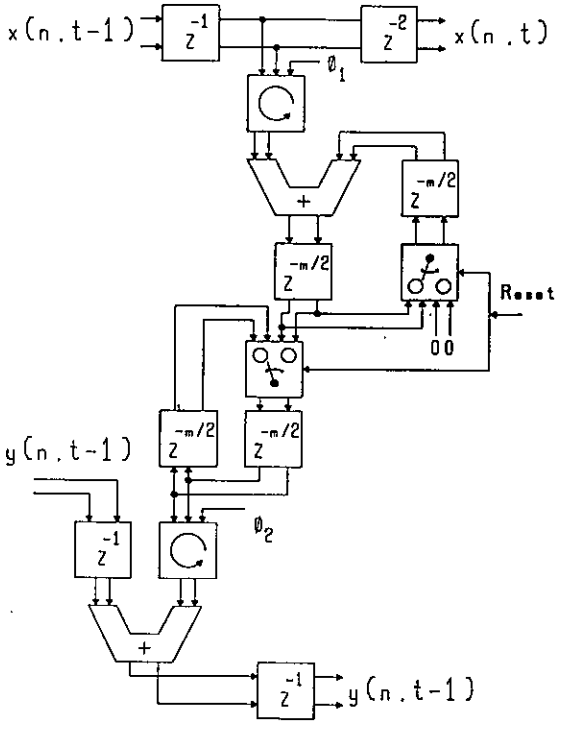


Fig. 3 Module of the DFT Processor

$$x(n) = -20 + 15\sin(\omega_t n) + 50\cos(3\omega_t n) - 160\sin(7\omega_t n) - 125\cos(15\omega_t n) + 45\sin(31\omega_t n) + 30\cos(63\omega_t n) - 10\cos(127\omega_t n)/512,$$

where $\omega_t = 2\pi f_t/256$ and f_t is the sampling frequency.

The magnitudes of the discrete Fourier coefficients obtained at the output of the DFT processor are plotted in Fig. 4b for positive frequencies.

Summary

We have proposed a processor which performs complex DFT and is based on a modified Bluestein algorithm. The required pre- and postmultiplication operations have been merged into the chirp filter. Despite the recursive form of the chirp filter, the CORDIC's are not inside the loop, and can be therefore pipelined. The processor uses CORDIC elements besides registers and adders. For N-point transform, the method requires N to be a perfect square. The total number of CORDIC elements is then given by $2*N^{1/2}$. The pipelined processor structure and strict use of local communications are implying high throughput rates. Because the CORDICs are not used in recursive loops we can employ the pipelining inside the CORDIC as well [5]. Furthermore, the presented concept offers the advantage of continuous operation.

References

(1) L.I. Bluestein, "A linear filtering approach to the computation of the discrete Fourier transform," IEEE Trans. Audio Electroacoust.,vol. AV-18,pp. 451-455, 1970.

(2) J.E. Volder, "The CORDIC trigonometric computing technique", IRE Trans. Electronic Comp., vol. EC - 8, no. 3, pp. 330 / 334, Sept. 1959.

(3) J.S. Walther, "A unified algorithm for elementary functions", Proc. Joint Spring Comput. Conf., pp. 379 - 385, 1971.

(4) Genrad Inc., "HILO-2 User Manual", Santa Clara, 1984.

(5) R. Udo, E. Deprettere and P. Dewilde, "On the implementation of orthogonal and orthogonalizing filters using pipelined CORDIC architectures", Proc. EUSIPCO-83, pp. 847 - 850.

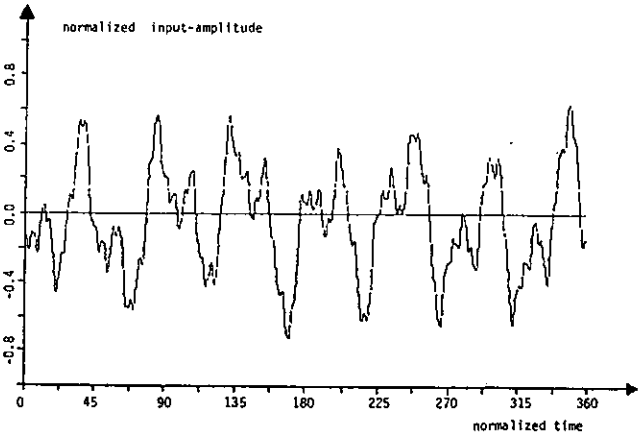


Fig. 4a Input signal in time domain

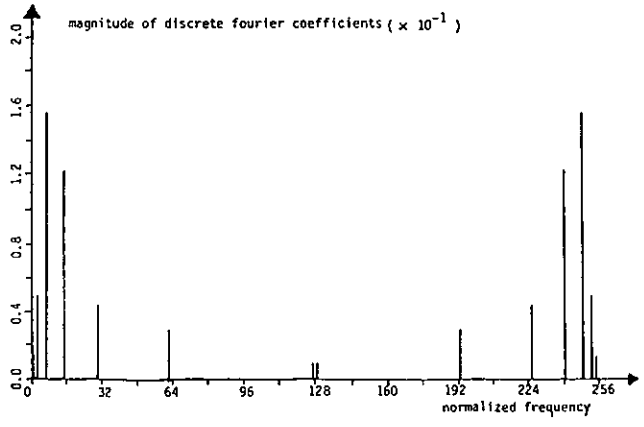


Fig. 4b Output 256-point DFT

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

On Algorithms and Architecture Suitable for Digital Signal Processing

Lars Wanhammar
Department of Electrical Engineering
Linköping University
S-581 83Linköping, Sweden

In this paper we discuss some relations between digital signal processing algorithms and architectures with multiple processing elements, MPE. Particularly we discuss high performance implementation schemes utilizing the parallelism inherent in the algorithm. Fundamental problems regarding balance between computational capacity and communication in MPE architectures are highlighted as well as possible tradeoffs exploring specific properties of the algorithms. Finally, an efficient MPE architecture which have perfect balance between computational capacity and communication is presented.

1. INTRODUCTION

Digital signal processing, DSP, has due to the development of robust and efficient algorithms and advances in VLSI technology, received widespread use. The major reason for this is the relative independence of element sensitivity, which is achieved in digital systems contrary to in analogue systems. The event of DSP processors have further increased the competitiveness of DSP over analogue techniques. However, there are still a large number of applications, for which DSP hitherto is not applicable. One such domain is low performance and low cost systems for which instead SC-techniques is appropriate. Another class of applications is in systems with large and complex computational load. The main reason for this is the general lack of systematic procedures for merging requirements of the algorithm and appropriate circuit architecture.

In this paper we principally restrict the discussion to the problem of implementing high performance/high speed DSP tasks in one or a few dedicated VLSI circuits using low power and small chip area. However, many results are more general.

2. DSP ALGORITHMS

The class of DSP algorithms of concern are the set of expressions, given below, which are to be evaluated recursively. The expressions are given in computational order. It is assumed that pipelining is used in the

$$x_1(n) := f_1[\ -\ , x_1(n-1), x_1(n-2),..., x_p(n-1), x_p(n-2),..., a_1, b_1, c_1,...\ ]$$
$$x_2(n) := f_2[\ -\ , x_1(n-1), x_1(n-2),..., x_p(n-1), x_p(n-2),..., a_2, b_2, c_2,...\ ]$$
$$x_3(n) := f_3[\ x_1(n), x_1(n-1), x_1(n-2),..., x_p(n-1), x_p(n-2),..., a_3, b_3, c_3,...\ ]$$
$$\vdots$$
$$x_N(n) := f_N[\ x_1(n), x_1(n-1), x_1(n-2),..., x_p(n-1), x_p(n-2),..., a_N, b_N, c_N,...\ ]$$

processors, and the communication channels. In some rare applications a loss in efficiency may be incurred. Since, the first value, $x_1(n)$, is in the pipeline and therefore not available when $x_2(n)$ is to be evaluated. This is not a problem in multiplexed systems. However, a penalty in terms of task latency is always incurred.

We assume that the DSP algorithm consists from a computational point of view of an unambiguously specified sequence of operations to be repeatedly performed within a given time interval. Furthermore, it is assumed that all operations can be planned [1, 2] ahead of execution. However, this is not a severe

limitation in most DSP applications. The algorithm is also assumed to have a high degree of parallelism. The operations themself consists of a well defined set of basic arithmetic and logic operations. These operations can again be considered as algorithms acting on machine numbers [9]. The complete DSP task is to be performed within a given time limit using a sufficient number of PE's.

Of main concern are the suboptimal case, where minimal chip area, A_{PE}, is occupied by the PE's. The chip area needed for the PE's can be minimized by making a number of design tradeoffs as discussed below.

Given a set of basic operations, the true inherent parallelism in the algorithm can be determined by a critical path analysis performed over multiple sample intervals [2]. It is not correct to only consider one sample interval. The inherent parallelism in the algorithm is important, because it limits the usable number of PE's as well as the chip area.

Several different types of PE's are needed in most algorithms, e.g., adders, multipliers, quantizers. By proper scheduling, using different mixes of PE types, with associate chip areas, a number of feasible points in the area-time domain can be determined, as shown in Fig. 1.

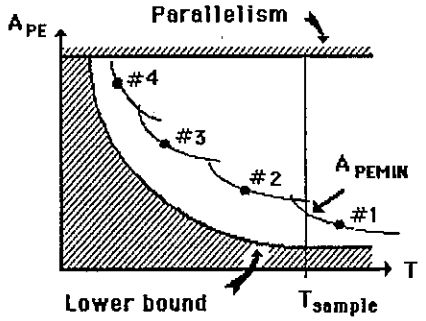


Fig. 1. Chip area versus execution time for different mixes of PE's.

Further, a tradeoff between speed and chip area can be made by optimizing the logic circuits realizing the PE's. Thus, the requirement on the sample rate, in Fig. 1, can be met by using a faster circuit with a slight increase in chip area. In principle, a lower bound on the chip area can be imagined using optimal scheduling and circuit realizations.

A class of interrelated algorithms with different computational properties can be obtained by changing the set of basic operations, e.g., using combined multiplier-accumulators instead of separate adders and multipliers. Thus, different area-time tradeoffs are possible. However, the numerical properties of these algorithms may differ as well as the suitability for implementation using a given architecture.

3. ARCHITECTURES FOR DSP

To any given DSP algorithm of the type discussed above, corresponds a class of ideal MPE architectures to be defined below.

A processing element, PE, shall perform a mapping of an input data set to an output set. Typical processing elements in a VLSI circuit are: adders, ALU's, multipliers and quantizers.

The memory elements, shall store data in such a way that the PE's, with due respect to the algorithm, can access appropriate data without loss of any computational time slots. Since the PE's normally needs several inputs simultaneously, we assume that the memories are partitioned into several independent memories or have several ports.

The interconnection network, IN, shall provide necessary communication channels between the PE's and store elements so that no computational time slots are lost.

In an ideal DSP architecture, the PE's shall be supplied with data without loss of any computational time slots with respect to the inherent parallelism in the algorithm. All computations shall in average be performed in a shorter time then the sample interval using a minimal number of PE's.

The types and number of PE's are determined from the application requirements. There are a number of different interconnection networks and memory organizations to chose from. Fig. 2 and Fig. 3 show the two generic MPE architectures. The major limitation of the

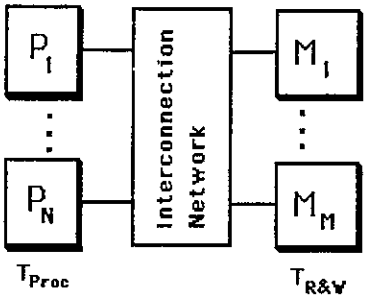


Fig. 2 Shared storage architecture.

shared storage architecture is the limited memory bandwidth. The message-based architecture uses instead direct interprocessor communication and each processing element has a private memory.

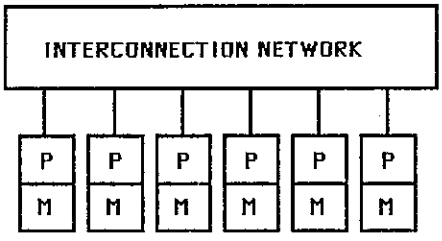


Fig. 3 Message-based architecture.

The shared storage architecture is more versatile, but it may be more expensive for a large number of PE-memory-pairs. It is suitable for tightly coupled algorithms, i.e., algorithms which contain both local and global data dependencies. However, due to the memory bandwidth bottleneck, the shared storage architecture is suitable only for applications where only a few processors cooperate synchroneously.

Each PE must be allocated two memory time slots for receiving inputs and storing the output value. Thus, in order to fully utilize N_p PE's we must have:

$$T_{proc} \geq 2N_p T_M \qquad (1)$$

where T_{proc} = processing time for one data item, $T_M = T_R = T_W$ are the read and write time of the memories. However, T_M and T_{proc} are usually of the same order. Hence, there is a fundamental imbalance between computational capacity and communication bandwidth. In the next sections we discuss methods to counteract this imbalance.

3.1 REDUCING T_M

The memory speed can be increased by using interleaving techniques, i.e., several memory banks. This causes expensive overhead because of the necessary duplicating of decoders, sense amplifiers, etc. Furthermore, losses in computational efficiency can be caused by memory conflicts. Fast cache memories private to each PE may in principle be used.

3.2 REDUCING THE COMMUNICATION DEMAND

There are numerous schemes to reduce the number of memory accesses. We will only mention the most common ones [9].

3.2.1 Using common data and parameters

The number of memory read cycles can be reduced in algorithms where all PE's operate the same data, e.g., array processors [3,7]. In such cases perfect balance between processor speed and communication bandwidth can be obtained. However, this solution often yield a too large computational capacity. The hardware is suitable for time-sharing between many input channels. Other favourable properties are that dedicated processors can be used and that the interconnection network is very simple and fixed.

3.2.2 Directly connected processing element

Direct interprocessor communication via latches introduces pipelining. Popular exponents of this approach are systolic arrays, which both exploit pipelining and parallelism in the algorithm.

3.2.3 Partitioning of the DSP task

The most common approaches in MPE architecture involves partitioning of the algorithm into parts with small intercommunication needs and to distribute fast private memories to each PE, e.g., using conventional processors.

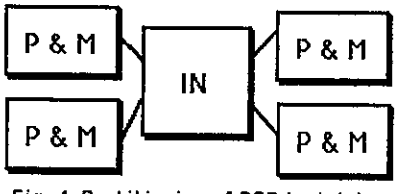


Fig. 4 Partitioning of DSP task into low communicating parts.

Numerous interconnection networks have been proposed, e.g, linear array, 2-dimensional mesh, boolean cubes, binary tree, and schuffle exchange [9].

3.3 USING SLOW PROCESSORS

Better balance can be obtained by increasing the execution time of the PE's. This can be done in at least two different ways.

The first method is to change the algorithm by increasing the granularity of the basic operations so that more complex and hence time consuming basic operations are performed without explicitly using the memories. These changes also tend to reduce the number of memory accesses.

The second method, and seemingly the most favourable one, is to make a tradeoff between chip area and execution time for the PE's, observing that the work performed by a processor is proportional to AT_{proc}. Thus, a perfect balance can be obtained by increasing T_{proc} and simultaneously reducing the chip area, keeping $AT_{proc} \approx$ constant (or improved) [2].

4. A MPE ARCHITECTURE WITH PERFECT BALANCE

Numerous circuit realizations of the PE's are possible with constant AT_{proc}. Moreover, in order to support many PE's, it is necessary that $T_{proc} \gg T_M$.

A practical choice is to use bit-serial arithmetic in the processors, with large T_{proc}, and in the interconnection network. The memories maintain the high speed, relatively, by using an appropriate serial-parallel interface, as shown in Fig. 5. The MPE architecture has perfect balance between processing capacity and communication bandwidth.

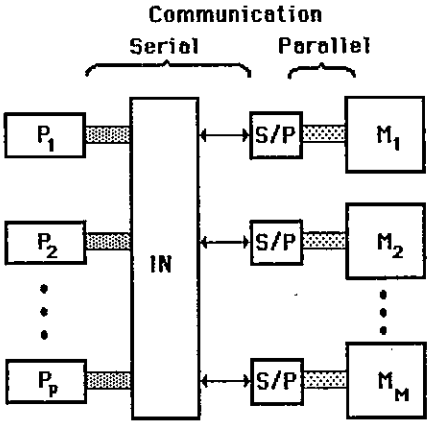


Fig. 5 MPE architecture with communication balance.

The maximal possible number of processors in the system shown in Fig. 5 is limited by the parallelism in the algorithm or by the data word length. However, several such systems can be interconnected as shown in Fig. 4. The necessary number of PE's, N_p, depends on the chosen set of basic operations and computational load of the DSP job.

The necessary communication bandwidth between processors and memories leads to the following inequality:

$$N_p \leq \left\lceil \frac{T_{proc}}{T_M} \right\rceil \frac{W_M}{2} \qquad (2)$$

where W_M is the word smallest length in any of the RAM's. The memory word length, W_M, is chosen to an integer mutiple of the data word length, W_d, are for practical reasons and in order to avoid access conflicts.

$$W_M = n\, W_d\,, \quad n = \text{integer} \qquad (3)$$

In principle it is possible to use different data word length as well as use multiple word data length in different PE's and operations.

ACKNOWLEDGMENTS

The author wishes to thank Tekn. D. Björn Sikström, Ph. D. Morteza Afghahi and Civ.Ing. Jack Pencz for stimulating discussions.

REFERENCES

[1] Afghahi M., Pencz J., Sikström B., Sjöström U., Wanhammar L.: "Towards a General Approach to Implement Digital Filters," Nordic Workshop on Digital Signal Processing, Lejondals Slott, Stockholm, Oct. 3-5, 1985.

[2] Afghahi M., Pencz J., Sikström B., Wanhammar L.: "On Parallelism in DSP Algorithms," To be published.

[3] Afghahi M., Matsumura S., Pencz J., Sikström B., Sjöström U., Wanhammar L.: "An Array Processor for 2-D Discrete Cosine Transforms," Proc. European Signal Processing Conf., EUSIPCO-86, The Hague, The Netherlands, Sept. 1986.

[4] Denyer P., Renshaw D.: *VLSI Signal Processing: A Bit-Serial Approach*, Addison-Wesley Publ. C., 1985.

[5] Dinha F., Sikström B., Sjöström U., Wanhammar L.: "A Multi-Processor Approach to Implement Digital Filters," Nordic Conf. on VLSI in Computer and Communications, pp. 114-119, Tampere, Finland, June 13-15, 1984.

[6] Dinha F., Sikström B., Sjöström U., Wanhammar L.: "LSI Implementation of Digital Filters - A Multi-Processor Approach," Intern. Conf. on Computers, Systems & Signal Processing, Bangalore, India, Dec. 10-12, 1984.

[7] Matsumura S., Sikström B., Sjöström U., Wanhammar L.: "LSI Implementation of An 8 Point Discrete Cosine Transform", Intern. Conf. on Computers, Systems & Signal Processing, Bangalore, India, Dec. 10-12, 1984.

[8] Sikström B.: "On the LSI Implementation of Wave Digital Filters and Discrete Cosine Transforms," Linköping Studies in Science and Technology, Diss. no. 143, Linköping University, Sweden, May 1986.

[9] Wanhammar L.: "Algorithms and Architecture Suitable for Digital Signal Processing," Nordic Workshop on Digital Signal Processing, Lejondals Slott, Stockholm, Oct. 3-5, 1985. LiTH-ISY-I-0787.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

ONE AMPLIFIER APPROACH TO A RATIO-INDEPENDENT CYCLIC A/D CONVERTER

Keping Chen and Sven Eriksson
Department of Electrical Engineering
Linköping University
S-581 83 Linköping, Sweden

New circuit configurations of high resolution, low power consumption and small die area analog-to-digital (A/D) converters using a parasitic-insensitive switched capacitor realization are described. The proposed cyclic A/D converter consists of only one operational amplifier. The linearity of the converter is independent of the capacitor ratio mismatch. The novel ratio independent pipelined converter using one operational amplifier per stage is also presented.

1. Introduction

There are three basic ways to convert an analog signal to a digital signal. Serial converters represent the linear search methods. The drawback is that for each sample 2^N clock cycles are required, where N is the number of digital bits. Flash converters represent parallel search methods. The conversion rate is very high. Unfortunately, 2^{N-1} levels of reference voltages and 2^{N-1} comparators have to be implemented. This leads to a large chip area and a large amount of power consumption.

The successive approximation performs a binary search. For an N bit conversion, N clock cycles are required. In many applications, the successive approximation is a good compromise between speed and chip area. The successive approximation conversion methods can be divided into two groups. One group of converters is constructed with a build-in D/A converter and a successive approximation register. This group of converters is commonly called successive approximation A/D converters. Another group of converters makes self signal scaling after each step of successive approximation. This type of converters is called algorithmic A/D converters. They realize an algorithm:

$$D_i = \text{Sign}\ (V_i)$$
$$V_{i+1} = 2V_i - D_i\ V_{ref},\quad i = 1, \ldots, N$$

where $V_I = V_{in}$ is the input analog voltage. N is the number of digital bits. The function Sign (·) performs a polarity check of an analog voltage. When V_i is positive, D_i is given a value of +1, otherwise, D_i is given a value of -1.

This algorithm can be implemented using a recursive (cyclic) structure or a pipelined structure. The cyclic A/D converters have minimal chip area, while the pipelined structures give a maximal conversion speed.

The cyclic A/D converter, realized as an SC circuit [1], is known for its circuit simplicity, small die area, and low power consumption. The major non-linearity error, caused by the capacitance ratio mismatch, can be eliminated by using ratio independent structures, [2-3]. Typically, three operational amplifiers are used. One is used for the purpose to obtain multiplication by two and subtraction or addition of a reference voltage. One works as a sample-and-hold amplifier, and the remaining as a comparator in order to extract the sign bits.

SC realization of a pipelined A/D converter has been successfully implemented on silicon by Masuda and et al. [5]. This realization utilizes two op amps per stage and is capacitor ratio dependent. The resolution is limited to 6-8 bits.

Since op amps consume power, occupy most of the A/D converter's die area, and are sources of noise, it is worth considering techniques that offer the

possibility to reduce the required number of op amps. A straightforward technique for the reduction of the number of op amps is to use time-sharing or multiplexing for the amplifiers. It is desirable that the number of clock phases and good properties, such as parasitic-insensitivity, are retained.

2. Ratio independent SC realization of A/D converters

In the SC circuit, the accuracy of the charge transfer mainly depends on the capacitor ratios. To obtain a capacitor ratio independent multiplication-by-two, the sampling capacitor has to sample twice from the analog buffer. The first sampled charge is transferred to the integrating capacitor. After the second sampling, the sampling capacitor will be put in the position of the integrating capacitor and the charge in the integrating capacitor will be retransferred to the sampling capacitor. As a result, the output voltage of the operational amplifier will be doubled.

3. Circuit descriptions

The circuit diagram of the proposed cyclic A/D converter is shown in Fig. 1. A fully differential op amp is to be used for the realization of the A/D converter. This would compensate for the common mode offset, reduce the clock feedthrough, etc. However, for the sake of simplicity, the circuit with a single output op amp is used in this paper just in order to illustrate the principle of the operation of the A/D converter. Only one op amp is needed for the complete A/D converter. The sample-and-hold function is performed via the capacitor C_h. The sampling capacitor C_s together with the op amp in the open loop mode works as a comparator, C_i is an integrating capacitor; it works purely as an intermediate storage.

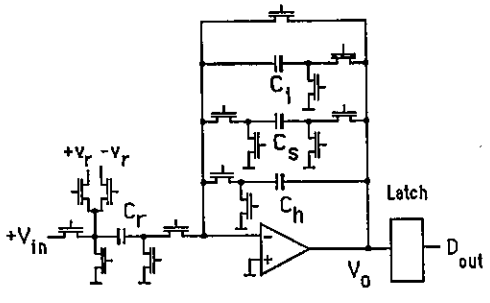


Fig. 1 SC realization of ratio independent cyclic A/D converter

Let us now examine how the output voltage can be doubled independently of the capacitor ratios. To start with, suppose that C_h is in the buffer mode. The output voltage now is assumed to be V_1. The bottom plate of the capacitor C_h is connected to the output of the op amp and its top plate is connected to the inverting input of the op amp. C_s is in the sampling mode, its bottom plate is connected to the output of the op amp, and its top plate is connected to the ground. During the next phase, C_h is disconnected from the op amp and C_i is in the integrating mode, the charge is transferred from C_s to C_i. The output voltage now becomes:

$$V_0 = \frac{C_s}{C_i} V_1$$

During the following phase, C_i is disconnected from the op amp, and C_h is in the buffer mode again, the output of the op amp is still V_1. C_s is in the sampling mode. In the next phase, the capacitor C_s is in the integrating mode, that is, its bottom plate is connected to the output of the op amp and its top plate is connected to the inverting input of the op amp. The bottom plate of C_i is connected to the ground. The charge stored in C_i will be transferred to C_s. Then, the output voltage is doubled.

$$V_0 = V_1 + \frac{C_s}{C_i}\frac{C_i}{C_s} V_1 = 2V_1$$

During the same phase, C_h is in the sampling mode to prepare for the next cycle. The polarity check of the output voltage is performed before the next cycle operation starts. During this phase of sign bit extraction, the top plate of C_s is connected to the inverting input of the op amp, and the bottom plate is connected to ground. The other capacitors are disconnected from the op amp. The output of the comparator forms one digital bit, which will also be used as the control signal to select addition or subtraction of the reference voltage. This operation is performed via capacitor C_r during the phase when the voltage is doubled.

The speed of the cyclic A/D converter can be improved using pipelined structures. The ratio independent cyclic A/D converters, proposed in the literature, [2-4], and in this paper, are not suitable to be pipelined. A possible building block for the

ratio independent pipelined A/D converter is shown in Fig. 2. The building block contains only two capacitors. C_s will be used as a sampling capacitor, as a sample-and-hold circuit for the next stage, and used as a capacitor in the offset free comparator.

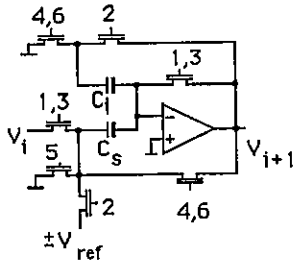


Fig. 2 A building block for ratio-independent pipelined A/D converter

The circuit requires six clock phases. The clock scheme is indicated by the numbers at the switches; for example, a switch together with the numbers 4,6 means that the switch is on during clock phase 4 and 6. The clock scheme for its adjacent stage will be shifted 3 phases from this stage. That is, the switch controlled by clock phase 5 in this stage will be controlled by clock phase 2 in the next stage, etc. Suppose that the input voltage signal is held during phase 1 and phase 3. C_s samples the input twice during phase 1 and 3. The first sampled value is temporarily shifted to C_i during phase 2. The positive or negative reference voltage is also added via C_s depending on the sign of the input voltage V_i. During phase 4, C_s is used as an integrating capacitor. The charge stored in C_i is transferred back to C_s. The output voltage of the op amp now becomes:

$$V_{i+1} = V_i + \frac{C_i}{C_s}\frac{C_s}{C_i}[V_i - \text{sign}(V_i)V_{ref}]$$

$$= 2V_i - \text{sign}(V_i)V_{ref}$$

This is desirable for each block to perform. The output voltage is held during phase 4 and phase 6 for the next stage. The sign of the output voltage is checked during phase 5, resulting in one digital bit.

4. Circuit evaluation and performance estimation.

The structures of our A/D realization have several important advantages:

a) Since only one op amp is needed, the circuit can be made with a small die area, low power consumption, and low noise.

b) Due to the use of the ratio independent realization for the cyclic and pipelined A/D converters, the conversion linearity is improved.

c) In our realization, the subtraction or addition of the reference voltages is carried out before or during the multiplication-by-two amplification, which would increase the dynamic range with 6 dB.

d) The use of a fully differential circuit in the A/D converter has contributed to the increase of the dynamic range, the reduction of power supply feed-through, cancellation of offset voltage, and clock feed-through rejection.

The drawback is that five or six clock phases are required. However, all ratio independent SC realizations need multi-phase clock systems. That is what one has to pay for increasing the resolution of the converters.

The computer simulation shows that the conversion linearity is insensitive to the capacitor ratio mismatch. Bread board couplings have been experimentally utilized in order to show the feasibility of the constructions. Computer simulations show that it is possible to make an A/D converter with a resolution of around 12 bits at a moderate sampling frequency.

Acknowledgment

This work was supported by the Swedish National Board for Technical Development.

REFERENCES

[1] R. H. McCharles, V. A. Saletore, W. C. Black, Jr., and D. A. Hodges: "An Algorithmic Analog-to-Digital Converter", IEEE Int. Solid-State Circuits Conf., Digest of Tech. Papers, 1977, pp. 96-97.

[2] P. W. Li, M. J. Chin, P. R. Gray, and R. Castello. "A Ratio-Independent Algorithmic Analog-to-Digital Conversion Technique", IEEE Journal of Solid-State Circuits, Vol. SC-19, 1984, pp. 828-836.

[3] C. C. Lee, "A New Switched Capacitor Realization for the Cyclic Analog-to-Digital Converter", in Dig. tech. papers, 1983 Int. Symp. Circuit and Systems, Newport Beach, CA, May 1983.

[4] K.Watanable and H. Matsumoto: "Switched-Capacitor Algorithmic Digital-to-Analog and Analog-to-Digital Converters", Proceedings of IEEE ISCAS'85, Kyoto Japan, June 1985, pp. 331-332.

[5] Shinji Masuda, Yoshishige Kitamura, Shuichi Ohya and Masanori Kikuchi: " A CMOS Pipeline Algorithmic A/D Converter", Proceeding of the IEEE 1984 International Solid State Conf., pp. 559-562.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A 20-BIT VLSI ARITHMETIC UNIT FOR DIGITAL SIGNAL PROCESSING IN THE LOGARITHMIC NUMBER SYSTEM

Fred J. Taylor
Department of Electrical Engineering
University of Florida
Gainesville, FL 32611

ABSTRACT

In this paper, the architecture performance of a 20-bit Logarithmic number system, or **LNS**, processor is reported. The processor is shown to compare well, if not outperform, existing floating point (FLP) processors of equivalent range and precision. The speed-power product ratio of an equivalent FLP processor, compared to the LNS processor, is reported to be twenty to one in the case of the square and square root operations and one to one in the case of addition and subtraction. For multiply and divide, this ratio is about five to one. Finally, these new computational engines are applied to a host of DSP problems and found to be an exciting alternative to slower and more complex traditional designs.

I. Introduction

As Barth noted, "there are two types of people, those who divide people into two types and those who don't." DSP systems can also be classified in a variety of binary manners. These include, high and low complexity, high and low throughput, high and low precision and so forth. Generally, one of these attributes will be gained at the loss of another. For example, one of the means of achieving very high arithmetic computational data rates has been to use short wordlength (fixed point) arithmetic or a massively parallel architecture. It will be shown that the proposed technique can result in an ultra fast processor without a complexity or power penalty.

II. Processor Requirements

The basic computational primitive, required to support a host of numeric intensive applications (e.g., DSP), must be able to perform the mappings A <- A ± B, A <- AB, A <- A/B, A<- A^2 or SQRT(A), A <- A ± BC. A viable DSP processor should perform these operations with speed and hardware elegance. Currently there are three options and they are:

- weighted number systems (e.g., 2's complement)
- unweighted number systems (e.g., residue arithmetic)
- homomorphic (e.g., logarithmic arithmetic)

All of these systems have both advantages and disadvantages. However, the less well known homomorphic systems offer some significant advantages over the others in multiplier intensive DSP applciations.

III. The LNS Processor [1-7]

The logarithmic number system, or LNS, has been studied for many years in a somewhat casual manner. Recently, semiconductor tecnology has provided the vehicle by which LNS arithmetic can be made a practical reality. In the LNS, numbers are represented as a signed exponent word of the form

$$x = \pm r^{\pm e_x}; \quad e_x = [\text{I bits} : \text{F bits}] \qquad (1)$$

where, in practice, x is coded as a (N+2)-bit word where the N bit exponent e_x is represented as a N-bit word consisting of F fractional bits and N-F = I integer bits. By adjusting the location of the radix point, both large dynamic range and high precision can be achieved.

Arithmetic in this system is performed in the following manner. For C = AoB, S = word sign bit, where o denotes: (2)

1. MULtiply
   $C = AB$, $e_c = e_a + e_b$; $S_c = S_a + S_b$
2. DIVide
   $C = A/B$; $e_c = e_a - e_b$; $S_c = S_a + S_b$
3. ADDition
   $C = A+B$; $e_c = e_a + \phi_r(v)$; $v = e_b - e_a$

   $A > B$ $S_c = S_a$; $\phi_r(v) = \log_r(1 + r^{-v})$
4. SUBtraction
   $C = A - B$; $e_c = e_a + \theta_r(v)$

   $A > B$ $S_c = S_a$; $\theta_r(v) = \log_r(1 - r^{-v})$
5. SQuare RooT
   $C = SQRT(A)$; $e_c = e_a/2$

   $A > 0$
6. SQuaring $C = A^2$; $e_c = 2e_a$

In the LNS, only exponents are manipulated. In practice, the mapping ϕ (ADD) and θ (SUB) are implemented as a memory table lookup. As a result, the wordwidth of the LNS is essentially limited by the table address space associated with high speed semiconductor memory (typically 12-bits for commercially available memory). Case in point is a reported eight-bit custom single chip LNS 3μm CMOS design which makes

extensive use of PLAs instead of dense ROM/RAM [5]. In spite of the reported wordlength limitations, researchers have nevertheless been encouraged with their preliminary results. Swartzlander found that for a realizable wordsize N, in bits, the LNS provided better overall FFT precision than an equivalent width floating point system [4]. The LNS has also been shown to be an effective media in which adaptive digital filters can be implemented [7]. Other studies imply that the LNS can offer an impressive speed-complexity tradeoff. However, the key to all these and future studies, is a long word-length add/subtract unit.

IV. Processor Design

Recently, members of the University of Florida and the Honeywell Corporation studied the LNS as a silicon statement. The theoretical foundation for this work was a data compression scheme developed at Florida which optimally partitioned the lookup table space into subtables. Based on this concept, a six function, 20-bit LNS processor was designed [6]. The exponent of the resulting radix-2 processor consisted of 12 fractional bits and 8 integer bits. The six-function prototype processor was designed using ISL 1.5 micron VHSIC technology. Compared to a commercially available unit, say the AMD floating point chip (FLP), it would contrast as follows:

Item	32-Bit AMD	20-Bit U of F Prototype	U of F Advantage
ADD/SUB	8MFLOP	10.9MFLOP	36%
MULT/DIV	8MFLOP	25.0MFLOP	212%
SQ./SQ. RT	unknown	50.0MFLOP	N/A
Power	7.5 watts	0.390 watts	182%
Cost	$700	prototype I^2L	N/A

Another comparison is offered in Table 1.

The summary statistics of the LNS engine are reported in Table 2. Beside offering outstanding speed/power metrics, the LNS processor has a remarkable regularity to its data flow. As a result, the LNS engine is an excellent candidate to serve as a fault tolerant systolic array processing primitive which is well suited to VLSI or wafer scale integration. It must be noted that the high performance of an LNS processor cannot be exploited by performing isolated arithmetic in a general computing environment (much like that for the 8087 co-processor). The overhead of conversion between an established number format (say FLP) to LNS and from LNS back to FLP would normally be too high to justify using an LNS processor for single arithmetic operations. However, the LNS processor is well suited for use as a dedicated processor in four scenarios:

- Where there is no necessity to adopt any of the established standardized number format (e.g., IEEE Standard) and hence no prerequisite conversion is required. This is examplified by a dedicated LNS systems which also includes logarithmic A/D converters.
- In the implementation of N >> 1 LNS operations where intermediate results can stay in an LNS format. Conversion would be performed only at the input-output boundary.
- Attached array processors where number system conversion (if required) could be performed during DMA transfer or insitu in the input and output data buffers.
- Replacement for slow FLP division units.

The reported LNS processor accepts a 3-bit op-code and two 20-bit LNS operands e_x and e_y (except for the single operand operations) and supplies as an output a 21-bit LNS number including a system overflow flag. The square and square root operations are performed by the use of a shift register. There are three exceptional cases that may arise. These cases are listed below along with the action taken by the hardware.

Exception	Action
1. Overflow:	Force result to MAX (largest number representable). Set the system overflow flag (SYS OVF) to true.
2. Underflow:	Force result to MIN (smallest number representable). Do not set SYS OVF flag.
3. Subtract:	same as underflow (operation X - Y where X = Y)

The exception unit consists of two parts:
i) A 3:1 multiplexer (19-bits wide).
ii) OVF/UF logic for SYS OVF
Most of the area is occupied by ROM and PLAs.

V. AN LNS ARCHITECTURE FOR DSP LNS APPLICATIONS

An LNS architecture for digital signal processing applications is shown in Figure 1. It's main components are an LNS processor and a 6-deep LIFO stack. Also required would be a control PLA. Only one 2-bit operand (denoted X) would be input externally. The other would come from the top of the stack (TOS). It is also possible for both operands to come from the stack as the top of stack (TOS) and next on stack (NOS). Such an architecture would allow us to perform the numerous multiply-accumulate operations present in most DSP algorithms. The function operand supported by the system would be:

Multiplication - X*TOS,TOS*NOS
Division - X/TOS,TOS/NOS
Addition - X+TOS,TOS+NOS

Subtraction - X-TOS, TOS-NOS
Square - X^2, TOS^2
Square root - $\sqrt{X}$, $\sqrt{TOS}$
Push X (load) - Shift stack down.
Pop X - Shift stack up.
Input/Output Format:
Input - one 20-bit LNS word
Output - one 21-bit LNS word

VI. DSP Applications

• Based on the material presented in Table 1, basic DSP arithmetic operations (including L-term FIR and rms mappings) are detailed in Table 3. The potential LNS advantage is well demonstrated in the comparative test.

• By use of a controlled set of prescribed sequence of conditional additions or subtractions, the CORDIC equations can be used to approximately solve either set of the following equations:

$$Y' = K(Y\cos\lambda + X\sin\lambda);\ R = K(X^2 + Y^2)^{1/2};$$
$$X' = K(X\cos\lambda + Y\sin\lambda);\ \theta = \tan^{-1}(Y/X)$$

where K is a constant. Though the concept of CORDIC arithmetic is said to be quite old [8], its implementations and applications continue to evolve especially in areas like computer graphics and analysis. All CORDIC operations are multiplier-free but iterative.

For a system with a fast multiplier, such as the LNS, the power series calculation of these variables can be more accurate, fast and regular from a data flow viewpoint. The efficient generation of R in the LNS has been already reported in Table 3 as a rms term. The production of θ can be accomplished through use of a series expansion of the form $\theta = \tan^{-1}(s) = \Sigma\pi\alpha_{ij}s^i$ where j denotes the cases $s > 1$, $s^2 < 1$, or $s < -1$. The exponentiations s^i (i, i = 1, 3, 5, ...) which make up for about two thirds of the total computational operations count. They can be efficiently and accurately performed in LNS by simple shifts and additions in the LNS.

• The Wigner-Ville Distribution (WVD) and its derivative (the Pseudo-WVD (PWVD)) are excellent time-frequency analysis tools [9]. Historically, the Short Term Fourier Transforms (STFT) has been the main analysis tool for studying signals with a time varying spectra. The STFT is premised on the questionable constant frequency assumption. A more general class of signals is the one of linear frequency variations. This class of signals can be efficiently processed using the PWVD. In the discrete case, the Wigner distribution is given by:

$$W(j,k) = 2\sum_{n=-N/2}^{N/2} q(j,n)w(n)w^*(-n)w_N^{nk}$$

where w(n) is a real window function, $w_N = \exp(-2\pi j/N)$ and q(j,k) is the kernel $x(j+n)x^*(j-n)$. To produce a Wigner kernel would require N complex multiplications, N real multiplications and N complex additions. Using the data from Table 3 it is found that the PWVD kernel can be created in a conventional FLP system and LNS system as:
Conventional: N(1262 + 320 + 222) = 1804 N
LNS: N(386 + 74 + 239) = 699 N
In other words, LNS is approximately 2.5 faster than conventional FLP processors. Of course the produced kernel can be processed by a standard N-point FFT. The latter has been studied for the LNS case by Swartzlander [4].

VII. Conclusions

A significantly improved LNS arithmetic unit has been reported. It more than doubles the wordwidth of all previously published single chip LNS devices. With the accompanying geometric increase in dynamic range and precision, the LNS can now successfully penetrate important DSP issues for the first time. The fast, low power budget processor chip was analyzed in terms of routine DSP tasks including FIRs, CORDIC transforms, and FFT-like transforms. In these cases the LNS unit is seen to enjoy a significant throughput advantage.

The LNS, as a study, is in its infancy. This work, it is hoped, demonstrates the potential of this class of processor. New work is underway to design a 32-bit unit, develop a fast systolic primitive, and an LNS echo cancellor. Other potential applications are obvious.

VIII. References

1. N.G. Kingsbury and P.J.W. Rayner, "Digital Filtering Using Logarithmic Arithmetic," Electron. Letts., Jan. 28, 1971.
2. E.E. Swartzlander and A.G. Alexopoulos, "The Sign Logarithm Number System," IEEE Trans. on Comput., Dec. 1975.
3. S.C. Lee and A.D. Edgar, "The Focus Number System," IEEE Trans. on Comput., Nov. 1977.
4. E.E. Swarzlander et al., "Sign/Logarithm Arithmetic for FFT Implementation," IEEE Trans. on Comput., June 1983.
5. J.H. Lang et al., "Integrated-Circuit Logarithmic Units," IEEE Trans. on Comput., May 1985.
6. F.J. Taylor et al., "A 20-bit Logarithmic Number System Processor," submitted to IEEE Trans. on Computers.
7. V.P. Shenoy et al., "Error Analysis of a LMS Adaptive Digital Filter Implemented with a Logarithmic Number System," Proc. ICASSP 84, San Diego, 1984.

8. J.E. Volder, "The CORDIC Trigonometric Computing Technique," IRE Trans. on Elect. Compt., Sept. 1959.
9. P. Flandrin and B. Escudie, "An Interpretation of the Pseudo-Wigner-Ville Distribution," Signal Processing, vol.6, no. 1, January 1984.

Table 1

Comparison of Commercially Available Short-Wordlength Floating-Point Processors

Operation in ns	INTEL 8087*	WEITEK WTL 1032-5 32-bits $f_{c/k}$=100ns	AMD-MSI 2500LS ** 16-bits mantissa	Sky 2910-Based 32-Bit
ADD	.019	910	74-111 derived	5300
SUBTRACT	.140	910	74-111 derived	8500
MULTIPLY	.021	910	160 derived	5500
DIVIDE	.159	N/A	N/A	15000
SQRT	.159	N/A	320	66370

* Based on 10^6 sequentially called executions on single precision data
** Fast increment/decrement = 10ns
Fixed point: add = 37ns, multiply = 150ns

Table 2

AREA: 62,5000 mil^2 (inclusive of the ROM space)

POWER: 390 mW

EXECUTION DELAYS:

Multiply	40ns (1)
Divide	40ns (1)
Add	92ns
Subtract	92ns
Square	20ns
Square root	20ns

RANGE: (see note 2)
Largest absolute number (positive or negative) $2^{+64}(1.84*10^{+19})$
Smallest absolute number $2^{-64}(5.42*10^{-20})$

PRECISION: $2^{\pm 12.52}$

NOTES:
1. The execution time given here for the multiply/divide operation is 50% higher than a single function dedicated LNS multiplier could achieve. The reason for this is delays associated with decoding the 6 functions.
2. The range of this system is twice that of a 20 bit FLP format with a 13 bit mantissa and a 7 bit exponent. The precision is slightly greater.

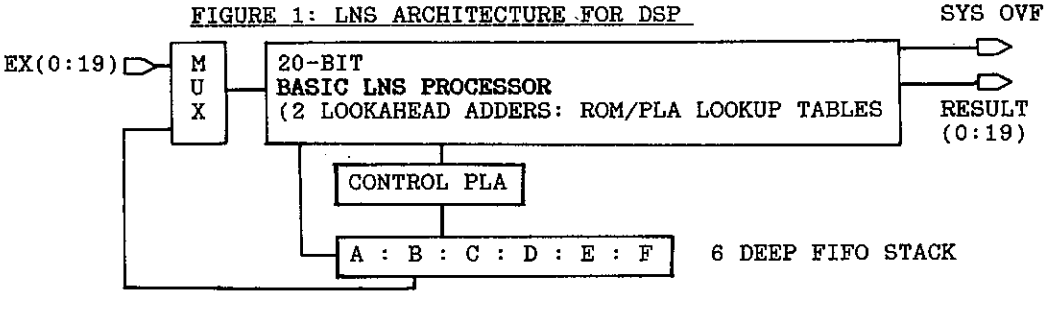
Table 3

Throughput Estimates for Various Basic DSP Operations Using Conventional and LNS Processors

	Operation	A Conventional[1] (ns)	B LNS[2] (ns)	A/B
	ADD,SUB	111	2(37)+45+119	0.933
R	MULT	160	37	4.324
	DIV	320(est)	37	8.649
E	SQRT[3]	320	10(S/R)	32.000
	SQUARE	320	10(S/R)	16.000
A	RMS[4]	320+160L +111(L-1)+320= 271L + 529	37L+ 119(L-1) +10+37= 156L - 58	~1.737
L	FIR[5]	111(L-1)+160L= 271L - 111	119(L-1)+37L= 156L - 119	~1.737
C	ADD,SUB	222	239	0.929
O	MULT	4(160)+222= 1252	4(37)+239= 386	3.269
M				
P				
L				
E	ABS	2(160)+111+ 320 = 751	2(10)+119+10= 149	5.040
X				

[1] Based on survey tables
[2] Based on a 45ns HMOS ROM/RAM
[3] Model: 2 x FLP multiplication delay
[4] rms calculation: $\left(\sum_{i=1}^{L} x_i^2\right)^{1/2}/L$
[5] FIR computation: $\sum_{i=1}^{L} a_i x_i$

FIGURE 1: LNS ARCHITECTURE FOR DSP



This work was supported under a NSF grant and is currently being supported under a ONR/SDI grant.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

FIXED-POINT IMPLEMENTATION OF THE FAST KALMAN ALGORITHM: USING A TMS32010 MICROPROCESSOR

R.ALCANTARA J.PRADO C.GUEGUEN
Département Systèmes et Communications
Ecole Nationale Supérieure des Télécommunications
46, Rue Barrault
75013 Paris, FRANCE

Abstract -Recently new versions of the "fast Kalman" algorithm, providing an extra reduction of the computational complexity, have been proposed. Some problems of the algorithm implementation on floating point computers have been studied, but no fixed-point implementation results with a microprocessor are readily available. In this work we present some experimental results concerning 16-bit fixed-point implementation of pre-windowed and exponentially windowed fast Kalman algorithm in the context of linear prediction and channel equalization. The numerical performance and computational requirements of existing versions of this algorithm are compared in reference to the features of the TMS32010 digital signal processor. The results of our study indicate that a trade-off between the computational requirements, the growth rate of numerical errors and the convergence rate is necessary when using this kind of algorithm.

1. INTRODUCTION

In recent years, new versions of the fast recursive least-squares (RLS) algorithm [1], ("fast Kalman" algorithm, 8p multiplications and divisions per recursion (MDR), where p is the filter order), providing an extra reduction of computational complexity down 5p MDR have been proposed [2]-[5]. All these algorithms minimize recursively the corresponding sum of squared errors and are mathematically equivalent to the standard "slow" RLS algorithm which required $O(p^2)$ MDR [6]. This improvement in computational cost is unfortunately accomplished with a deterioration of the numerical robustness of the fast algorithms when an exponentially weighting factor is used. Furthermore, when finite precision arithmetic is used the numerical properties (stability and accuracy) of all these algorithms are different and the numerical deficiencies eventually destroy the normal operation [7]-[9].

The increasing availability of low cost digital signal processors, having a multiplier/accumulator [10],[11], and the widespread acceptance of adaptive filters, made interesting and important the problem of their implementation in fixed-point arithmetic. To date, some problems of algorithm implementation have been studied using floating point computers but no implementation results using a microprocessor, with fixed-point arithmetic, are readily available. The traditional measures (MDR) of the computational requirements of the algorithms give a rough indication of the real execution time, and only through a specific digital hardware unit can the complexity of the adaptive filter realization be precisely specified, since a wide variety of hardware realizations can be used.

The purpose of this paper is to give some experimental results concerning 16-bit fixed-point implementation of the prewindowed and exponentially windowed fast Kalman algorithm in the context of linear prediction and channel equalization. The numerical performance and the computational and memory requirements of the fast RLS algorithms are compared in terms of TMS32010 operations [10]. Arithmetic operation counts are discussed; in particular the total execution time (including the time for non-arithmetic operations) using the instruction cycle of the TMS32010 is determined.

Due to the analytical difficulties that arise from the inclusion of coefficient update algorithms in finite word-length analysis, the quantization effects were studied by using computer simulation. The results of the simulation were employed to detect the most sensitive variables in each version and to determine appropriate variable scaling (the dynamic range of the variables in the fast RLS algorithms is unknown). Preliminary results of the quantitative analysis of the latter problem and of the tracking performance of RLS algorithms have been presented in [13]-[16].

The comparison of the realization complexity among different fast RLS algorithms confirms the importance of the computation sequence of the arithmetic operations (e.g. a/b and a*(1/b) don't give the same result when finite word-length is used). A wide variety of algorithms can be established if we change the computation sequence of the operations, and their numerical performance in fixed-point realization will be different. We limit our study to the convergence period of the algorithms and we assume that a stabilization method, [8],[17],[18], is necessary in the case of the quantization effects destroying the numerical stability. More extensive results of the

microprocessor implementation of the fast RLS algorithms(e.g. for the complex case and for stabilization techniques) may be found in [20].

2. THE FAST RLS ALGORITHMS

The developed details of all algorithms implemented in this paper, as well as other versions, can be consulted in [1]-[5],[19],[20]. Our interest here is to compare the computation sequence of the variables in the algorithms and to show the principal differences in each version. We can establish two families of the fast RLS algorithm, the first one given by the fast Kalman algorithm, [1] table 2, and the second introduced first in [4] and modified in [5], table 3. The basic structure of all fast RLS algorithms is the same and the two families are related by the pseudo-likelihood variable $\gamma_p(t)$, [5], that permits passing from one version to another through the "Kalman gain" vector ((8) and (9), related by the equation (12)) and the prediction errors (called a priori (1a),(2a),(3a) and a posteriori (1b),(2b),(3b)). Roughly, the improved versions of the fast Kalman algorithm can be obtained by replacing the computation of the a posteriori prediction errors using the definitions, equations (1b),(2b) and (3b), with the relations using the pseudo-likelihood variable, equations (1c),(2c),(3c) and resolving simultaneously the Kalman gain vectors (equations (8a) and (8b)) for the computation of the of the a priori backward error, (2e). Thus, the principal characteristic of the new versions of the fast Kalman algorithm is the reduction of the number of arithmetic operations per iteration, see table 1. However, the improvement in the computational complexity is accomplished with a fundamental deficiency, the deterioration of the numerical robustness of the algorithms. We will see in the next section that a better numerical performance can be obtained by replacing the equation (2e), table 3, by the definition of the backward error energy, equation (2a).

Table 2. The fast Kalman Algorithm, [1]

$$e_p^f(t) = y(t) + A_p^T(t-1)\, Y_p(t-1) \quad (1a)$$

$$A_p(t) = A_p(t-1) + K_p(t-1)\, e_p^{fT}(t) \quad (5a)$$

$$\varepsilon_p^f(t) = y(t) + A_p^T(t)\, Y_p(t-1) \quad (1b)$$

$$\alpha_p^f(t) = \lambda\, \alpha_p^f(t-1) + \varepsilon_p^f(t)\, e_p^{fT}(t) \quad (4a)$$

$$K_{p+1}(t) = \begin{bmatrix} 0 \\ K_p(t-1) \end{bmatrix} - \begin{bmatrix} I \\ A_p(t) \end{bmatrix} \frac{\varepsilon_p^f(t)}{\alpha_p^f(t)} = \begin{bmatrix} M_p(t) \\ \mu(t) \end{bmatrix} \quad (8a)$$

$$e_p^b(t) = y(t-p) + B_p^T(t-1)\, Y_p(t) \quad (2a)$$

$$K_p(t) = [M_p(t) - B_p(t-1)\mu(t)][1+e_p^{bT}(t)\mu(t)]^{-1} \quad (8c)$$

$$B_p(t) = B_p(t-1) + K_p(t)\, e_p^{bT}(t) \quad (6a)$$

$$e_p(t) = d(t) + H_p^T(t-1)\, Y_p(t) \quad (3a)$$

$$H_p(t) = H_p(t-1) + K_p(t)\, e_p^T(t) \quad (7a)$$

$$\varepsilon_p^b(t) = y(t-p) + B_p^T(t)\, Y_p(i) \quad (2b)$$

$$\varepsilon_p(t) = d(t) + H_p^T(t)\, Y_p(t) \quad (3b)$$

$$K_{p+1}(t) = \begin{bmatrix} K_p(t) \\ 0 \end{bmatrix} - \begin{bmatrix} B_p(t) \\ I \end{bmatrix} \alpha_p^{-b}(t)\, \varepsilon_p^b(t) \quad (8b)$$

$$W_{p+1}(t) = \begin{bmatrix} W_p(t) \\ 0 \end{bmatrix} - \begin{bmatrix} B_p(t-1) \\ I \end{bmatrix} \frac{e_p^b(t)}{\lambda \alpha_p^b(t-1)} \quad (9b)$$

$$K_p(t) = \gamma_p(t)\, W_p(t) \quad (12)$$

Table 3. The Fast RLS algorithm, [5]

$$e_p^f(t) = y(t) + A_p^T(t-1)\, Y_p(t-1) \quad (1a)$$

$$\varepsilon_p^f(t) = e_p^f(t)\, \gamma_p(t-1) \quad (1c)$$

$$\alpha_p^f(t) = \lambda\, \alpha_p^f(t-1) + \varepsilon_p^f(t)\, e_p^{fT}(t) \quad (4a)$$

$$\gamma_{p+1}(t) = \lambda\, \gamma_p(t-1)\, \alpha_p^f(t-1)\, \alpha_p^{-f}(t) \quad (10c)$$

$$W_{p+1}(t) = \begin{bmatrix} 0 \\ W_p(t-1) \end{bmatrix} - \begin{bmatrix} I \\ A_p(t-1) \end{bmatrix} \frac{e_p^f(t)}{\lambda \alpha_p^f(t-1)} = \begin{bmatrix} D_p(t) \\ \delta(t) \end{bmatrix} \quad 9a$$

$$A_p(t) = A_p(t-1) + W_p(t-1)\, \varepsilon_p^{fT}(t) \quad (5b)$$

$$e_p^b(t) = -\lambda\, \alpha_p^b(t-1)\, \delta(t) \quad (2e)$$

$$\gamma_p(t) = [1 + \gamma_{p+1}(t)\, e_p^{bT}(t)\, \delta(t)]^{-1}\, \gamma_{p+1}(t) \quad (11d)$$

$$\varepsilon_p^b(t) = e_p^b(t)\, \gamma_p(t) \quad (2c)$$

$$\alpha_p^b(t) = \lambda\, \alpha_p^b(t-1) + \varepsilon_p^b(t)\, e_p^{bT}(t) \quad (4b)$$

$$W_p(t) = D_p(t) - B_p(t-1)\, \delta(t) \quad (9c)$$

$$B_p(t) = B_p(t-1) + W_p(t)\, \varepsilon_p^{bT}(t) \quad (6b)$$

$$e_p(t) = d(t) + H_p^T(t-1)\, Y_p(t) \quad (3a)$$

$$\varepsilon_p(t) = e_p(t)\, \gamma_p(t) \quad (3c)$$

$$H_p(t) = H_p(t-1) + W_p(t)\, \varepsilon_p^T(t) \quad (7b)$$

3. RESULTS OF THE MICROPROCESSOR IMPLEMENTATION

The methodology used here for the implementation of the fast RLS algorithms on the TMS32010 processor is empirical. We simulate the algorithms with a high level language (e.g. FORTRAN) in floating point and fixed-point arithmetic. Afterwards, the results of the fixed-point simulation are used to write the algorithms in TMS32010 Assembly language. This approach facilites the estimation of the dynamic range, unknown, of the variables in the algorithms and permits the validation of the results of the microprocessor implementation. The results of the implementation are given in the table 1 for a 10 order predictor. The details of the experiments can be found in [20]. The second and fourth column correspond to the original improved fast RLS algorithms while the third and fifth correspond to the same algorithms but using the backward error definition. The latter are more robust in respect to the quantization errors, and we remark that the computational complexity, for reduced order, is not augmented when implemented on the TMS320. This fact is explained by the multiplication/accumulate operation of the processor.

EXPONENTIALLY WINDOWED FAST RLS ALGORITHMS COMPLEXITY AND STORAGE REQUIREMENTS					
ALGORITHM	LINEAR PREDICTION				
	MDR	µs / sample		pgm. memory	
Fast-K [1]	8p	215.0		193	
FLS1 [2]	6p	226.6	213.5	240	217
FLS2 [3]	6p	245.6	232.5	264	241
FLS3 [4]	5p	201.7	202.0	253	222
FLS4 [5]	5p	223.0	223.1	285	254
FLS5 [5]	5p	222.3	222.4	285	254
normalized FLS5 [5]	9p		516.0		472

Table 1.

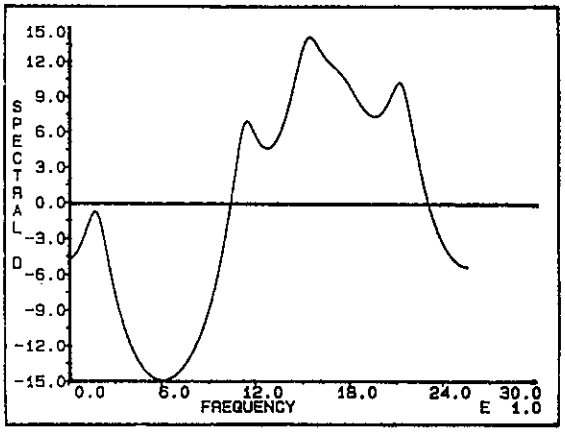


Figure 1

A stability condition is imposed by the choice of the forward prediction energy (4a), the weighting factor used, the word length used and the signal characteristics to be processed. The operation conditions of the algorithms on the microprocessor were determined empirically by computer simulations, and they were used for the processing of the real signal on the TMS32010. The figures 1 and 2 show the implementation results of the speech signal processing with the fast RLS algorithm. A weighting factor equal to 0.99 was used. The figure 1 shows the spectre (iteration 500) and the figure 2 the time evolution of $\gamma_p(t)$

The better convergence rate of the fast RLS algorithms is obtained for the small initial value of the forward energy (4a). In fixed-point arithmetic, the initial forward energy is limited by the word length of the processor used. Furthermore, the convergence of the algorithm will be slow if a high value is chosen. Our experiments showed that the convergence can be improved, when the initial forward energy is high, using the weigthing factor different to one. However, the weigthing factor deteriorate the numerical stability of the fast RLS algorithms. Therefore, a trade-off between the growth rate of the numerical errors and the convergence rate was necessary in our experiments. The figure 3 shows the convergence of the "fast Kalman (F-K) equalizer". The initial

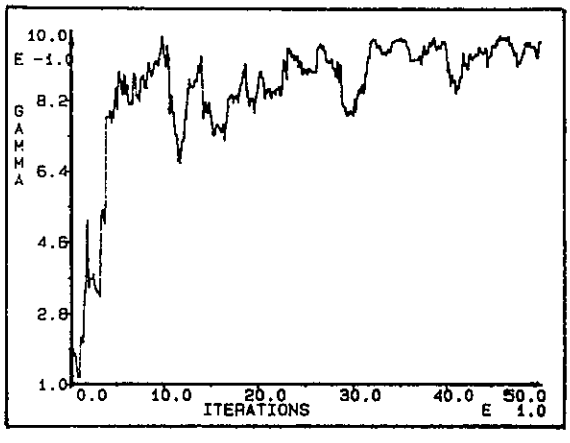


Figure 2

forward error energies in floating point and fixed-point were 0.001 and 1.0 respectively and, the weighting factor was fixed to 0.98 . The executing time given in the table 1 can be improved with a multiprocessor implementation, [21].

4. CONCLUSIONS.

A comparative study of the microprocessor implementation of the fast RLS algorithms was presented. The results showed that when a specific hardware circuitry, like a TMS32010 digital signal processor, is used, the traditional measures of the computational requirements give a rough estimation of the real executing time. Furthermore, since the improvement versions of the fast Kalman algorithm are less robust to the quantificazion effects, we obtained a better numerical behavior using the original versions with a p extra algorithmical complexity. The application examples showed that the microprocessor implementation of the fast RLS algorithms can be now possible by using the recently introduced digital signal processor.

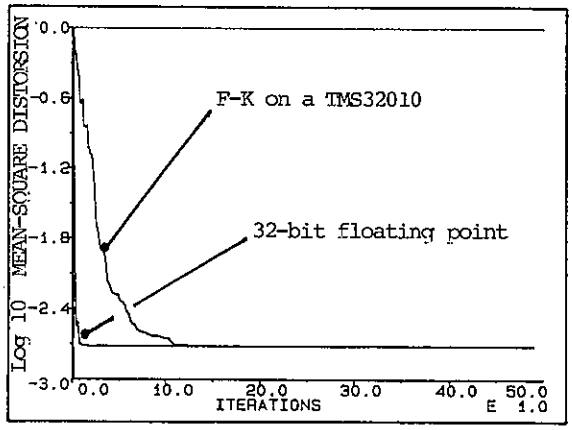


Figure 3. F-K equalizer using a TMS 32010

REFERENCES

[1] L.Ljung, M.Morf and D.Falconer, "Fast Calculation of Gain Matrices for Recursive Estimation Schemes," Int. J. Control, Vol. 27, No. 1, pp. 1-19, January 1978.

[2] J.G.Proakis, Digital Communications, Mc Graw Hill Inc. New York,1983.

[3] J.M.Cioffi and T.Kailath, "Fast, Fixed Order, Least-Squares Algorithms for Adaptive Filtering," in Proc. ICASSP 83, Boston, MA, April 1983.

[4] G.Carayannis,D.G.Manolakis and N.Kalouptsidis, "A Fast Sequential Algorithm for Least-Squares Filtering and Prediction," IEEE Trans. ASSP, Vol. ASSP-31,No. 6, December 1983.

[5] J.M.Cioffi and T.Kailath, "Fast recursive least-squares transversal filters for adaptive filtering," IEEE Trans. ASSP,Vol. ASSP-32, No. 2, April 1984.

[6] L.Ljung and T.Sordestrom, "Theory and practice of recursive identification," Cambridge, MA:MIT Press, 1983.

[7] M.S.Mueller, "Least-squares algorithms for adaptive equalizers," Bell Syst. Tech. J., Vol. 60, pp. 1905-1925, October 1981.

[8] D.W.Lin, "On digital implementation of the fast Kalman algorithms," IEEE Trans. ASSP, Vol. ASSP-32, No. 5, October 1984.

[9] S.Ljung, "Fast algorithms for integral equations and least-squares identification problems," Ph.D. dissertation, Linkoping Univ., Sweden, 1983.

[10] TMS32010 User's Guide, Texas Instruments, Inc., Dallas, TX, May 1983.

[11] K.Marrin, "Six DSP processors tackle high-end signal processing applications", Computer Design, March 1986.

[12] C.Samson and V.U.Reddy, "Fixed point error analysis of the normalized ladder algorithms," IEEE Trans. ASSP, Vol. ASSP-31, No. 5, pp 1177-1191, October 1983.

[13] E.H.Satorius, S.W.Larish and L.J.Griffiths, "Fixed-point implementation of adaptive digital filters," in Proc. ICASSP 83, Boston, MA, April 1983.

[14] F.Ling and J.G.Proakis, "Numerical accuracy and stability: two problems of adaptive estimation algorithms caused by round-Off error," in Proc. ICASSP 84, San Diego, 1984.

[15] F.Ling and J.G.Proakis, "Nonstationary learning characteristics of least-squares adaptive estimation algorithms," in Proc. ICASSP 84, San Diego, 1984.

[16] M.Bellanger and C.C.Evci, "Coefficient wordlength limitation in FLS adaptive filters," in Proc. ICASSP 86, Tokyo, 1986.

[17] P.Fabre and C.Gueguen, "Fast recursive least-squares algorithms: preventing divergence," in Proc. ICASSP 85, San Diego, 1984.

[18] E.Eleftheriou and D.D.Falconer, "Restart methods for stabilizing FRLS adaptive equalizers in digital HF transmission," in Proc. GLOBECOM'84, Atlanta, GA, Nov. 1984.

[19] R.Alcantara, J.Prado and C.Gueguen, "Les algorithms des moindres carres recursifs rapides complexes," in Proc. IASTED Int. Symp., Paris, France, June 1985.

[20] R.Alcantara, "Implementation of fast recursive least-squares algorithms on the digital signal processor," Doctorate thesis, ENST Paris, France, May 1986.

[21] V.B.Lawrence and S.K.Tewksbury, "Multiprocessor implementation of adaptive digital filters," IEEE Trans. Commun., Vol. COM-31, No. 6, pp. 826-835 , June 1983.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

PROCESSOR ARRAYS VERSUS PIPELINES FOR CELLULAR LOGIC IMAGE OPERATIONS

Robert P.W. DUIN and Pieter P. Jonker

Pattern Recognition Group
Department of Applied Physics
Delft University of Technology
The Netherlands

Pipelined processors may be designed such that they perform the same image processing task as a given processor array. The resulting pipelined processor does not have the scanning problem. It will be slower, smaller and cheaper than a full size processor array. Besides, it may be extended with options hard to realize for a processor array. This is worked out for the case of cellular logic operations.

1. INTRODUCTION

Two ways of constructing parallel hardware for processing images are the processor array (PA) and the pipeline (PL). In a PA the pixels are processed in parallel. It consists of a two-dimensional grid of processors, interconnected according to the desired pixel-neighborhood relations. A PA may have the same size as the images to be processed. It also may be smaller and scan the images in some way. In a PL successive operations on the image are processed in parallel. The pixels are fed sequentially into a pipeline of hardware processors, each performing a particular operation on all pixels.

The two architectures will be discussed. A comparison will be done for a PA and a PL, both designed for cellular logic operations. This makes their basic processing elements (PE's) similar. Thereby, features, advantages and disadvantages can be illustrated most clearly.

2. PROCESSOR ARRAYS

In fig. 1 the interconnection scheme is shown between the processor elements of a PA. On each node a processor is available with one or more pixels stored in its memory. The principle of the PE's is shown in fig. 2. They may have an 11-bit input: one for the local pixel (A), eight for its neigborhood, one for an additional pixel or pixel-bit (B) and one for a carry bit (C). The processors may have three outputs: the processed pixel result, the value to be shown to the neighbors and a carry bit. Combinations on input or output are possible. All processors are identical and run the same program (SIMD).

For full progammability the 11-bit input would require a 2048-bit wide program path. This is completely out of the question. The processors have therefore to be restricted in their possibilities considerably. If this is done carefully most of the image processing algorithms can stil be used. Realised PA's like the CLIP4, the MPP and the DAP differ in the way this non-full-programmability is solved. See Preston and Duff [1]. See also Gerritsen [5] for an extensive discussion and comparison of the features of PA's for image processing.

The size of a processor array may be smaller than the size of the actual images to be processed. A PA may scan a larger image. The overhead A_1 caused by this may be in the order of a factor 10 for a hardware solution up to a factor 100 for a complete software solution. This means that if the image has M pixels and the PA has m processors, the processing time is $\lceil M/m \rceil * A_1$ times the time needed for processing an image with M=m. $\lceil x \rceil$ denotes the smallest integer number larger than x. If a total of N basic operations (these are operations that can be defined for a fully programmable PE as in fig. 2, including the loading of data and instruction, processing and storage of result) are needed and the pixels have b bits, then for the processing time of a PA can be written:

$$t_{PA} = A_1 * A_2 * \lceil M/m \rceil * B * N * t_c \qquad (1)$$

A2 is the number of cycles with cycle time t_c needed for one basic operation. Remember that actual PA's are not fully programmable. Missing operations have therefore sometimes to be split into several other ones. A2 may be somewhere between 5 and 50.

3. PIPELINES

In a pipeline a number of operations are performed in parallel by a series of processors on a sequence of pixels, see fig. 3. These processors may be different. After each processing step the addresses are updated and the pixels in the pipe are shifted one position. The operations may be dyadic, monadic or windowed. In the last case a shift register inside a PE keeps a complete neighborhood available for processing. Processors may have more outputs, e.g. one for the result, one for passing the original pixel value and one for a carry bit. A PL with a series of n

identical processors computes in one pass n iterations of the same operation over the whole the image. By combining the serial and the parallel approach more bitplanes may be processed simultaneously. If the pixels have B bits and the PL is b bits wide only ⌈B/b⌉ passes through the image are needed. Advantages of PL's are the possibilities of flexible and data dependent adressing and of recursive window operations.

The total time needed for processing an N step algorithm over an image of M pixels of B bits by a b-bit wide pipeline of n processors can be written as:

$$t_{PL} = A_3 * A_4 * \lceil N/n \rceil * M * \lceil B/b \rceil * t_c \qquad (2)$$

A3 takes into account the overhead caused by splitting an algorithm of N steps into parts of n steps (usually 1). A_4 is the overhead caused by the fact that B-bit pixels have to be processed in steps of b-bit at a time (B > b).

4. A CASE STUDY

In this section the concepts of the PA and of the PL are compared on the basis of 3*3 cellular logic processing. This may be extended with grey value pixel operations in a bit-serial way. The PE of a PA performing this task is already shown in fig. 2. The PE of a PL dedicated to this type of operations should have only bit-serial inputs as the 3*3 neighborhood can be stored within the PE using delay lines, see fig. 4. A network of identical PL-PE's is shown in fig. 5. Note that the PE's of a PA run the same program, while the PE's of the PL shown in fig. 5 may run different programs. A PA of the above type is the CLIP4, see Duff [2]. A PE of a PL as discussed is comparable to the cellular logic part of the DIP-1, described by Gerritsen and Aardema [6], and by Boekamp et al. [7]. A VLSI chip design called the CLP (Cellular Logic Processor) is based on this concept, see Jonker and Duin [8] and Kraaijveld et al. [9].

The CLIP4 PE is shown in fig. 6. The binary processor (2 bits in, 2 bits out) is fully programmable needing 8 input control bits. Together with 8 bits needed for masking the neighbors and 2 bits for enabling B and C this yields 18 control bits to be set for each processing step. The selection made here in reducing the programmability is such that most cellular logic operations still need just one processing step. An important exception is the skeleton, e.g. the PA algorithm by Arcelli et al. [4] needs 16 steps for each iteration. Using the carry bit grey value operations are possible. Grey value table-look-up is difficult as in any PA. A powerful feature of the CLIP, the 'free propagation', which can be used for recursive types of operations, will not be discussed here.

The CLIP4 system as it is designed by University College, London, has a 96*96 processor array, see Duff [2]. A scanning version, the CLIP4S has a 512*4 array that performs a hardware scan over a 512*512 image, see Fountain [3]. A commercial version of the system in modules of 32*32 processors is manufactured by Stonefield Electronics Ltd. This systems uses the same CLIP4 chips, but adds additional memory to each processor.

For the CLIP4 system the constants in (1) may be estimated. The hardware scanning overhead A_1 is about 8 and the number of cycles needed for a basic processing step A_2 is al least 20 For the moment we will neglect the effect of the free propagation as well as the fact that some operations like the skeleton need more than one basic step per iteration. The cycle time of the CLIP4 chip is 400 ns. In order to make a fair comparison with a more recent technology, we assume that 100 ns is possible. Substitution of these numbers in (1) yields for the processing time of the CLIP4 PA architecture roughly:

$$t_{PA} = 20 * \lceil M/m \rceil * B * N * 10^{-6}s \qquad (3)$$

Note that this is a hypothetical result, as neither, the CLIP4 nor the CLIP4S can scan images of an arbitray size.

The PL architecture as presented in fig. 5 has the following features. It can be made fully programmable by using look-up-tables of size 2048*3 bit. Frequently used operations can be stored in ROM, other ones may be loaded in RAM. Loading slows down the overall operational speed, but for large images this will still be neglectable to the processing time. The PE may be able to perform recursive operations as well as using a recursive pipe as described by Gerritsen and Aardema [6]. This would double the table length. The PL produces a result pixel in each cycle step. The cycle time of the CLP chip is 100 ns. The overhead caused by splitting the algorithm in more steps as well as caused by treating bitplanes separately is very small as storing and retrieving intermediate results takes no additional time if it is included in the pipeline. The result for the processing time of the discussed PL architecture is thereby:

$$t_{PL} = \lceil N/n \rceil * M * \lceil B/b \rceil * 10^{-7}s \qquad (4)$$

This is, like (3), a hypothetical result as the CLP chip discussed above does not exist yet. Moreover, it has slightly different features.

The results (3) and (4) are compared in table 1 for two images and for algorithms with N=1 and 100 steps. Three different PA sizes are given and three PL's of the above discussed chips configurations: just one chip, ten serial chips and eight parallel lines of ten serial chips. Note that PA's larger than the image do not have to scan, so A_1=1 in (1). Note also that a pipeline with 8 bits in parallel may process 8 parts of an 1-bit image simultaneously. It is not claimed that the given numbers have any accuracy. They just give the order of magnitude.

From the table we see that a non-scanning PA is always superior. However, if scanning is necessary, 80 PL chips may be as powerful as a PA of 128*128 processors. This can also be

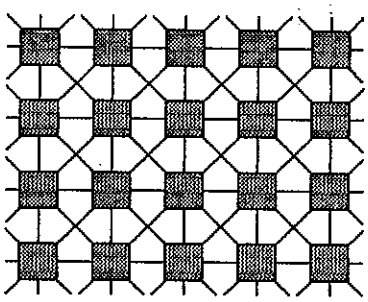


Fig. 1. An 8-connected processor array network.

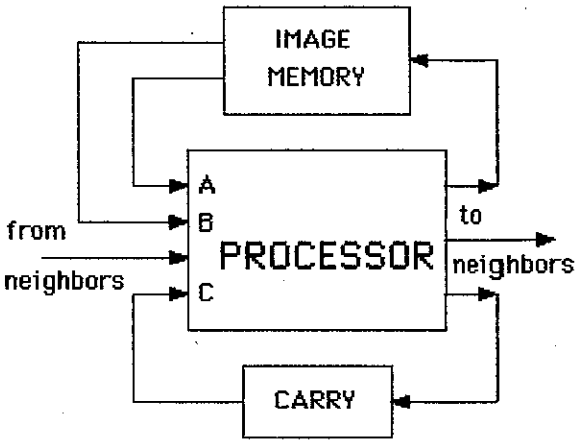


Fig. 2. A processing element.

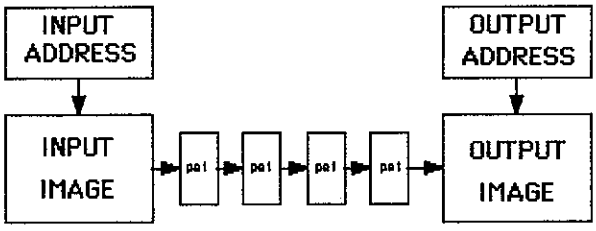


Fig. 3. A pipeline configuration.

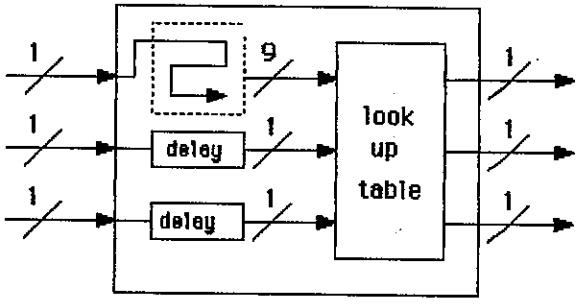


Fig. 4. A processing element for a pipeline.

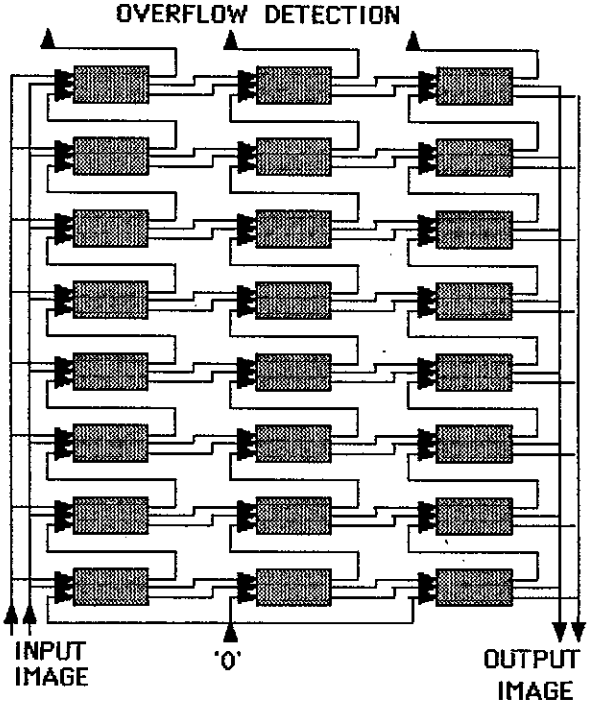


Fig. 5. A PL with 8 parallel series of 3 PE's.

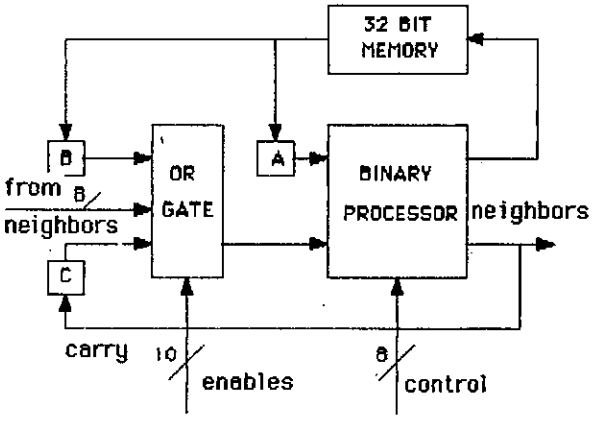


Fig. 6. The CLIP4 PE (simplified)

derived more generally by dividing (3) by (4) and neglecting the $\lceil . \rceil$ operation:

$$t_{PA}/t_{PL} = 200\ n\ b\ /\ m \qquad (5)$$

which says that one PL element (n=1, b=1) may be about as powerful as 200 PA elements (m=200). It is interesting to compare this with the complexity of the chips. The CLIP4 chip has eight processing elements on board and is realised with 3000 transistors. The CLP chip has just one (pipelined) processing element and needs about 20000 transistors (which includes the, partially loadable, table area). The factor 200 is thereby paid by a much higher chip density.

image size (M)	32 * 32		512 * 512	
bits/pixel (B)	1		8	
operations (N)	1	100	1	100
PA size (m)				
32 * 32	2µs	200µs	40ms	4 s
128 * 128	2µs	200µs	3ms	300ms
512 * 512	2µs	200µs	20µs	2ms
PL size (b*n)				
1 * 1	100µs	10ms	200ms	20 s
1 * 10	100µs	10ms	200ms	2 s
8 * 10	10µs	1ms	25ms	250ms

table 1 Estimated processing times for some configurations of a PA and a PL.

5. DISCUSSION AND CONCLUSION

The above comparison is heavily based on neighborhood operations. If one is interested in image processing algorithms that consist of more global types of operation, the presented discussions are of limited use.

The main difference between a PA and a PL is that in the PL the data addressing is done outside and independent of the PE's, while in a PA the positions of the PE's determine the pixel address. This causes a larger flexibility of the PL in image size, address dependent operations, etc. For the PA this causes the complicated scanning problem and the difficulty in combining pixels at large distances in the image.

A second difference is that in the PL the image is outside the PE's and is thereby easily accessible from other processor units. In a PA the memory is intensely connected with the PE's, which makes it hard to access externally.

A third difference is that the number of PE's in PA is much larger than in a PL. In order to keep the PA manageable its PE's have to be small and simple. The PE's of a PL may therefore be much faster and may have a more powerful instruction set or large look-up-tables.

Gerritsen [5] has stated that the PA is the most promising basis for future developments in architectures for image processing. Our conclusion from the comparison of PL's and PA's for cellular logic operations is that if the PA is smaller than the image size, a PL may be constructed that yields the same performance. As the PL has additional advantages (easier I/O and communication with other processors) it may be preferred for those cases. However, if one demands the fastest solution that is possible, the answer is still for many image processing applications: build a PA of the desired size.

Finally, most of our arguments are in one way or another technology dependent. They may be made invalid by new developments, e.g. 8-bit PA's or pyramidal architectures. What we have investigated here is the actual worth of pipelines versus processor arrays.

ACKNOWLEDGEMENT

The research was supported by the Netherlands Organisation for the Advancement of Pure Research (Z.W.O.).

REFERENCES

[1] Preston Jr, K., and Duff, M.J.B., Modern Cellular Automata, Plenum Press, N.Y., 1984.

[2] Duff, M.J.B., CLIP4, in: Fu, K.S. and Tadao Ichikawa (eds.), Special Computer Architectures for Pattern Processing, CRC Press, Boca Raton, Florida, USA, 1982, pp. 65-86.

[3] Fountain, T.J., Postranecky, H., Shaw, G.K., The CLIP4S System, 3rd Int. Conf. of the BPRA, St. Andrews, Scotland, Sept. 1985.

[4] Arcelli, C., Cordella, L., Levialdi, S., Parallel thinning of binary pictures, Electronic Letters 11, 1975, pp. 148-149.

[5] Gerritsen, F.A., A Comparison of the CLIP4, DAP and MPP Processor-Array Implementations, in: Duff, M.J.B. (ed.), Computing Structures for Image Processing, Academic Press, London, 1983, pp. 15-30.

[6] Gerritsen, F.A. and Aardema, L.G., Design and Use of DIP-1: A Fast, Flexible and Dynamically Microgrogrammable Pipelined Image Processor, Pattern Recognition, vol. 14, 1981, pp. 319-330.

[7] Boekamp, R., Groen, F.C.A., Gerritsen, F.A., van Munster, R.J., Design and Implementation of a Cellular-Logic VME Processor Board, SPIE Conf. 596, Arch. and Algorithms for Digital Image Processing, Cannes, Dec. 1985.

[8] Jonker, P.P. and Duin, R.P.W., Considerations on a VLSI Architecture for Cellular Logic Operations, Proc. IEEE Comp. Soc. Workshop on Comp. Arch. for Pattern Analysis and Image Database Management, Miami Beach, Florida, Nov. 1985, pp. 453-462.

[9] Kraaijveld, M.A., Jonker, P.P., Nouta, R., Duin, R.P.W., The VLSI Realisation of a Binary-Image Processor, EUSIPCO 1986 (these proceedings).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DECOMPOSITION OF THE PEN DISPLACEMENT SIGNAL IN THE ANALYSIS OF HANDWRITING MOTORICS.

E.H. Dooijes

Computer Science Department, University of Amsterdam.
Nieuwe Achtergracht 166, 1018 WV Amsterdam,
The Netherlands.

In handwriting motorics research it is important to determine the so-called principal directions of pen motion. A self-consistent technique is presented for obtaining this information from the time signals representing the pen movements.

1. INTRODUCTION

Our project [1] is concerned with the description, in terms of a dynamic model system, of the neuromuscular apparatus involved in the production of skilled, current script. This is done by analyzing the signals resulting from sampling the pen tip's position every 10 milliseconds during the writing of test words by normal, right-handed subjects.
In our model three degrees of freedom are associated with the writing movements. One is related to the slow left-to-right motion ("progression"), the remaining two to the fast letter-forming components. We call *principal directions* of motion those directions emerging if either of the actuators responsible for the fast or *principal* components is excited separately. According to the model, these actuators are antagonistic muscle pairs, independently excited by bilevel ("bang-bang" type) signals which are derived from motor patterns, stored in the brain as a result of many years of practice. A further objective of our project is the reconstruction of the driving signals.
The pen displacement signal contains -apart from the progression component- projections of the principal components onto orthogonal coordinate axes, fixed to the recording apparatus (a data tablet). It is generally not possible -or at least not physically meaningful- to interpret these projections likewise as the the outputs of systems with bilevel excitation. Therefore, one has to perform a transformation of the recorded signals from the laboratory XY coordinate system to a new, generally oblique X'Y' system, moving left-to-right with respect to the former one.
In fig.1a a typical test word is shown; 1b and 1c display the recorded X and Y components of the pen tip trajectory. For later reference we give in (1) the expressions relating for a fixed point P the coordinates x', y' defined in the oblique system X'Y' to its coordinates x, y in the Cartesian system XY:

$$x' = (x - \cot g\mu . y)(\cos\lambda + \sin\lambda . \cot g\vartheta) \qquad (1a)$$
$$y' = (-tg\lambda . x + y)(\sin\mu + \cos\mu . \cot g\vartheta)$$

with $\vartheta = \mu - \lambda$. We will refer to the abbreviated form

$$x' = \alpha . x + \beta . y \qquad (1b)$$
$$y' = \gamma . x + \delta . y$$

Figure 2 explains the meaning of the various parameters.

2. IDENTIFICATION OF THE PROGRESSION COMPONENT

There is evidence [1] that the progression component can be adequately modeled as a linear function of time. After its parameters have been estimated by least-squares fitting (fig.1b) the progression component is removed (fig. 1d). For clarity the XY coordinate system referred to in fig.1

has been rotated a few degrees with respect to the laboratory system, so as to eliminate the progression's Y component.

3. IDENTIFICATION OF THE PRINCIPAL COMPONENTS

The presence of a characteristic slant direction in normal script can be explained by assuming that instants of maximal velocity in one principal direction tend to coincide with instants of zero velocity in the other direction. This *reciprocal motion* (RM) condition is strictly fulfilled for the class of Hilbert transform pairs, discussed below.
Although it seems unlikely that actual writing signals exhibit a simple linear relationship, as do Hilbert transform pairs, and a strict RM relation is neither to be expected, the Hilbert transform model suggests a simple procedure for the determination of the principal directions.
A Hilbert transformer is a non-causal filter introducing a 90° phase shift while having an essentially flat amplitude response over its pass band. Formulated for discrete time signals with unit sampling interval, it can be shown [2] that an output sequence $\{y_n\}$ is related to an input sequence $\{x_n\}$ by

$$y_n = \pi^{-1}\sum x_m(1 - \cos\pi(n-m))/(n-m)$$

the summation index m running from $-\infty$ to ∞ with exclusion of the term with m = n. Since we can also write

$$x_n = -\pi^{-1}\sum y_m(1 - \cos\pi(n-m))/(n-m)$$

the sequences $\{x_n\}$ and $\{y_n\}$ are said to form a Hilbert transform pair. Assuming that the expectation $\langle x_n\rangle = 0$, the cross-covariance function (ccf) of a Hilbert transform pair is

$$c_{xy}(j) \equiv \langle x_n.y_{n+j}\rangle = \langle x_n.\pi^{-1}\sum x_m(1 - \cos\pi(n+j-m)/(n+j-m))\rangle$$

It is readily seen from this expression that $c_{xy} = 0$ and $c_{xy}(j) = -c_{xy}(-j)$. Hence a Hilbert transform pair has an antisymmetrical ccf.

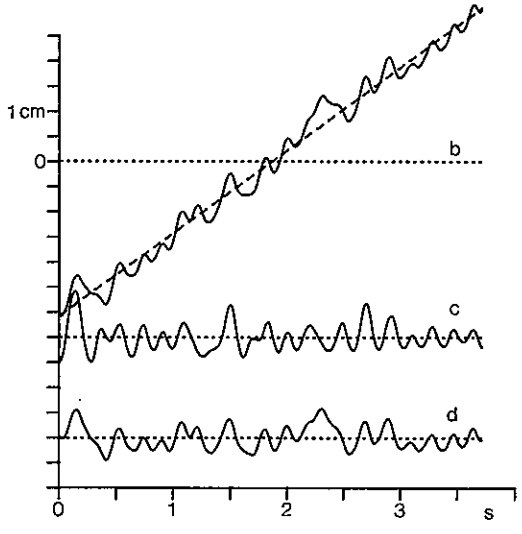


Figure 1: a) script pattern; b) X-component; dashed: uniform progression; c) Y-component; d) X-component, progression removed.

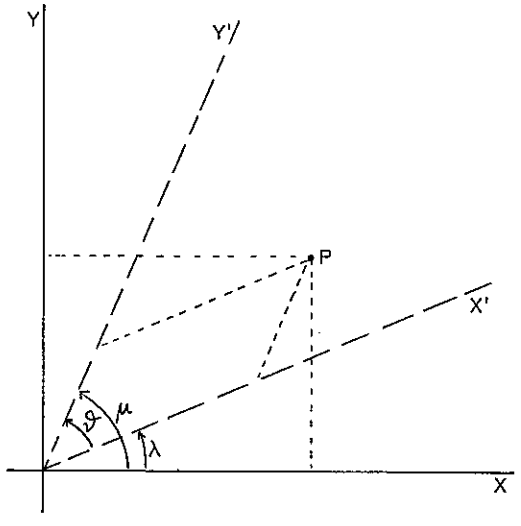


Figure 2: Cartesian and oblique coordinate systems.

We now introduce the notion of *conjugate directions*. Let x(t) and y(t) be a pair of signals defined in the Cartesian XY coordinate system. The cross-covariance function $c_{x'y'}(\tau)$ of the signals x'(t), y'(t) resulting from a transformation to the oblique X'Y' system is related to the auto- and cross-covariances of x(t), y(t) in the following way:

$$c_{x'y'}(\tau) = \alpha\gamma.c_{xx}(\tau) + \alpha\delta.c_{xy}(\tau) + \beta\gamma.c_{xy}(-\tau) + \beta\delta.c_{yy}(\tau).$$

$\alpha,\beta,\gamma,\delta$ denote the elements of the transformation matrix (1) whose values depend on the choice of λ and μ. Expanding $C(\lambda,\mu) \equiv c_{x'y'}(0)$ reveals that C is symmetric in λ and μ; hence $C(\lambda,\mu) = 0$ defines a set of *conjugate directions* $\{\lambda,\mu\}$. For any given λ (or μ), its conjugate μ (λ) can be found by solving this non-linear equation. As it turns out, in the region of interest bounded (rather arbitrarely) by $30° < \mu < 120°$, each λ has a unique conjugate μ. Now, if we assume that a Hilbert pair relationship exists between the principal components of the handwriting signal, the principal directions can be found by selecting from the set $\{\lambda,\mu\}$ of conjugate directions the pair λ,μ which shows antisymmetry of the corresponding ccf. We do so by monitoring the criterion function

$$J = \sum (c_{x'y'}(j) + c_{x'y'}(-j))^2. \qquad (2)$$

The summation index j runs from 1 to M. The value to be assigned to M will be discussed shortly. The correct λ,μ pair is characterized by a minimal value of J. Considering words like "Amsterdam", it is seen that the sense of looping of the pen tip reverses several times in the course of writing. Each reversal implies a 180° phase reversal of one of the component signals at a time. Hence in practice we would expect at best a *piecewise* Hilbert relationship, the role of x'(t) and y'(t) as input and output of a virtual Hilbert transformer alternating at each turning point. However, a piecewise Hilbert relationship, in this sense, still has the effect of producing an antisymmetrical ccf, which can be regarded as the cumulation of a number of ccf's corresponding to short Hilbert-pair segments. In view of this, strict antisymmetry of the bulk ccf will be confined to a relatively small neighbourhood of lag zero. In fact, other factors such as the inadequacy of the inherently linear Hilbert transform model also contribute to this phenomenon. The empirically determined choice of M = 5 time units in (2) reflects the extent of this neighbourhood.

4. RESULTS AND DISCUSSION

On applying this procedure to a collection of test words, a unique minimum of J is found for all except the simplest test words (like "eeeee"). The minimum is far more pronounced for words like "Amsterdam" than for words like "elelele". This should indeed be expected, as for almost sinusoidal signal pairs zero correlation at lag zero is sufficient to guarantee their Hilbert-pair relationship. Since intermediate cases occur as well it is not easy, and probably not meaningful either, to derive for separate test words internal error bounds on the principal directions. Instead we rely on the fact that the principal directions obtained from various test words written by a single subject show a high degree of consistency, the order of magnitude of the dispersion being 5°. Moreover, our results agree with those of an independent experiment [1] for determining the angle ϑ between the principal axes.

REFERENCES

[1] E.H.Dooijes, Analysis of Handwriting Movements. Doctoral thesis, University of Amsterdam 1984.

[2] B.Gold, A.N.Oppenheim, C.M.Rader, Theory and implementation of the discrete Hilbert transform. In: L.R.Rabiner, C.M.Rader (eds), Digital Signal Processing (IEEE Press 1972).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AUTOREGRESSIVR MODELING OF SURFACE EMG WITH APPLICATION TO FATIGUE

O.Paiss and G.F. Inbar

Department of Electrical Engineering,
Technion, Israel Institute of Technology,
Haifa 32000, Israel

The following is an investigation of the ability of the autoregressive (AR) model to describe the processes underlining the recorded surface EMG (Electromyogram). Measurements were carried out on the biceps brachii muscle.
An AR model was calculated for the signal. The AR coefficients, the reflection coefficients and the poles of the AR model were plotted to track their time dependence.
With local muscular fatigue some AR parameter decline with time. There is then a decrease of the similarity of the signal to white noise and it tends to a lower frequency band as observed in the past. During central fatigue - induced by severe sleep deprivation - no consistent changes were observed in the SEMG spectrum or the parameters of its AR model.

1. Introduction.

In many signal processing applications it is desirable to have a condensed parametric representation of the information content of the signal. In biological signal analysis this approach has been applied not only to the relatively simple quasi-periodic deterministic signals, like the ECG but also to the stationary stochastic EEG signal.
Surface EMG (SEMG) is a stochastic quasi-stationary signal with a few time or frequency domain features to rely on when attempting to parametrize it in order to condense its representation. Its spectrum has been shown to contain two major components. The low frequency peak in the range between a few hertz to about forty hertz that represents the firing frequency of the large, lastly recruited, motor units (MU) of the muscles [1,2]. The high frequency range, above the 40 Hz and as high as the recording electrode arrangement permits , represents the morphology of the recorded MU action potentials, (MUAP), as influenced by the position of the electrodes in relation to the innervation zone and the electrodes geometry [3].
There were a few successful attempts of modeling SEMG for data reduction and classification. It was used for the purpose of prosthetic activation [4], for clinical diagnostic classification [5], and for simulating normal and pathological EMG [6]. It is important to establish the relationship between the AR models parameters and the internal physiological processes responsible for the recorded SEMG. The present paper,which is a short version of [7], attempts to do so. The paradigm selected to test the AR modeling applicability to SEMG is muscular fatigue [8], for a known phenomenon, and central fatigue [9] for a phenomenon not studied before with these methods.

2. Surface EMG's Spectrum.

The SEMG was differentially recorded using a pair of surface electrodes that were placed along the biceps muscle 3.5cm apart, and sampled at a rate of 500 samples per second.
The SEMG's spectrum was calculated from 51.2 seconds of signal. 100 periodograms were averaged with 50% overlap to give the estimated spectrum $\hat{S}(w)$. Each periodogram was calculated by FFT of one block of 256 samples of the original signal. Some spectra appear in Fig. 1. The spectrum's envelope (i.e. its general shape) has its maximum at the range of 40-50 Herz,and a small peak at the frequency f_λ that is in the 12-14 Hz range.

3. AR Modeling Of SEMG.

3.1. The Parameters of the model.

In the autoregressive (AR) model (also called linear prediction model) each sample $x(n)$ is described as [10]:

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + e(n) \qquad (1)$$

where:
$x(n)$ - samples of the modeled signal.
a_k - AR coefficients.
$e(n)$ - residual or error sequence.
p - model's order.
The model can be interpreted as a linear system with $e(n)$ as its input and $x(n)$ its output. $e(n)$ is white noise and $x(n)$ is the SEMG. The transfer function of the system:

$$H(z) = \frac{X(z)}{E(z)} = \frac{1}{1+\sum_{k=1}^{p} a_k z^{-k}} \qquad (2)$$

contains poles only. Thus the model can work for signals with well defined peaky spectrum like speech and EEG and can be fitted also to SEMG, as shall be shown below. The power spectrum of the sequence $x(n)$ can be estimated from the model. From eq. (2)

$$S_p^1(w) = |X(w)|^2 = \frac{1}{\left|1+\sum_{k=1}^{p} a_k e^{-jwk}\right|^2} \qquad (3)$$

The AR coefficient (a_i) are calculated by an algorithm [10] that minimizes the residual energy $\sum_n e^2(n)$. From

the 256 samples in each block, the following parameters are calculated:
a_i, $i=1,2,\cdots,p$ - The AR coefficients.
k_i, $i=1,2,\cdots,p$ - The reflection coefficients.
P_i, $i=1,2,\cdots,p$ - The poles of the AR filter $H(z)$.
V_p - The normalized residual energy.
The poles are complex numbers that appear in complex conjugates. The calculation of the poles, in addition to a_i, is needed because a change in one pole will change all the a_i's and thus make it impossible to track changes in the latter,and vice versa, so both parameters should be investigated.
The k_i's are produced by the iterative algorithm that calculates the a_i's from the normal equations [10]. They have useful properties: $|k_i|<1$ for all i, and k_i is independent of the model's order p.
Tracing temporal changes in the parameters is done with a moving window containing N=256 samples from which the parameters are calculated. After the calculations the window advances in time ,thus continuous graphs of the a_i's and k_i's are formed. Those graphs are shown in Fig. 2 with a non-fatiguing 1Kg load at the hand.
The oscillations that are being seen in the graphs do not reflect actual changes in the AR model but are rather due to the random nature of the sampled SEMG. This randomness causes a *significant variance in the estimation of the parameters*. The bias of the estimates was found to be negligible
A lower bound for the variance of the a_i's is given by the Cramer-Rao Lower Bound (CRLB) [11]:

$$Var\{\hat{a}_i\} \geq \frac{1}{N}[(1+a_1^2+a_2^2+\cdots+a_{i-1}^2) - (a_p^2+a_{p-1}^2+\cdots+a_{p-i+1}^2)]; i = 1,2,\cdots,p \quad (4)$$

where:
N - no. of samples from which the a_i's are found
a_i - true AR coefficients
$\hat{a}_1$ - estimated AR coefficients
p - AR model's order.
In order to compare the measured variance with the lower bound, ergodicity is assumed for the random process $\hat{a}_i(t)$, it is assumed that their average over 51.2 sec. is the true value and the power of their oscillations is the variance of the process.

Table 1: Oscillations' power compared to CRLB for AR models of order 2. (N7 is white noise filtered by a seven poles filter and the others are SEMG.

Coeff.	Average Value	File	CRLB	Oscillations' power
a_1	-0.875	IZ2	0.00337	0.00505
a_1	-1.164	RN2	0.00264	0.00432
a_1	-1.151	N7	0.00293	0.00566
a_2	0.370	IZ2	0.00337	0.00386
a_2	0.570	RN2	0.00264	0.00277
a_2	0.499	N7	0.00293	0.00334

The measured variances are not far from the lower bounds for the SEMG (the distances are not bigger than for the filtered white noise case). This supports the claim that the oscillations are due to the variance of estimation. In [7] it is proven that using a model whose order is exessive,raises this lower bound.
The oscillations do not decrease even for a large p. Despite the oscillations, it can be claimed that the graphs do not have trends and that the AR model for the SEMG is time invariant in case of a weak constant force.
The description of the poles' position can't be plotted vs. time like the AR parameters in Fig. 2, because P_i is complex and the poles are not ordered in the index i. Thus it is best to display all the poles simultaneously in the Z plane. Every step of the moving window contributes one dot for each pole, as is seen in Fig. 3. Since the AR filter, H(z), is stable, all the poles lie inside the unit circle. In this method for displaying the poles the time dimension is lost. The dots appear in large clusters whose spreading results from the variance of the estimates - like the oscillations that were met previously in Fig. 2. An odd number of poles, p (which allows one real pole) decreases the spreading of the poles. When the number of poles increases, the clusters corresponding to high frequencies become larger, as seen in Fig. 3.
The order of the AR model can be estimated from the resulting parameters or from the residual signal.Details can be found in [7].

3.2. SEMG Spectral Estimation From the AR Model.

From the AR coefficients a_i, the spectrum can be estimated by $\hat{S}_p^1(w)$ of Eq. 3. In order to reduce the variance of the estimate, a 100 estimated periodogram $\hat{S}_p^1(w)$ were averaged to give $\hat{S}_p(w)$. Fig. 4 shows $\hat{S}_p(w)$ for various model orders p. The spectrum $\hat{S}(w)$ as estimated via FFT is shown for the same signal in fig. 1a.
If one does not only seek the general shape of the spectrum but also for the spectral peak f_λ (= the average firing rate of the dominant motor units), one observes in Fig. 4 that the necessary model order is around 30 . For a higher sampling rate a larger p is needed and vice versa. If the sampling rate is divided by 2 (there will still be no aliasing errors), the needed order p will be halfed .
Decreasing the order p, while keeping the resolution in a part of the estimated spectrum $\hat{S}_p(w)$, can be performed by using the Selective Linear Prediction (SLP) [10]. Using this method the AR model fits only a portion of the spectrum defined by $w_1<w<w_2$. An advantage of this method, compared with lowering the sampling rate, is that the desired frequency range does not have to start at zero but at any ω_1. Spectra estimated by a SLP method appear in Fig. 5. It is seen that if a small frequency range is wanted, the spectrum can be estimated with a lower p without loss of resolution. For identifying f_λ, an order of 7 is sufficient instead of 30.

4. Example Of Fatigue.

4.1. Local Fatigue.

A prolonged exertion of force causes local fatigue in the muscles involved that causes a reduction in the propagation velocity of the action-potentials in the muscle fibers. This reduction leads to a widening of the MUAPs, so their spectrum shifts towards the low frequencies. In addition to these changes, during exertion of force there is an increase in the firing rate [1,8].
The time dependence of some parameters of the AR model is displayed in Fig. 6; they are calculated during exertion of 100% MVC (=maximum voluntary contraction) force leading to a fast local fatigue. In the a_i's or k_i's, for small i, there is a pronounced trend.
For two experiments , the average values of a_1 (p=30) dropped from -0.962 to -1.081 and from -0.829 to -1.029, when comparing values before and after MVC. Those trends are related to the mentioned spectral shift. The shift implies that the SEMG becomes more narrow banded and less white noise, and consequently the resi-

dual energy V_p decreases because the signal is more predictable,and the autocorrelation function $r(i)$ widens , and k_1 (which is $-r(1)$) decreases. The influence on other a_i's and k_i's is more complicated.
The $a_1(t)$'s behavior is similar to that of the time change of the mean frequency of the spectrum of the SEMG. The trend of a_1 and k_1 can be used for monitoring fatigue, other a_i's are not as suitable for this purpose as seen in Fig. 6. The poles of the AR model do shift towards the low frequencies , but the shift is hard to follow.
All those changes in the parameters are due only to changes in the MUAP shapes and are not influenced by the firing rate, since the firing rate influence is observed only with high order models.

4.2. Central Fatigue.

Central fatigue can be caused by sleep deficit. Its possible influence on the SEMG was tested on 3 men, who were deprived of sleep in the following manner: 24 hours of no sleep plus 36 hours of 7 minutes sleep every half hour. Their SEMG was measured in the fatigue state and later, after rest (both in the 5% MVC level). There were no consistent changes in the spectrum of the SEMG or in the AR coefficients.

5. Conclusions

The general shape of the spectrum of the SEMG describes the spectrum of the spatially filtered MUAPs of the dominant motor units. Its peak is the famous Piper frequency. The low frequency spectral peak, corresponding to the firing rate of these units, appear at the frequency f_λ that is in the 12-14 Hz range.
The parameters calculated from the model (a_i, k_i, P_t, V_p) can be used to characterize the processes responsible for the generation of the EMG,but they suffer from a significant variance in their estimation process. The variance is expressed as oscillations in the graphs of the estimated coefficients as a function of time, and as large clusters of the poles location in the Z plane.
This variance must be studied if one is to classify signals according to the AR coefficient values. For example: in one case the average value for a_1 was -0.875 and the variance was 0.00505. The distribution of a_1 is nearly normal. It can be calculated that the 99% confidence interval is ±0.015 around the average value. The estimates of the a_i's were nearly unbiased and errors were only found in the third digit.
The problem of determining the AR model's order p was discussed in [8]. The conclusion was that the determination is subjective. In order to describe the spectrum of the MUAP (the envelope of the spectrum of the SEMG) a low order model (2 to 6), is sufficient. To identify the MUs firing rate the model's order has to be about 30 (when the sampling rate is 500). In the last case the order can be reduced without loss of resolution in the lower frequencies, by decreasing the sampling rate or by using selective linear prediction.
Local fatigue, caused by prolonged exertion of high force, changes greatly the shape of the MUAP and thus changes the general shape of the spectrum of the SEMG which shifts to the left. It was seen that the AR coefficients a_1 and k_1 can be used to monitor fatigue.
Central fatigue induced by sleep deprivation, did not cause any consistent changes in the spectrum or in the AR model of the SEMG. It is possible that under the given experimental conditions no changes occur in the underlying processes responsible for the generation of SEMG. It is also possible that if such changes occurred, they were not detected by the present method.

REFERENCES

(1) De Luca, C.J., "Physiology and mathematics of myoelectric signals", IEEE Trans. BME 26, 313-325 (1979).

(2) Lago, P. and Jones, N.B., "Effects of motor-unit firing time statistics on EMG spectra", Med. Biol. Eng. & Comput. 15, 648-655 (1977).

(3) Lindstrom, L.H. and Magnusson, R.I., "Interpretation of myoelectric power spectra: A model and its applications", Proc. IEEE 65, 653-662 (1977).

(4) Graupe, D., Kohn, K.H., Kralj, A. and Basseas, S., "Patient controlled electrical stimulation via EMG signature discrimination for providing certain paraplegics with primitive walking functions", J. Biomed. Eng. 5, 220-226 (1983).

(5) Inbar, G.F. and Noujaim, A.E., "On Surface EMG spectral characterization and its application to diagnostic classification", IEEE Trans. BME 31, 597-604 (1984).

(6) Maranzana-Figini, M., Molinari, R. and Sommariva, G., "The parametrization of the electromyographic signal: An approach based on simulated EMG signals", Electromyogr. Clin. Neurophysiol. 24, 47-65 (1984).

(7) Paiss, O. and Inbar, G.F., "AR modeling of surface EMG and its spectrum with application to fatigue",Technion EE pub. no. 534 (1985).

(8) De Luca, C.J. and Broman, H., "Myoelectric manifestation of localized muscular fatigue in humans", CRC Critical Reviews in Biomed. Eng. 11, 251-280 (1984).

(9) Johnson, L.C. and Naitoh, P., "The operational consequences of sleep derivation and sleep deficit", NATO AGARDograph No. 193 (1974).

(10) Makhoul, J., "Linear prediction: A tutorial review", Proc. IEEE 63, 561-579 (1975).

(11) Friedlander, B., "On the computation of the Cramer-Rao bound for ARMA parameter estimation", IEEE Trans. ASSP 32, 721-727 (1984).

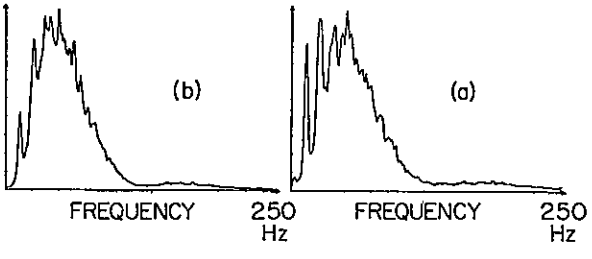


Fig. 1: SEMG spectra from various experiments (average of 100 periodograms representing 51.2 sec. of data).

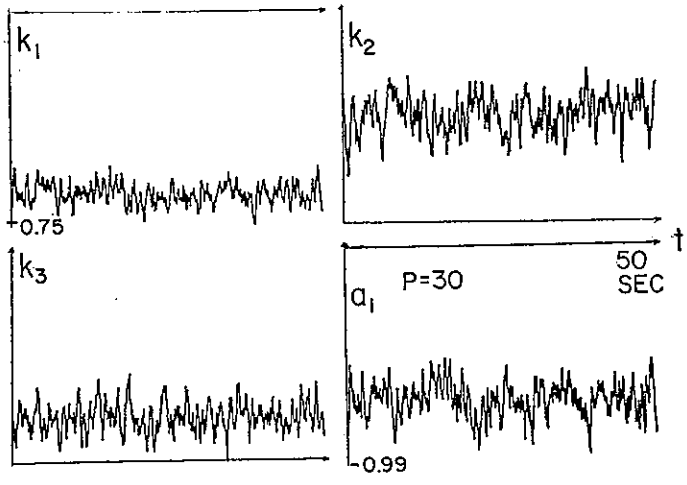


Fig. 2: AR and reflection coefficients estimated as a function of time. All graphs are on the same scale (except bottom right), and all represent 51.2 sec. of the same data.

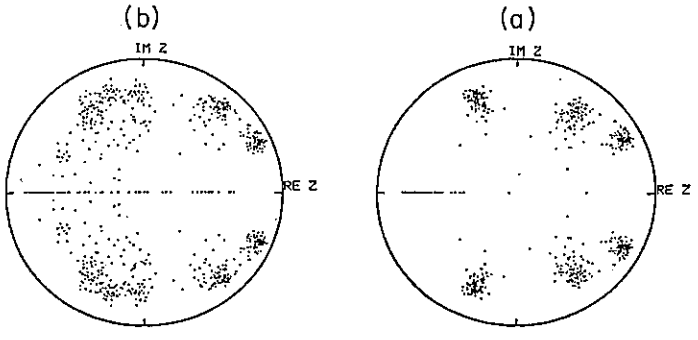


Fig. 3: Poles location of AR models of SEMG signal. a) $p=7$, b) $p=9$ (51.2 sec. of data).

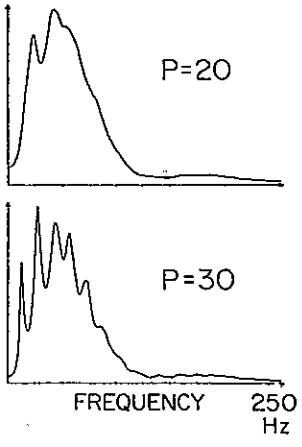


Fig. 4: Estimated spectra from the AR model coefficients for various model orders p.(average of 100).

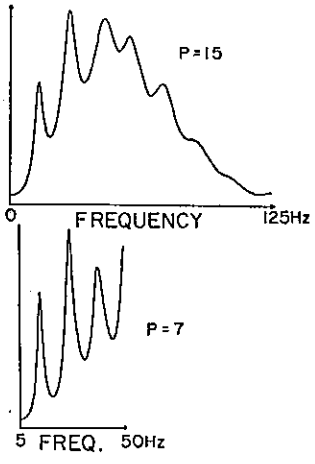


Fig. 5: Estimated spectra from the AR model coefficients computed using SLP method for various order p and frequency ranges (average of 100).

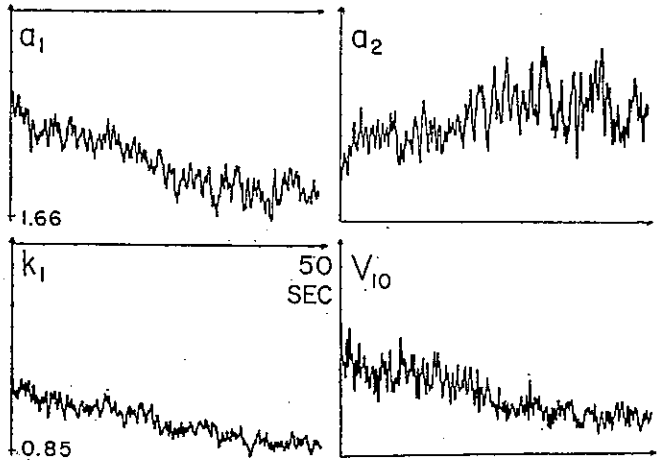


Fig. 6: AR parameters' trends of SEMG during exertion of MVC leading to local fatigue.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

MULTISPECTRAL PROCESSING OF MAGNETIC RESONANCE IMAGE SEQUENCES

W. Döler, T. Schormann, F. A. Stichnoth*

Institute for Medical Physics and Biophysics, *Department of Radiology,
University Göttingen

Contrast in MR images depends on the specific tissue parameters (proton density, T1-, T2-relaxation times) and on parameters of the pulse sequence (TR: repetition time, TE: echo time, TI: inversion time). Since so many variables are influencing image contrast, the images are often difficult to interpret and the observer has to extract relevant informations from all images of the sequence. In order to proceed with this well-known difficult task of picture interpretation, we propose to make use of techniques of "Principal Components Analysis" and pattern recognition.

1. INTRODUCTION

Generation of magnetic resonance images is comparable to multispectral imaging in remote sensing. Here the same geographic area shows different contrast for the various spectral channels [1, 2]. In the case of MR images the contrast of the same anatomic region is affected by the parameters of the pulse sequence. For a spin echo sequence, which is frequently used in MR imaging, the signal S arising from a region with specific parameters ρ (proton density), T1-, T2-relaxation times is given by the relation [3]:

$$S = \rho \left[1 - 2 \exp\left(\frac{TE - 2TR}{2T1}\right) + \exp\left(\frac{-TR}{T1}\right)\right] \exp\left(\frac{-TE}{T2}\right)$$

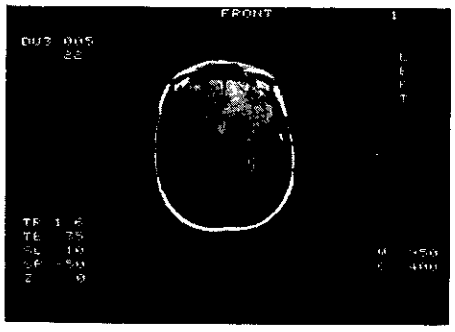
The grey level of each pixel and thus the image contrast depends, beside the tissue specific parameters (ρ, T1, T2), on the choice of the pulse sequence parameters TR (repetition time) and TE (echo time). Figure 1 shows an image sequence with different echo times (TE = 35 ms, 70 ms, TR = 1.6 s). The different contrast is clearly to be seen.

Compared to multispectral image sequences, the parameters of the pulse sequence correspond to the various spectral channels. For this reason multispectral image processing techniques should be applicable to MR images. The following investigations are dealing (a) with the generation of primary images ("Principle Components Analysis") from MR image sequences for contrast enhancement and data compression and (b) with the application of pattern recognition techniques to MR images for tissue characterisation.

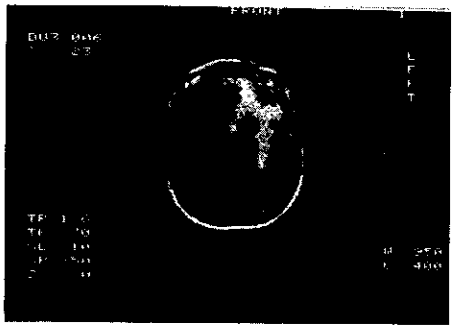
2. PRINCIPAL COMPONENTS ANALYSIS

2.1 Basic Principles

PCA has its origin in statistics. In digital image processing applications it is known as Karhunen-Loève or Eigenvector transformation [4]. By application of this method to statistic dependent variables the correlation and therefore the redundance of the data is removed. Fig. 2 shows the two dimensional histogram of the images in Fig. 1. Its value at the coordinates G1, G2 is the number of corresponding



(a)



(b)

Fig. 1 Spin echo sequence
(a) TR = 1.6 s TE = 35 ms
(b) TR = 1.6 s TE = 70 ms

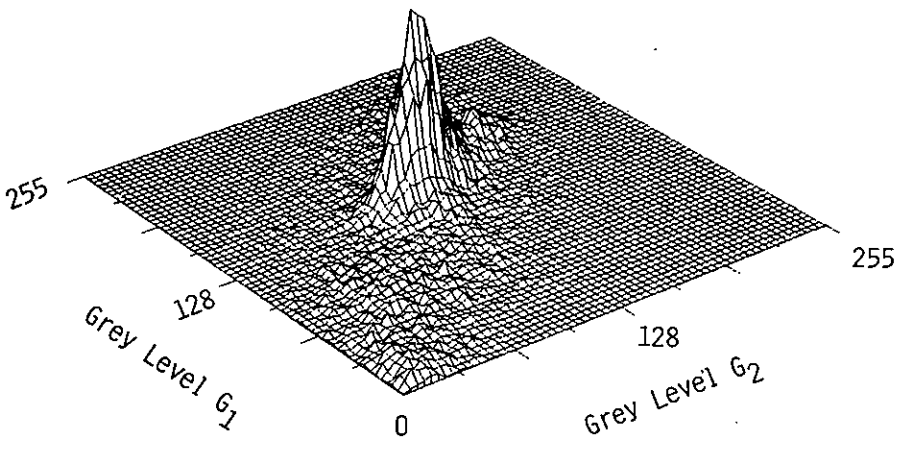


Fig. 2 Two dimensional histogram of images in Fig. 1

pixel pairs having grey level G1 in the first image (echo time TE1) and grey level G2 in the second image (echo time TE2). The two dimensional histogram can be interpreted as scatter diagram, known from correlation analysis of random variables. As to be seen, the grey levels of both images are correlated.

Decorrelation of the data can be achieved by relating the grey levels to a new coordinate system, in which they are statistical independent. These considerations are also valid for more than two variables.

2.2 Principal Component Calculation

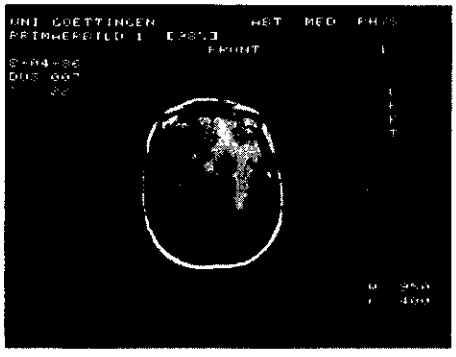
The image sequence is represented by image vectors $\underline{X}(i)$ of dimension M (= n x n). The index i (i = 1, ...K) belongs to an image with a specific echo time TE(i). After normalization of the images and subtraction of their average intensity levels the transformed images $\underline{Y}(i)$ (principal component images) are obtained by the following transformation [5]:

$$\underline{Y}(i) = \sum_j e_{ij}\, \underline{X}(j) \qquad (1)$$

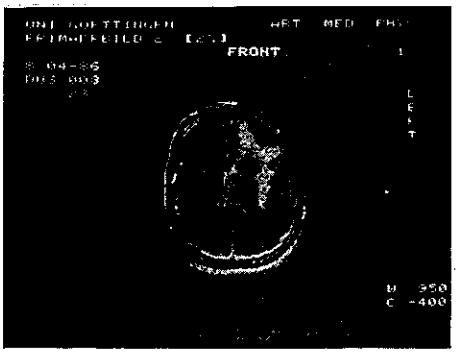
The transform coefficients $E(i) = (e_{i1}, \ldots e_{iK})$ are eigenvectors of the covariance matrix $\underline{A}$ with coefficients

$$a_{ij} = K^{-1}\, \underline{X}(i)'\, \underline{X}(j) \quad i,j = 1, ..K \qquad (2)$$

Fig. 3 shows the principal component images of Fig. 1. As stated in the preceeding section, their grey levels are uncorrelated. The two dimensional histogram (Fig. 3) shows that the values are distributed around a line parallel to the G1'-axis, which is a hint to uncorrelated data. A detailed mathematical analysis may be found in [4].



(a)



(b)

Fig. 3 Principal component images
(a) First principal component V = 98 %
(b) Second principal component

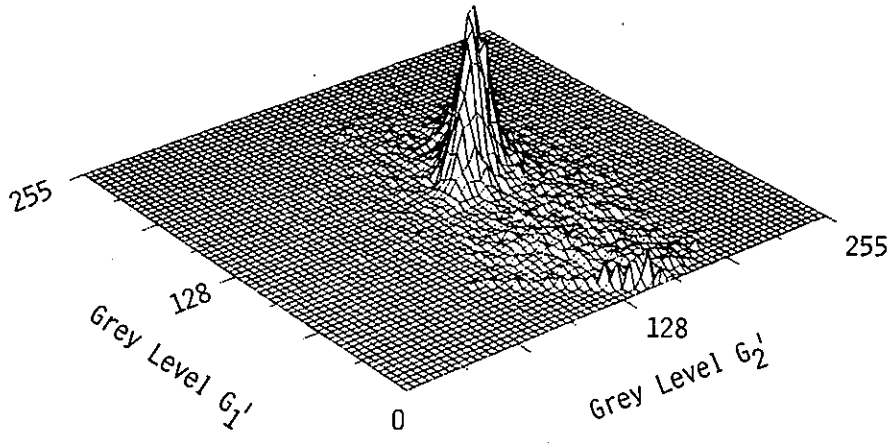


Fig. 4 Two dimensional histogram of principal component images in Fig. 3

The order of the principal component images is given by the quantitative distribution of the corresponding eigenvalues λ_i. The energy packing property of the eigenvector transform manifest itself as a significant increase in contrast (variance) in the first principal components. The relative variance V, which is computed by

$$V = \sum_{i=1}^{m} \lambda_i \Big/ \sum_{i=1}^{K} \lambda_i \qquad (3)$$

indicates, how much information (contrast) of the image sequence is packed into the first (m) principal components [6]. The investigated sequences with K = 2, 3, 4 images have V values of more than 95 %. Comparing the image sequence of Fig. 1 with the first principal component in Fig. 3, the increase in contrast is clearly to be seen. Fig. 5 shows the corresponding grey level histograms of these images.

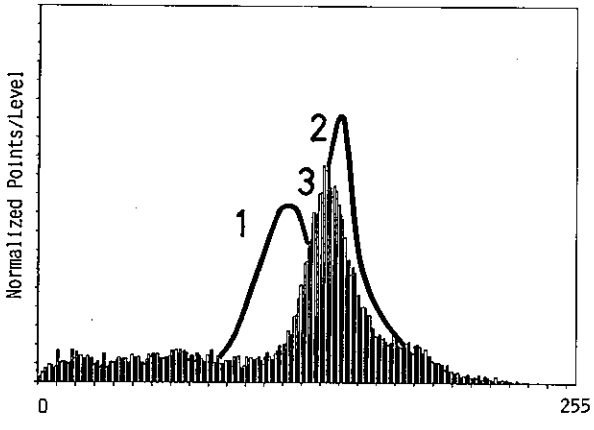
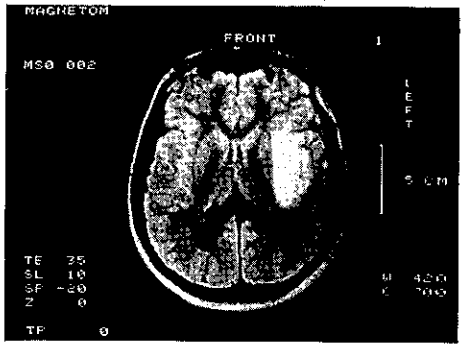
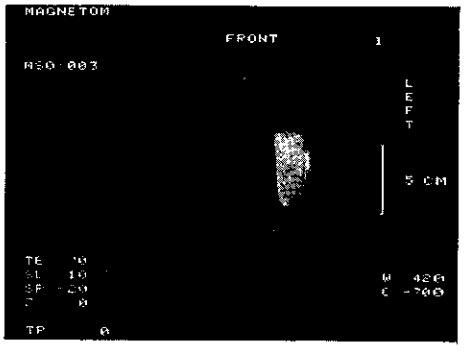


Fig. 5 Grey level histogram
(1,2) Images in Fig. 1 (schematic)
(3) Principal component image (Fig. 3a)

In Fig. 6 and 7 a second example demonstrates principal component calculation of MR-images.



(a)



(b)

Fig. 6 Spin echo sequence
(a) TR = 2.0 s, TE = 35 ms
(b) TR = 2.0 s, TE = 70 ms

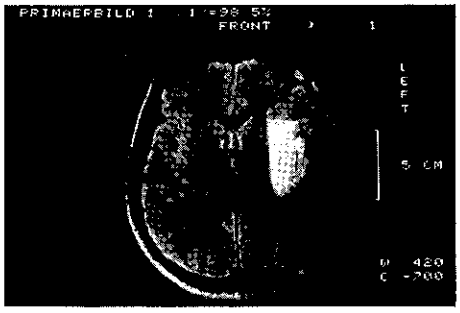


Fig. 7 First principal component V = 98.5 %

3. CLASSIFICATION OF MR-IMAGES

3.1 Introduction

Pattern recognition techniques are often used in multispectral image analysis. First applications of these methods to MR-images by satellite imaging systems are described in [7]. For medical image analysis, a supervised classification is a proper procedure, because the analyst with his a priori knowledge can teach the classifier to recognize the informational classes.

3.2 Maximum-Likelihood Classification

A clustering procedure based upon the gaussian maximum likelihood decision rule is used to perform a supervised classification of the MR-image sequence [8]. The algorithm is non iterative and requires a priori specification of the number of clusters by selection of apropriate test regions. Under the assumption that the pixels are normal distributed, the test regions in the images of the sequence are characterised by a gaussian density function with specific parameters μ and σ. For each pixel vector of the sequence and each class the K-dimensional multivariate density function is calculated. The pixel is assigned to that class where the function has its maximum. The classes are coded by different colors as shown in Fig. 8. Compared to Fig. 1 the tumor region is fairly good classified. Same results are obtained from other examples. Classification and differentiation of other tissue types is dependent on the choice of the test regions (size, homogenity) and seems to be more difficult compared to multispectral image analysis.

4. CONCLUSIONS

The investigations show that multispectral image processing techniques are applicable to MR-image sequences. PCA can be used for contrast enhancement and data compression for storage and archiving. Classification of MR-images is a more sophisticated procedure, which yet has to be tested in a clinical MR-study.

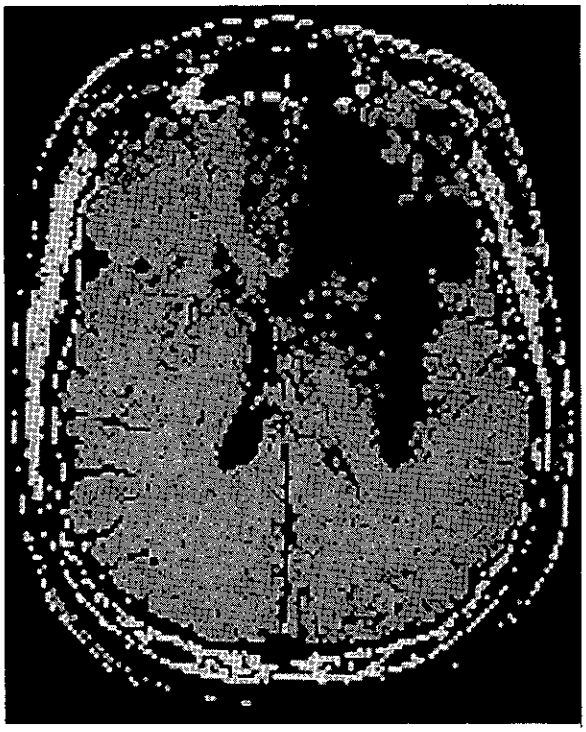


Fig. 8 Classification of the image sequence in Fig. 1
(yellow, green): brain
(red, violett) : tumor
(blue) : skull
(black) : not classified

REFERENCES

[1] Ready, P. J. and Wintz, P.A., IEEE Trans. on Communications, COM-21 (1973) 1123
[2] Duvernoy, H. and Leger, J., Optics Communications 32 (1980) 39
[3] Ziedses des Plantes, B.G. et al., Radio Graphics 4 (1984) 869
[4] Gonzales, G.A. and Wintz, P., Digital Image Processing (Addison Wesley, London, 1977)
[5] Murakami, H. and Vijaya Kumar, B.V., IEEE Trans. Patt. Anal. Machine Intell., PAMI-4, (1982) 511
[6] Hall, E.L., Computer Image Processing and Recognition (Academic Press, New York, 1979)
[7] Vannier, M.W. et al., Radiology 154 (1985) 221
[8] Swain, P.H. and Davis, S.M., Remote Sensing: The Quantitative Approach (McGraw-Hill, New York)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

T.7

A MULTIPLE CHANNEL DATA ACQUISITION SYSTEM - AN APPLICATION IN CARDIOLOGY

N.F.S.Especial, F.J.S.Almeida, J.C.R.L.Fernandes, A.M.Aleixo, A.K.Suleman, J.C.Amado

Laboratório Nacional de Engenharia e Tecnologia Industrial
Departamento de Electromecânica e Electrónica
Azinhaga dos Lameiros, 1699 Lisboa Codex, Portugal*

This work presents a modular and compact 36 simultaneous channel data acquisition system (MC301) and its current application in cardiology (cardiac mapping), the MAPCARD system. The details and main specifications of both systems are referred, as well as a brief description of some application fields. Finally some of the results already obtained are also presented.

1. INTRODUCTION

The increasing power and facilities of personal computers together with the decreasing price of electronic components are creating new interesting opportunities to develop powerful "PC-based instrumentation".

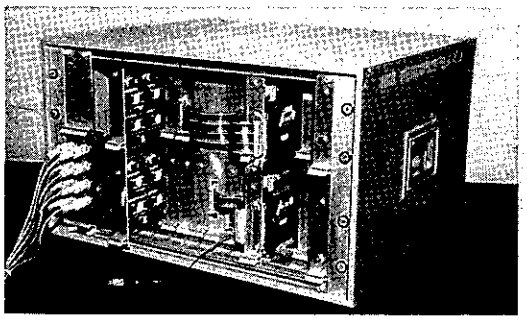
This work presents a modular simultaneous multichannel signal data acquisition system (MC301) that can have a wide range of applications due to size, price and modularity.

The unit has several important features that can be software programmed and can work standing alone or connected via communication protocols to any computer. The unit has been applied to develop a very powerful and convenient diagnostic method in cardiology.

The whole system, MAPCARD, includes the unit and the personal computer and its peripherals. The personal computer takes charge of the user interface, controls the data acquisition unit, and makes some simple but important signal processing tasks. In particular the computer determines the relevant points of ECG signal, makes the baseline adjust, the root mean square of all channels, and amplitude sums of Q, R, S and ST80. Another important task carried out in the computer is the graphical presentation of data and results, as illustrated in section 3.3.

2. THE DATA ACQUISITION SYSTEM MC301

The MC301 is a modular and compact simultaneous multiple channel microcomputer data acquisition system, with storage capabilities, designed for a wide range of applications in the biomedical field. The system is an application of "A Multiple Channel Data Acquisition System" [1], developed by the authors.



View of the MC301 system

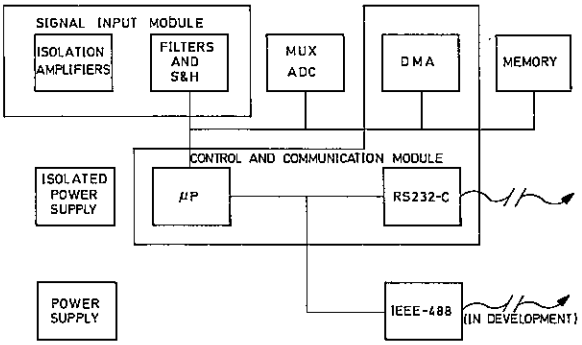
Figure 1

The main specifications of the system, shown in Fig. 1, are:

- On line configuration of:
  - .number of channels to be acquired simultaneously (up to 36, single ended);
  - .sample rate (1,2,4 msec);
  - .low pass filter cutoff frequency (60, 150, 250 Hz);
  - .gain (500 to 4000; 4 steps)
  - .50 Hz notch filter (ON/OFF);
- Isolated inputs;
- Automatic calibration;
- RS-232C (IEEE-488 in development) communication protocol;
- Input noise; <10 μV, p-p;
- Maximum input range; -5mV to +5mV
- Throughout rate up to 36 kHz
- ADC resolution 12 bit
- Data acquisition buffer memory up to 4x56kB;

The MC301 system has five main modules, as shown in Fig.2 (Signal Input; MUX/ADC; Control and Communications; Memory; Power Supplies), with the following blocks:

*J.C.R.L.Fernandes is at the Instituto de Engenharia de Sistemas e Computadores, Lisboa, Portugal
A.M.Aleixo is at Hospital Santa Cruz, Lisboa, Portugal



Block diagram of the MC301 system

Figure 2

.ISOLATION AMPLIFIERS - 36 Optically-coupled linear isolation amplifiers with input noise less than 4 μV, p-p (0.05-100 Hz) each one followed by AC coupled amplifiers with programmable gain (500, 1000, 2000, 4000).

.FILTERS AND S&H - 36 notch filters centered on 50 HZ each one followed by a 6th. order switched capacitor Butterworth lowpass filter and a monolithic sample and hold. The notch filters may be inserted if the mains noise is noticeable. The cutoff frequencies of the low-pass filters can be software programmed to 60, 150 or 250 Hz according to the sampling rate. The acquisition time of the sample and hold circuits is no greater than 10 μsec.

.MUX-ADC - Two microprocessor interfaced 12-bit data acquisition systems, each one with 18 channels, are connected in parallel to provide the required throughput rate. The ADC converters work under the supervision of a programmable DMA controller in rotating priority mode.

.DMA - Controls the data transfer from the MUX-ADC block to the buffer memory.

.MEMORY - The memory block has 120 kbyte, 6kbyte refer to an EPROM with the operating system and 2kbyte are allocated to scratchpad calculations. The data acquired through 36 channels is stored on a 2x56 kbyte static CMOS RAM with battery back-up for user data (buffer memory). With a sampling rate of 1000 Hz/channel the system has a capacity equivalent to 1.5 sec./channel.

.POWER SUPPLIES - The system has two power supplies with window crowbar protections. The input stages of the isolation amplifiers have an isolated power supply.

.RS-232C(IEEE-488) - Handles the system communication with the host computer or terminal using this universal communication protocols.

The system bus is IBUS [2], developed at INESC and used by several research institutions in Portugal. This bus is similar to Multibus and is compatible with the IEC mechanical standards.

The MC301 unit is a 19"x6He rack ("Eurocard Standard" DIN 41994). All the boards have 233.4x160 mm with two (a+c) 64 contacts connectors (DIN 41612), each one with a width of 5Te (25.1 mm).

The slot distribution into the rack are: Isolation Amplifiers - 4 slots (12 channels/board); Filters and S&H - 3 slots (9 channels/board); Mux-ADC - 1 slot; DMA, μP(8085), RS-232C and IEEE-488-1 slot; Memory - 2 of 4 slots available.

PLM-80 is the system programming language and assembly was only used in some critical routines. The operating system is built using a multitasking kernel, the GMT80, for applications in real time [3], [4]. The software is modular and has two basic types of tasks: the system and specific application tasks.

The system can be used as stand alone unit or linked to a host computer with permanent data storage (hard and/or flexible disk). The host computer will deal with user interface, the main data processing task, namely digital processing algorithms and data results display. The system communication with the host computer is through RS-232C (or IEEE-488)..

3. AN APPLICATION IN CARDIOLOGY

3.1. The Cardiac Mapping

Body surface potential mapping is at present the only non-invasive technique providing a spaciotemporal representation of the electrical activity of the heart [5], [6].

With a multichannel ECG data acquisition allowing simultaneous acquisition and with the evolution of digital computers and analog and digital data processing it is now possible to have a new insight of the electrophysiological heart events with large fields of applications in cardiology allowing, for example:
- The study of the sequence of activation on the cardiac surface.
- The determination of areas in which excitation and recovery currents leave and reenter the heart in the various phases of normal and abnormal ventricular activation.
- The monitoring and measurement of injury extent of acute ischemia and "infarct-size", and its changes after medical and mechanical interventions as a method of evaluation of the late prognosis of these patients.
- The possibility of using stress tests with body surface potential mapping in an attempt to improve the sensibility of these tests in identification of "high-risk" patients. New mathematical constants reflecting ventricular function can be used in order to evaluate the ventricular involvement and the opportunity to define the ventricular "mass".

Improvements in methodology and technical specifications can be expected from research groups and are an important ground of clinical investigation. These developments are far from a steady-state situation.

3.2. The MAPCARD system

A powerful application of the MC301 system was the mobile MAPCARD system for cardiac mapping that consists of:

1-The Data Acquisition System MC301, presented in section 2, with dedicated software.
2-The Host Computer, with application software and user interface.
3-The Peripherals, presently a printer and a A4 plotter.

The host computer can be any personal computer with at least the following characteristics: 256 kbyte RAM, flexible disk with 360 kbyte, graphic capabilities (640x200), communication protocol, a Pascal compiler (to avoid software conversion) and 2 interfaces, one for the printer and the other for the plotter.

The present system is built around a Sperry HT (compatible with the IBM XT), with 4 or 7 MHz clock, 20 Mbyte hard disk, 512 kbyte RAM, graphic resolution of 640x400 pixels with 16 colours, and numeric data co-processor.

The MAPCARD system is mounted on a very compact trolley that can be easily wheeled to different hospital locations. The number of control buttons was reduced to a minimum: on/off, reset and selection of channel strip. All the remaining commands are processed via a user-friendly interface protocol in the host computer.

The main specifications of the MAPCARD are:
- 36 channels acquired simultaneously (single-ended referred to the Wilson Central Terminal).
- Circular buffer occupying 112 kbyte.
- Data display of any channel 5 sec. after the end of data acquisition.
- Data inspection and validation in all channels.
- Automatic adjust of baseline in all channels
- Flexible disk data storage of each acquisition.
- Automatic and/or interactive analysis of characteristic ECG points.
- Plotter and printer facilities for all information: ECG as function of time of each channel, RMS, isopotential and isochrone coloured maps and sum values of Q, R, S, ST80.
- User-friendly interaction with the user.

The application software was written in Pascal and consists of two main blocks: the user to the MC301 system interface (INTERFAC) and the data processing and display (DPD).

The INTERFAC block is an easy to use menu driven software package designed for real time communication and control of the MC301 data acquisition, graphic display, data preprocessing and data retrieval.

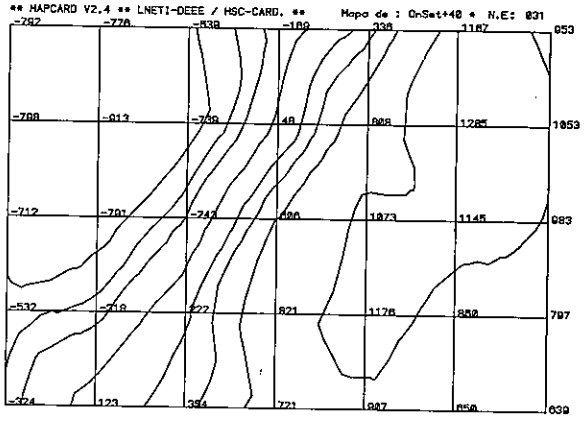
INTERFAC requires no programming skills or complex commands sequence. The display screen is divided into two windows: operator information and user options. The operator information window provides the current state and configuration of the acquisition system. The user options window provides a list of the available commands and the parameters that may be changed by the operator.

The QRS detection is made using a weighted average of the speed and acceleration of the ECG signal and baseline adjustment is performed by linear interpolation with the baseline picked in the PR interval.

The main software modules of this block are:
- COMMAND_INTERACTION
- DATA_RECEPTION
- GRAPHIC_DISPLAY
- QRS_DETECTION
- BASE_LINE_ADJUST
- PATIENT_ID
- DATA_STORAGE

The DPD block is also an easy to use menu driven software package that deals with: data processing and analysis, ECG waveforms drawing, clinical documentation, 2D (as shown in Fig. 3) and 3D isopotential and isochrone surface map plotting.

The data processing routines are responsible for: digital filtering to remove line frequency interferences and base line drifts; automatic and/or interactive determination of characteristic ECG points; sum values of Q, R, S wave amplitudes and ST segment elevation; RMS (Root Mean Square) computation.



Isopotential map 40 msec. after QRS onset

Figure 3

The digital filter is a linear phase integer--coefficient bandpass filter.

The graphic display software modules developed for data analysis and clinical documentation provide: the display of the ECG waveforms of all channels, in groups of seven; the ECG waveform display of any channel with a cursor for time and amplitude measurements; zooming of any user defined area of the ECG waveform.

The main software modules of DPD block are:
- FILTER
- QRS_IDENTIFICATION
- SUM_WAVES
- ECG_DRAWING
- ZOOM
- ISO_MAP
- DOCUMENTATION

3.3. Method and results

The MAPCARD system has been used with 35 channels in a matrix (5x7). The 7 vertical columns (1-7) are self-adhesive strips of 5 electrodes (A-E). The inter-electrodes distance is 3.5 cm positioned over the chest with anatomical references: A1 over the second right intercostal space and strips 6 and 7 over anterior and left mid axillary line, respectively.

All recordings of body surface mapping are carried out quickly and can be repeated without causing disturbance to the patient (the whole process lasts no more than five minutes).

All the mapping area was divided in 5 sub-areas according to the projection of coronary artery supply: anterior, related to left anterior descending artery; high anterior, with left main artery; lateral, with left circumflex artery; postero-basal, with right coronary artery; apical, with the last one if the system is right dominant, or with the left circumflex in a left dominant system.

Real tests started at the hospital in November 1985, and 27 normal individuals have been studied [7]. The total and sub-areas sum values of R, S, Q waves and ST have been evaluated and isopotential maps from QRS onset have been drawn in order to determine normal values and to be compared with other abnormal groups. For non invasive assessment of left ventricular disfunction 25 post myocardial infarction patients were studied [8].

Although preliminary, due to the size of the population studied, the existing results are in accordance with other reported works [9], [10].

4. CONCLUSIONS

A microprocessor based multiple channel data acquisition system (MC301) was designed for a wide range of measurement applications. An example of its application in cardiac mapping (MAPCARD system) was presented.

The already available results clearly suggest that an interesting and powerful diagnostic method in cardiology is being consistently developed using a simple and economic "PC" based equipment.

REFERENCES

[1] Especial,N.F.S.;Fernandes,J.C.R.L.;Almeida, F.J.S., A Multiple Channel Data Acquisition System, in: CEAUTL,UTL (ed), 1º Simpósio de Electrónica das Telecomunicações (CEAUTL, 1984) pp. 124-126, (in Portuguese).

[2] Verissimo,P.J.E.;Arroz,G.S.;IBUS- A Microcomputer Bus, in:ENDIEL,1982(in Portuguese).

[3] Gomes,J.D.;Machado,M.;Pinto,J.V.; GMT80 - A Real Time Multitasking Kernel,in:CEAUTL,UTL (ed),1º Simpósiode Electrónica das Telecomunicações (CEAUTL, 1984) pp.224 (in Portuguese).

[4] Almeida,F.J.S.; Especial,N.F.S.; Fernandes, J.C.R.L.; Data Acquisition System Software--A GMT80 Application, in:CEAUTL,UTL (ed),2º Simpósio de Electrónica das Telecomunicações (CEAUTL, 1986) in print (in Portuguese).

[5] Guise,J.;Guardo,R.;Lafortune,M.; Technical Advances in Body Surface Potential Mapping Instrumentation, in: IEEE 1980 Frontiers of Engineering in Health Care, pp. 196-199.

[6] Especial,N.F.S.; Almeida,F.J.S.; Fernandes, J.C.R.L.;Aleixo,A.M.; Gil,V.; Seabra-Gomes, R.; Cardiac Mapping Computerized Data Acquisition System, in print (in Portuguese).

[7] Aleixo,A.;Gil,V.;Seabra-Gomes,R.; Especial, N.F.S.; Almeida,F.J.S.; Fernandes,J.C.R.L.; Amado,J.C.; Computerized Precordial Mapping-- Analysis of Normal Values, X Congresso Luso-Espanhol de Cardiologia, Lisboa, 1986, in print (in Portuguese).

[8] Aleixo,A.;Gil,V.;Adão, M.;Real, T.; Seabra--Gomes,R.;Especial,N.F.S.; Almeida,F.J.S.; Fernandes,J.C.R.L.; Amado,J.C.; Non Invasive Assessment of Left Ventricular Disfunction Through Computerized Body Surface Potential Mapping, Paper submitted to 13th International Congress on Electrocardiology, Washington, D.C., September 1986.

[9] Ikeda,K.;Kubota,I.; Igarashi,A.; Yamaki,M.; Tsuiki,K.;Yasui,S.; Detection of Local Abnormalities in Ventricular Activation Sequence by Body Surface Isochrone Mapping in Patients with Previous Myocardial Infarction, Circulation 72, No.4 (1985), pp. 801-809.

[10] Miller III,W.T.; Spach,M.S.; Warren,R.B.; Total Boby Surface Potential Mapping During Exercise: QRS-T-Wave Changes in Normal Young Adults, Circulation 62, No.3 (1980), pp. 632-645.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

QUANTITATIVE ANALYSIS OF MAGNETIC RESONANCE TIME DOMAIN SIGNALS

H. BARKHUIJSEN, R. de BEER, W.M.M.J. BOVÉE, A.M. van de BRINK,
A.C. DROGENDIJK, D. van ORMONDT, and J.W. van der VEEN

Dept. of Applied Physics, Delft University of Technology,
P.O.Box 5046, 2600 GA Delft, The Netherlands

A magnetic resonance time domain signal is often made up of a limited number of exponentially decaying sinusoids plus white noise. Traditionally, quantitative analysis of the signal is carried out in the frequency domain, after applying FFT in conjunction with a time window. We show that quantitative analysis directly in the time domain is feasible, and in fact yields several advantages. Various methods are applied and compared.

1. INTRODUCTION

Quantification of Magnetic Resonance (MR) time domain signals is important for analytical work. A MR signal comprises a number of damped sinusoids plus white noise, and up to now the usual way of analysis is to first convert the data to the frequency domain and then determine the area under the resulting spectral peaks by integration. This method entails problems when spectral peaks overlap. A possible solution to this is to fit appropriate model functions to the spectrum using a least squares (LS) method [1]. However, two problems remain when analysing in the frequency domain: 1. Often the initial data points are disturbed by various strong side effects. Truncation of these data points duly removes the related, unwanted part of the spectrum, but in turn causes a distortion of the base line, which still hampers the model fitting procedure. 2. To the best of our knowledge, the available frequency domain LS fitting procedures are nonlinear and iterative.

When the analysis is carried out directly in the time domain the latter problems need not arise. This is because omitting initial data points does not affect the model fitting as such, and because noniterative LS fitting procedures have recently become available, at least for the case that the damping of the sinusoids is governed by simple exponentials. Important advantages of these procedures are: Starting values of the parameters are not required. 2. The aspects of convergence is nonexistent. 3. The computing time is reduced. In the present study we apply two noniterative methods, one based on Linear Prediction [2], and the other on the State Space formalism [3].

In spite of the advantages of noniterative methods, various cases may still demand an iterative, nonlinear LS time domain fit [4,5]. Here we apply a special form of such a procedure, namely the Variable Projection method [6].

2. THEORY

The signal in question belongs to a class that is characterized by approximate exponential damping as a function of time t, i.e. according to exp(-bt) where b is a constant. b may be different for each sinusoid in the signal. With the usual SNR, possible deviations from this functional behaviour are difficult to discern. Consequently, the sampled signal, x_n, n = 0, ..., N-1, can be adequately represented by the model function

$$x_n = \sum_{k=1}^{K} c_k \exp[(-b_k+i\omega_k)n\Delta t + i\phi_k] + w_n \qquad (1)$$

in which Δt is the sample interval, c_k, b_k, ω_k, and ϕ_k, k=1, ...,K, are the amplitudes, damping factors, (angular) frequencies, and phases respectively, and w_n is white noise. Note that the model function depends linearly on the complex amplitudes, defined by $c_k' \equiv c_k \exp(i\phi_k)$, k=1, ...,K, but nonlinearly on the damping factors and (angular) frequencies. The latter can be compounded into the quantities $z_k \equiv \exp[(-b_k+i\omega_k)\Delta t]$, k=1, ...,K, the so-called signal roots. As a consequence of the nonlinear dependence on the damping factors and frequencies, these model parameters cannot be estimated by means of a simple, linear, noniterative, LS fitting procedure.

In [2] the nonlinear dependence on the damping factors and frequencies is circumvented by first fitting a backward Linear Prediction (LP) equation of order M (0,5N<M<0.75N) to the data. Subsequently, the wanted quantities z_k are found by rooting the associated LP polynomial of order M, and selecting the K largest roots (signal roots). The latter reside outside the unit circle owing to predicting backward. The M-K remaining (extraneous) roots are evenly distributed inside the unit circle, so long as the SNR is reasonable [2].

The fitting of the LP equation mentioned above entails Singular Value Decomposition (SVD) of an (N-M)xM matrix X (data matrix), whose elements

are $X_{ij} = x_{i+j-1}$. Advantages of using SVD are: 1. The number of significant singular values equals the number of sinusoids in the signal, i.e. K in Eq. (1). 2. The roots of the LP polynomial can be improved by truncating the insignificant singular values. 3. A noise correction can be applied to the significant singular values. Once the signal roots are known, the complex amplitudes defined above can be estimated by means of fitting the equation.

$$\hat{x}_n = \sum_{k=1}^{K} c_k' z_k^n \quad , \; n=0,\ldots,N-1, \qquad (2)$$

to the data. This completes the estimation of the model parameters. We indicate this method here by LPSVD.

The second noniterative method [3] investigated by us has been construed in the context of the so-called State Space formalism, and is also based on the validity of Eq. (1). It amounts to fitting a Hankel matrix to a data matrix of size (N-M+1)xM, where we use 0.5N<M<0.75N. As with LPSVD, the procedure makes use of SVD to obtain the number of significant sinusoids, K. However, the polynomial rooting and subsequent root selection, taking place with LPSVD, is avoided. After estimating the signal roots, the complex amplitudes can be estimated in the same way as with LPSVD. We indicate this method by HSVD, where H stands for Hankel.

The iterative, nonlinear LS fitting method chosen by us, the Variable Projection (VP) method [6], possesses the special feature that it eliminates the (complex) amplitudes from the LS equations. The resulting LS equations, which contain only the nonlinear parameters, are then solved by an iterative, nonlinear procedure requiring starting values. Finally, the amplitudes are estimated by a noniterative, linear LS procedure, similar to LPSVD and HSVD. It follows that VP requires no starting values for the complex amplitudes. In the present contribution we apply VP only in conjunction with the model function of Eq. (1). Nevertheless, even within this limitation it may yield improved results [4].

3. RESULTS AND DISCUSSION

The methods described in the previous section have been applied to an in vivo NMR free induction decay measurement of ^{31}P in a tumour in an anaesthetized mouse. The sampled signal comprised 768 complex data points, the first 128 of which are shown in Fig. 1. A large spike, caused by side effects, is present at the beginning of both the real and the imaginary part. The cosine FFT spectrum of the signal is shown in Fig. 2a. The spectrum of the spike has been effectively removed by omitting the first three data points, but this measure in turn caused a distorted ('rolling') base line. By eye we distinguish eight significant peaks in the spectrum of Fig. 2a. These are indicated by numbers. Clearly, peaks 1 and 2 overlap, as do peaks 5, 6, and 7. This overlap, and the rolling baseline hamper

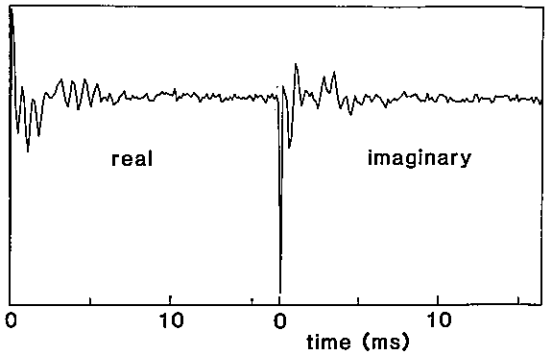


Fig. 1. Complex MR time domain signal.

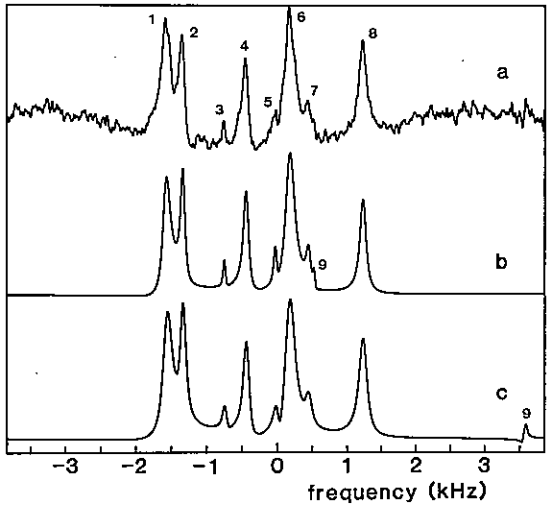


Fig. 2. Cosine FFT of time domain signal minus initial three data points (a), of LPSVD time domain fit (b), of HSVD time domain fit (c).

the determination of the individual peak intensities through analysis in the frequency domain.

Now we turn to fitting in the time domain. For LPSVD we used data points n=3,...,127, for HSVD n=4,...,127. The shapes of the data matrices were chosen approximately square (M=63 or 62), or distinctly rectangular (M=88). In all cases the singular values indicated the presence of nine significant sinusoids. To be on the safe side, one extra singular value was taken into account with both LPSVD and HSVD (K=10) for producing the numbers in Table 1 (listed for $k \leqslant 8$). Each parameter is accompanied by the corresponding Cramér-Rao lower bound, and the amplitudes and phases are valid for t=0. It turned out that the eight peaks indicated in Fig.2a were correctly identified. All amplitudes for k>8 (not shown) are small, as was to be expected. An exception to this rule occured for HSVD, M=62, where

TABLE 1

Parameters of the model function of Eq. (1), estimated with two noniterative methods (LPSVD and HSVD), and an iterative method (VP), for in vivo NMR on a mouse. $\nu_k = \omega_k/2\pi$. See text.

Method	k	M	ν_k (kHz)	b_k (kHz)	c_k (a.u.)	ϕ_k (deg)
LPSVD	1	88	-1.5711±0.0051	0.410±0.032	2.46±0.15	30± 4
LPSVD	1	63	-1.5713±0.0053	0.417±0.034	2.50±0.16	27± 4
VP/LP	1		-1.5719±0.0050	0.439±0.031	2.53±0.13	22± 3
HSVD	1	88	-1.5558±0.0059	0.453±0.037	2.82±0.19	11± 4
HSVD	1	62	-1.5610±0.0102	0.475±0.064	2.91±0.34	21± 7
VP/H	1		-1.5616±0.0059	0.469±0.037	2.87±0.15	16± 4
LPSVD	2	88	-1.3256±0.0035	0.196±0.022	1.23±0.11	0± 5
LPSVD	2	63	-1.3245±0.0036	0.188±0.023	1.19±0.11	-4± 5
VP/LP	2		-1.3254±0.0031	0.203±0.020	1.23±0.09	-8± 4
HSVD	2	88	-1.3281±0.0042	0.237±0.026	1.52±0.14	0± 5
HSVD	2	62	-1.3268±0.0068	0.220±0.043	1.41±0.23	4± 9
VP/H	2		-1.3271±0.0037	0.229±0.023	1.45±0.11	0± 5
LPSVD	3	88	-0.7328±0.0114	0.121±0.071	0.20±0.08	-15±22
LPSVD	3	63	-0.7355±0.0090	0.046±0.057	0.14±0.06	-15±25
VP/LP	3		-0.7378±0.0065	0.048±0.041	0.15±0.05	-13±18
HSVD	3	88	-0.7352±0.0099	0.057±0.062	0.14±0.07	-26±26
HSVD	3	62	-0.7262±0.0210	0.140±0.132	0.23±0.16	-39±38
VP/H	3		-0.7330±0.0072	0.033±0.046	0.13±0.05	-29±22
LPSVD	4	88	-0.4298±0.0043	0.265±0.027	1.48±0.12	-9± 5
LPSVD	4	63	-0.4283±0.0043	0.253±0.027	1.42±0.12	-16± 5
VP/LP	4		-0.4300±0.0037	0.268±0.023	1.47±0.09	-15± 4
HSVD	4	88	-0.4282±0.0045	0.239±0.028	1.28±0.12	-18± 5
HSVD	4	62	-0.4265±0.0081	0.256±0.051	1.41±0.24	-17±10
VP/H	4		-0.4283±0.0039	0.236±0.024	1.28±0.09	-17± 4
LPSVD	5	88	-0.0161±0.0090	0.111±0.056	0.27±0.10	-31±20
LPSVD	5	63	0.0060±0.0087	0.011±0.054	0.14±0.07	-89±29
VP/LP	5		0.0023±0.0064	0.087±0.041	0.26±0.07	-75±17
HSVD	5	88	0.0078±0.0108	0.114±0.068	0.27±0.12	-79±25
HSVD	5	62	0.0130±0.0182	0.015±0.114	0.12±0.13	-109±64
VP/H	5		-0.0040±0.0071	0.065±0.045	0.20±0.10	-56±20
LPSVD	6	88	0.1794±0.0061	0.490±0.038	3.59±0.28	6± 4
LPSVD	6	63	0.2007±0.0111	0.625±0.070	4.41±0.35	-9± 5
VP/LP	6		0.1839±0.0068	0.568±0.043	4.02±0.26	0± 4
HSVD	6	88	0.1693±0.0068	0.514±0.043	3.73±0.30	10± 5
HSVD	6	62	0.1748±0.0218	0.341±0.137	1.88±1.18	10±36
VP/H	6		0.1735±0.0063	0.508±0.040	3.73±0.20	10± 4
LPSVD	7	88	0.4574±0.0257	0.315±0.162	0.62±0.31	-21±29
LPSVD	7	63	0.4389±0.0169	0.323±0.106	0.71±0.23	13±19
VP/LP	7		0.4490±0.0201	0.322±0.127	0.55±0.22	-6±22
HSVD	7	88	0.4428±0.0176	0.262±0.111	0.42±0.15	5±21
HSVD	7	62	0.4016±0.0459	0.226±0.288	0.33±0.51	89±88
VP/H	7		0.4412±0.0180	0.220±0.082	0.37±0.10	9±18
LPSVD	8	88	1.2359±0.0049	0.333±0.031	1.67±0.11	-7± 4
LPSVD	8	63	1.2364±0.0051	0.334±0.032	1.69±0.12	-11± 4
VP/LP	8		1.2363±0.0045	0.342±0.028	1.69±0.09	-6± 3
HSVD	8	88	1.2330±0.0058	0.400±0.036	2.03±0.14	-8± 4
HSVD	8	62	1.2344±0.0101	0.411±0.063	2.11±0.25	-5± 7
VP/H	8		1.2319±0.0052	0.386±0.033	1.98±0.10	-6± 4

sinusoid k=6 was decomposed into two components, the parameters of the other component (k>8) being ν=0.27±0.13, b=1.7±0.8, c=4.9±1.5, ϕ=-14±18. This decomposition was not discernible in the FFT of the time domain fit; moreover, the time domain fit yielded a smaller residue norm than the corresponding LPSVD fit, where no decomposition took place. The FFT's of the LPSVD and HSVD time domain fits for nine singular values (K=9) are shown in Figs. 2b and 2c respectively. Note that the sinusoids k>8, selected by LPSVD and HSVD, differ. This outcome is not surprising in view of the noise features seen in Fig. 2a.

The VP method was applied using the signal roots yielded by LPSVD and HSVD (VP/LP and VP/H respectively) for nine singular values (K=9), as starting values. It turned out that the parameters resulting from this iterative method agreed with those of the respective noniterative methods. This indicates that VP does not seem to favour either LPSVD or HSVD. In fact, the different sinusoids for k>8 of the two methods were both taken up and fitted by VP, which in turn is partly responsible for the differences between VP/LP and VP/H for k<9.

4. CONCLUSIONS

1. The noniterative methods LPSVD and HSVD yield reasonable estimates of the model parameters, including the number of sinusoids.
2. The iterative nonlinear VP method performs well when given the damping factors and frequencies of the noniterative methods as starting values. This opens the way to more sophisticated applications, for instance the incorporation of prior knowledge about certain sinusoids and the use of an improved model function.

ACKNOWLEDGEMENTS

The authors thank Dr. A. van der Bos for calling their attention to the Variable Projection method, and making a computer program available. The same person and Dr. P.M.T. Broersen are thanked for useful discussions. This work was performed as part of the research program of the "Stichting voor Fundamenteel Onderzoek der Materie" (FOM), with financial support from the "Nederlandse Organisatie voor Wetenschappelijk Onderzoek" (ZWO).

REFERENCES

[1] Dumoulin, C.L. and Levy, G.C., Bull. Magn. Reson. 6 (1984) 47.
[2] Kumaresan, R. and Tufts, D.W., IEEE Trans. Acoust., Speech, Signal Processing ASSP-30 (1982) 833.
[3] Kung, S.Y., Arun, K.S., and Bhaskar Rao, D.V., J. Opt. Soc. Am. 73 (1983) 1799.
[4] Parthasarathy, S. and Tufts, D.W., Proc. IEEE 73 (1985) 1528.
[5] Barkhuysen, H., Beer, R. de, Ormondt,D. van, J. Magn. Reson. 67 (1986) 371.
[6] Golub, G.H. and Pereyra, V., SIAM J. Numer. Anal. 10 (1973) 413.
[7] Barkhuysen, H., Beer, R. de, Bovée, W.M.M.J. and Ormondt, D. van, J. Magn. Reson. 61 (1985) 465.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

RECOGNIZING MUSCLES BY AUTOMATIC IMAGE ANALYSIS

A. DENIZON * , J.P. IMBAUD ** , A. LACOURT *

* : Institut National de la Recherche Agronomique
Station de Recherches sur la Viande
THEIX - 63122 CEYRAT - FRANCE

** : Université de Clermont-Ferrand II
Laboratoire d'Electronique
B.P. 45 - 63170 AUBIERE - FRANCE

The study of connective tissue in bovine muscles is performed in this paper. To characterize this network by image analysis, two algorithms are implemented. The first one binarizes the initial image and the second one, here described, determines eleven parameters relative to this network. Three different muscles are studied and results are discussed.

1. INTRODUCTION

Technological and economic changes which are currently transforming meat production channels have needed to better understanding of parameters that play an important role in meat tenderness. Tenderness is indeed a determining factor in estimating market values.

Biological factors such as age, sex, fattening state, are all involved in determining tenderness as are the *post-mortem* transformations which muscles undergo during the different technological stages where muscle is changed into meat.

The texture of a muscle is composed of two main components : muscular fibers and connective tissue. Connective tissue, frame of the muscle, surrounds each muscular fiber and then establishes a network which we will try to evaluate with bidimensional image analyses.

DUMONT **[4]** and later SALE **[5]** stressed the important relationship between connective tissue distribution inside the muscle and tenderness. The aim here is to point out the role of connective tissue in evaluating the parameters which allow every type of muscle to be characterized, by using image analysis.

2. MATERIAL

The three bovine muscles studied are the *Pectoralis Profundis* (PP), the *Longissimus Dorsi* (LD) and the *Masseter* (M) *(Figure I)*.

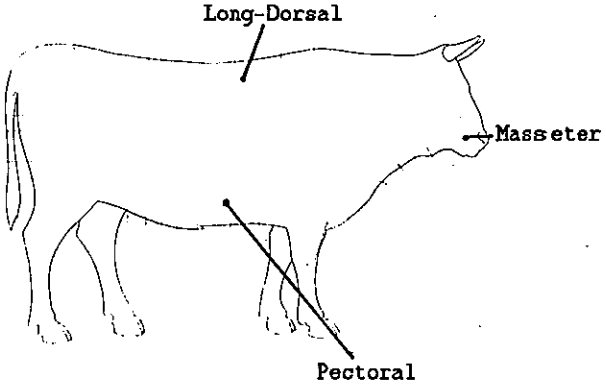


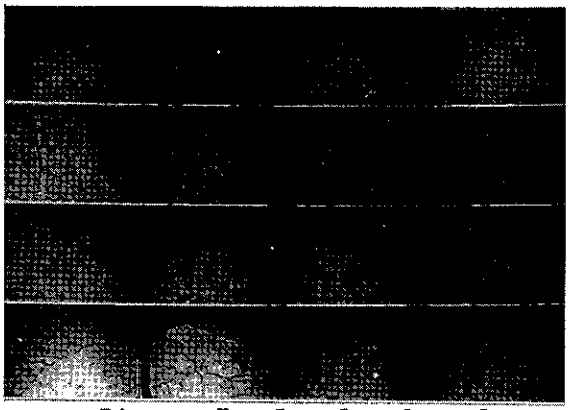
Figure I : Localisation of the three muscles studied

The LD, the most tender of these three muscles, is situated under the ribs and is used for grilling. The M muscle is near the jaws. This is a muscle that seems to have more connective tissue than the other two. The PP of medium tenderness, is from the chest of the animal.

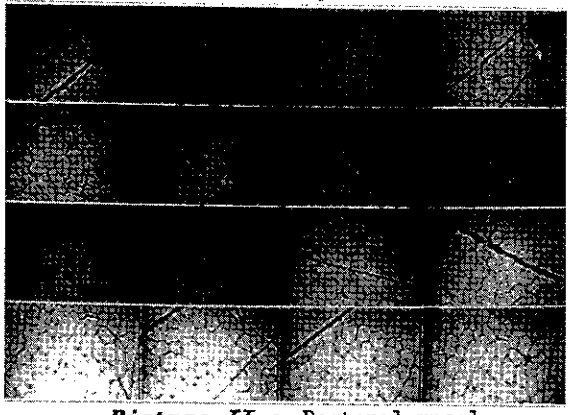
For every type of muscle, four samples were taken, 1 cm square, in the direction of fibers and frozen.

Histological cuts, 8µm thick, were obtained with a cryotome. Connective tissue was selectively colored by the periodic acid Schiff technique after amylase and β-glucosidase actions, in order to eliminate all traces of glycogen from the myofibrillar structures.

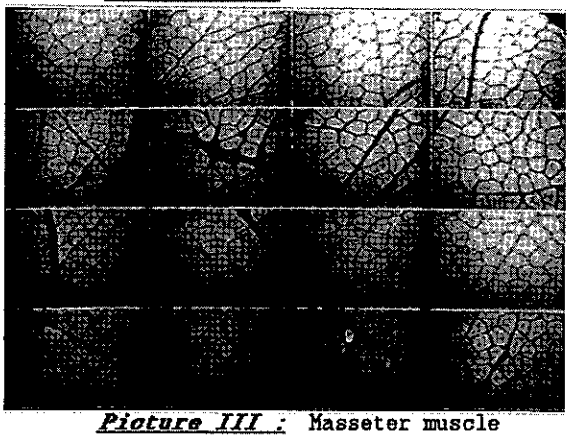
The surface of studied cuttings was close to 0.2 μm^2, at 100 microscopic enlargement. Each cut were photographed from four different places, for a total of 16 shots per muscle *(Pictures I, II and III)*.



Picture I : Long-Dorsal muscle



Picture II : Pectoral muscle



Picture III : Masseter muscle

The photos were numbered in rows of 512, with each row being made up of 512 pixels. The pixels were coded on 8 bits, i.e. they could assume 256 values, from black to white. This work has been realized with the image processing system PRIVE [3].

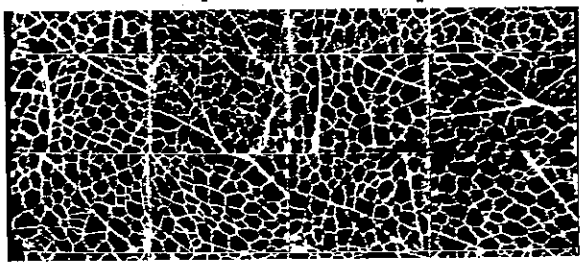
3. DESCRIPTION OF THE STUDY

3.1. Binarisation of the image

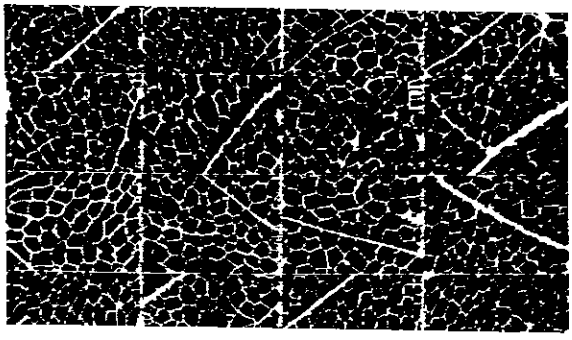
In order to extract the connective tissue edges from the initial image, an original algorithm described in details in [1] and [2] has been implemented. It main steps are summarized as following:

* noise suppression in the image,
* extraction of edges by a linear method applied to rows, then to columns and reunion of all the results of both extractions,
* suppression of artifacts,
* skeleton obtention and closing of unjoined edges after investigation,
* internal smoothing of the edge of each fiber.

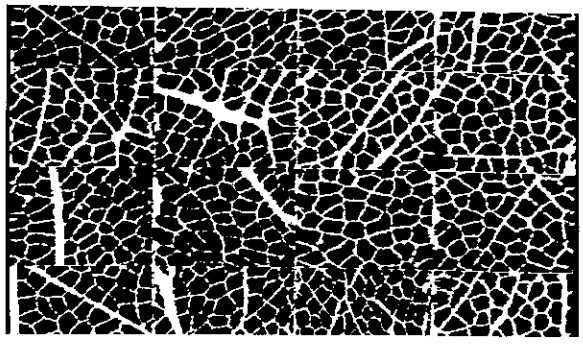
The results of the edge extraction on the sixteen images of each muscle are illustrated by the next three pictures :



Picture IV : Edges extraction: LD muscle



Picture V : Edges extraction : PP muscle



Picture VI : Edges extraction : M muscle

3.2. OBTAINING THE PARAMETERS

For each image,a set of

measurements, presented below, was implemented. A detailed explanation is related in **[2]**.

* Percentage of connective tissue with respect to the whole image : T

This T percentage is calculated by counting the number of points which are on edges, and dividing by the whole number of pixels in the image :

$$T = \frac{N}{512^2}$$

To be significant, this number T is applied only to the images that show thin edges. Images including large contours of perimysium have not been taken into consideration because the percentage of connective tissue is too high, so thus falsifie calculations of the mean value.

* Middle surface of fibers : S and number of fibers : NF

To do this, all fibers in contact with the sides of the image, which can have a wrong surface, have been excluded. The number of fibers is expressed with respect to the whole surface of the cut. Knowing the real surface of the cut : 0,45 mm wide and 0,246 mm high, that is 0,11 mm^2, we give the number of fibers for 1 mm^2.

The method used is the individual detection of fibers by a specific algorithm and the filling of its shape in order to know its surface expressed in number of pixels and in μm^2.

* Middle perimeter of fibers : P

For each of the fibers indexed above, a following of its internal contour using FREEMAN's coding *(Figure II)* was designed.

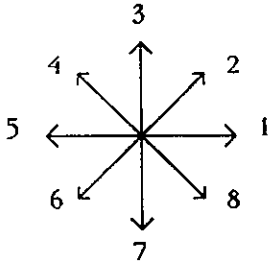
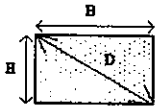


Figure II : Coding of displacements in an image with a square frame according to FREEMAN

To calculate the perimeter P, according to the value of the displacement during the contour following, we add to the variable P *(Figure III)* :

* 1 if the displacement is equal to 1 or 5,
* 1,8 if the displacement is equal to 3 or 7,
* 2,06 if the displacement is even (2,4,6 or 8).

The real value of the perimeter is obtained multiplying the value P (in a relative scale) by 0,5 µm.



DIMENSION	RELATIVE SCALE	REAL SCALE (µm)
Height H	1	0,5
Breadth B	1,8	0,9
Diagonal D	2,06	1,03
Surface S	1,8	0,45 μm^2

Figure III: Relative and real scales

* Ratio square perimeter on surface : P^2 / S

This ratio is calculated for the images in which fibers connected to the sides have been eliminated.

* FERET's diameter of each fiber : D and rate of elongation : E

For each fiber that was not cut by the sides of the image, FERET's diameter was measured ; that is the diameter of the circle minimum including the whole fiber.

The elongation rate obtained by dividing the fiber's FERET diameter by the breadth measured at the middle of the diameter and perpendiculary to it, were calculated as well.

* Number of nodes : NN and branches : NB

Logically, a branch is a continous part of connective tissue which does not tear up into several directions. So, a node is the geometrical position from where these subdivisions start *(Figure IV)*.

We express these numbers for a surface of 1 mm^2.

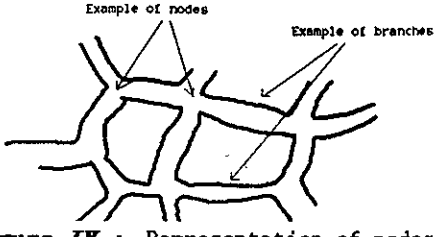


Figure IV : Representation of nodes and branches

* Middle length of branches : L

Knowing for each branch, the two nodes which are connected at each extremity and their coordinates (x_1,y_1) and (x_2,y_2), the length of this branch is :

$$L = \sqrt{(x_1 - x_2)^2 + 1,8^2 \cdot (y_1 - y_2)^2}$$

* Middle thickness of branches: EP

The middle thickness is calculated for each branch ; adding the breadths EPi in each pixel of the skeleton of the

branch, and dividing by the number of pixels of its skeleton a :

$$EP = \sum_{i=1}^{\alpha} \frac{EPi}{\alpha}$$

This parameter was used only on images in which the presence of more important connective threads from perimysium was not discerned.

4. RESULTS AND DISCUSSION

The following table gives the results of the measurements taken in the 48 pictures of all three muscles dealt with.

MUSCLE	PECTORAL	LONG-DORSAL	MASSETER
Percentage of connective tissue T Standard deviation	20,5 % 1,4 %	28,5 % 1,9 %	28,7 % 3,2 %
Number of fibers NF (for 1 mm²) Standard deviation	436 64	555 91	336 27
Surface of fiber S(um²) Standard deviation	1489 232	1178 149	1657 214
Perimeter P (um) Standard deviation	134 13	123 20	149 13
Ratio P2 / S Standard deviation	26,2 2,5	33,0 10,5	26,3 2,4
FERET's diameter D(um) Standard deviation	46,5 6,0	37,0 2,5	46,0 3,0
Elongation rate E Standard deviation	2,52 0,52	2,41 0,60	1,93 0,2?
Number of nodes NN Standard deviation	1655 255	2618 864	1791 318
Number of branches NB Standard deviation	2182 373	3318 836	2409 473
Length of branches L(um) Standard deviation	18,2 5,1	13,8 7,9	15,8 2,0
Breadth of branches P(um) Standard deviation	3,21 0,56	4,32 0,57	5,05 1,02

Biochemical dosages of PP muscle show usually a collagen rate twice as large than in LD. Then we can affirm that the surface percen- tage of connective tissue of LD is overes-estimated in our study with respect to its real value.

In M muscle pictures, the percentage of connective tissue includes collagen but also the extra-cellular volume.

Consequently, the values shown by variables S, P, P^2/S and EP are also wrong.

Theoretically, variables D and E don't reflect the muscle characteristics but point out the histological cut angle.

In our study, fibers number NF can discriminate between the three type of muscles.

The middle length of branches is too variable for concluding any significant results.

However, the two parameters NN and NB seem interesting. For one muscular fiber, we obtain :

	PP muscle	LD muscle	M muscle
Branche number	5,00	5,98	7,17
Node number	3,80	4,72	5,33

These variables which describe accurately the structure of muscle tissue seem to characterize each muscle, and that independently of valuation of the collagen concentration.

5. CONCLUSION

We have seen that the study of the eleven parameters can differenciate the three muscles that we examine. This method gets interesting results with regard to the comparison of connective networks.

In order to improve our results, we shall have to use other methods :

* biochemical dosage to calculate the percentage of connective tissue,

* electronic microscopy to measure the thickness of connective walls,

and we can hope to get a correction coefficient for variables needing it.

If these methods are not efficient the exploitation of the parameters which characterize the connective tissue without respect of rate collagen, as NF,NN, NB coulb be a good solution to get relationships with muscle texture.

REFERENCES

[1] : DENIZON - BONTON - IMBAUD - SALE
"Description d'outils logiciels pour l'analyse du tissu musculaire"
5□ Congrès AFCET - Grenoble - novembre 1985

[2] : DENIZON
"Reconnaissance automatique de muscles par traitement d'images"
Thèse D.I. - Université de CLERMONT-FERRAND II - mai 1986

[3] : DERUTIN
"Contribution à la réalisation d'un système de traitement numérique d'images vidéo"
Thèse D.I.- Université de Clermont-Ferrand II - décembre 1982

[4] : DUMONT - LEFEVRE - SCHMITT - BARBU
"Application de méthodes d'analyse multidimensionnelle à la différenciation des muscles sur la base de leur trame conjonctive"
10 th european meeting of statisticians - LEUVEN (Belgique) - 22,26 août 1977

[5] : SALE
"Que peut-on attendre des méthodes instrumentales d'évaluation de la tendreté de la viande ?"
Bull. Tech. CRZV - THEIX - INRA - 1980 - n□ 40 - p. 19,26

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

T.11

Identification of sounding thermometer by Prony's method.

de BRUCQ Denis, FORTIER Natalie, GOUAULT Jean.

LACIS-ITEPEA U.E.R. des Sciences BP 67
76130 MONT-SAINT-AIGNAN FRANCE

SUMMARY

For medical applications, we developped a thermometer to measure the skin temperature of a patients's forehead. We put in contact sounder and skin. The identification of parameters and in particular the skin's temperature comes from the evolution cures. A stationary linear model is not sufficient to describe the phenomenon : we introduce time dependent coefficients. We computed the model's parameters by successive approximations.

1. Introduction

Our laboratory achieves (cf [3]) a sounding thermometer for medical applications. It measures the skin temperature and its main purpose is to provide the temperatures of the left and right parts of the forehead. Bad irrigation implies low temperatures.

The initial temperature for the skin is $\theta_{1,0}$ and for the sounder is $\theta_{2,0}$. Then we put sounder and skin in contact ; the evolution curves, of a measured temperature θ_1 shows a minima. From the minima of two experimental curves, it is posible to deduce the skin's temperature (cf [4]). So it is essential to obtain the position of the minima.

From a simplified model, the heat transfer equations are with self-explained notations :

$$C_1 \dot{\theta}_1(t) = h(\theta_2-\theta_1)+h_1(\theta_0-\theta_1)$$
$$C_2 \dot{\theta}_2(t) = h(\theta_1-\theta_2)+h_2(\theta_r-\theta_2) \quad (1)$$

Through mathematical transformations, we obtain the canonical form :

$$\begin{pmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{pmatrix} = H \begin{pmatrix}x_1(t)\\ x_2(t)\end{pmatrix} \quad (2)$$ state equations

$$y(t) = x_2(t) + w(t) \quad (3)$$ observation equation.

Here the white noise w is additive.
Sampling is necessary for achieving identification.
In specified period $t=k\delta t$, eliminating $x_1(t)$, we deduce :

$$x_2(k)+a_1 x_2(k-1)+a_2 x_2(k-2)=0 \quad (4)$$

$$y(k) = x_2(k)+w(k). \quad (5)$$

Prony's technique starts here (cf [2]).
The solutions are in the vectorial space :

$$x(k)=\beta_1 z_1^k+\beta_2 z_2^k \quad (6)$$

where z_1 and z_2 are the solutions of the polynomial equation :

$$z^2 + a_1 z + a_2 = 0 \quad (7)$$

If Γ is the autocorrelation of the observation process y then

(8) $(a_0, a_1, a_2)\ \Gamma = \sigma^2\ (a_0, a_1, a_2)$

where $a_0 = 1$ and σ^2 is the variance of the noise.
From patient to patient, only slight differences appear. Perturbation method increases precision and velocity. We study the theoritical problem to implement successive approximation techniques before obtaining an improved sounding thermometer. We approximate the matrix H changing H by $H+S(w)+R(w,t)$ where $S(w)$ takes account of the random fluctuations of the physical phenomena from one measurement to another and where $R(w,t)$ describes the temporal dependance of the physical phenomena. To use the 250 data to reconstruct the initial temperatures, we need a time dependent model.

2. SOLUTIONS OF THE PERTURBED EQUATION

We consider the evolution equation

$$\frac{dx}{dt}(t) = (H+P)\ x(t)$$

here P is small in comparison with H constant that is to say we suppose $HP \simeq PH$, then

Proposition - 1 : Let H a constant matrix and P such that $HP=PH$ then if $U(t,t_0)$ is the solution of

(1) $\frac{\partial V}{\partial t}(t, t_0) = H\ U(t,t_0)$ with $U(t_0,t_0)=1$ and if $V(t,t_0)$ is the solution of

(2) $\frac{\partial V}{\partial t}(t,t_0) = P\ V(t,t_0)$ with $V(t_0,t_0)=1$ then

(3) $x(t) := U(t,t_0)\ V(t,t_0)\ x(t_0)$ is solution of

(4) $\frac{dx}{dt}(t) = (H+P)\ w(t)$

Proof :

$$\frac{dx}{dt}(t) = \left(\frac{\partial U}{\partial t}(t,t_0)V(t,t_0) + U(t,t_0)\frac{\partial V}{\partial t}(t,t_0)\right)x(t_0)$$

$$= (H\ U(t,t_0)V(t,t_0) + U(t,t_0)P\ V(t,t_0))x(t_0).$$

As

$$U(t,t_0) = \sum_{n=0}^{\infty} \frac{(t-t_0)}{n!} H^n = \exp H(t-t_0)$$ and

$HP=PH$, we obtain $UP=PU$ then

$$\frac{dx}{dt}(t) = (H+P)\ U(t,t_0)\ V(t,t_0)x(t_0). \blacksquare$$

Corollary - 2 : For H and S constant matrices such that
$HS=SH$, the solution $x(t)$ of

$$\frac{dx}{dt}(t) = (H+S)\ x(t)$$ is

$$x(t) = \exp H(t-t_0)\ x \exp S(t-t_0)\ x(t_0).$$

In pratice when, S is random and small the hypothesis

$HS = SH$ is natural and gives

$$\exp H(t-t_0)\ S = S \exp H(t-t_0).$$

We describe the time dependent phenomena with the help of a rationnal matrix function :
$R(t) = N(t)\ D(t)^{-1}$.
Decomposition in fractionnal fractions gives

$$(5)\ R(t) = M_0 + M_1 t + \ldots + M_n t^n + \sum_{i,j}{}' \frac{B(i,j)}{(1+c(i,j)t)^j}$$

where $M_0, \ldots, M_n$, $B(i,j)$ are matrices and each term $1+c(i,j)t$ defines a pole of R.

Proposition - 3 : Let M be a matrix and f a continuous function, then the solution $V(t,t_0)$ of

(6) $\frac{\partial V}{\partial t}(t,t_0) = M\ f(t)\ V(t,t_0)$ is

(7) $V(t,t_0) = \exp M \int_{t_0}^{t} f(s)ds.$

Proof : Trivial.

We observe that

$$\int_0^t s^n ds = \frac{s^{n+1}}{n+1}$$

$$\int_0^t \frac{ds}{1+cs} = \frac{1}{c} \operatorname{Ln}(1+cs) \text{ and}$$

$$\int \frac{ds}{(1+cs)^j} = -\frac{1}{c(j-1)(1+cs)^{j-1}}$$

If M allows a diagonal descomposition

$$M = T \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} T^{-1}$$

where T is the matrix of eigen vectors then

$$(8)\ V(t,t_0) = T \begin{pmatrix} \exp d_1 \int_{t_0}^{t} f(s)ds & 0 \\ 0 & \exp d_2 \int_{t_0}^{t} f(s)ds \end{pmatrix} T^{-1}$$

From the above results the approximated solution x(t) is taken as

$$x(t) = \exp H(t-t_0)\ \exp S(t-t_0)$$
$$\exp M \int_{t_0}^{t} f(s)ds\ \ x(t_0)$$

3. Parameter identification

The model taken is

$$(1)\quad \frac{dx}{dt} = \left(h+S+\frac{B}{1+ct}\right) x(t)$$

$$(2)\quad y(t) = C\ x(t) + w(t).$$

So the first estimate of y is

$$(3)\quad \hat{y}(t) = A(0)\ \exp L(0)t + A(1)\exp L(1)t.$$

The negative values L(0) and L(1) are those computed by Prony's techniques from previous experiments.
We modified the solution

$$(4)\ \hat{y}(t) = A(0)\exp(L(0)+M(0))t + A(1)\exp(L(1))+M(1))t$$

with M(0) and M(1) scalars.

In the eigen basis for B, the solution of

$$\frac{dx}{dt} = \frac{B}{1+ct}\, x(t) \text{ is}$$
$$x(t) = A(0)(1+C(0)t)^{E(0)} + A(1)\ (1+C(1)t)^{E(1)}$$

thus with the new modification $\frac{B}{1+ct}$ we take

$$(5)\ \hat{y}(t) = \sum_{j,k} A(j,k)\exp(L(j)+M(j))t$$
$$(1+C(k)t)^{E(k)}.$$

We introduce an error between experimental datas and the model

$$(6)\quad \rho = \sum_k (y(k\Delta t) - \hat{y}(k\Delta t))^2$$

The coefficients A are linear ponderations of time dependent functions. The equations

$\frac{\partial \rho}{\partial A} = 0$ are linear.

For the coefficients M,C and E, the equations

$\frac{\partial \rho}{\partial M} = 0$ $\frac{\partial \rho}{\partial C} = 0$ $\frac{\partial \rho}{\partial E} = 0$ are not linear,

But S and $\frac{B}{1+ct}$ are small and we can use Taylor's formula to linearize these equations. Two programs written in Pascal with two hundred lines each calculate the new values A(0),A(1),M(0),M(1) and A(0,0),...,A(1,1),M(0),M(1),C(0),E(0),E(1).

A micro computer is sufficient to perform the calculation.

4. Conclusions :

We use the two Pascal programs for 250 simulated results Y satisfying relation :

A(0,0)=-0,52 A(0,1)=-0,11
A(1,0)=0,34 A(1,1)=-0,04
LA(0)=-0,037 LA(1)=-0,008
CO(0)=0,001 CO(1)=1,049
EX(0)=1,276 EX(1)=0,968

After 5 iterations on 150 values, we obtain
A(0)=+2,77 A(1)=-1,76
LA(0)=-0,045 LA(1)=-0,014

Then after 16 iterations on 250 values, we obtain the exact values.
Because of the extra non stationnary terms the first programm does not converge clearly on 250 values.
Then we applied the programs on experimental curves known to the precision of 1 percent. In that case the time T=0 of contact is not exactly known. More over asymptotic value $Y(\infty)=\lim Y(t)$ has to be estimated. In spite of these two new difficulties, we obtain
A(0)=2,56 A(1)=-2,09
LA(0)=-0,038 LA(1)=-0,008
and an error of 3,5 10^{-2} on 250 terms. For the improved model the error decreases to 4,37 10^{-3} and we estimated $Y(\infty)$= 3,447.
So non stationary terms are necesary to describe heat transfert from the censor to the skin. These new terms insure convergence of the algorithm for long experiments. They have to be included to increase precision.

Bibliography

[1] Grenier Y., Estimation Simultanée AR et MA d'un modèle non stationnaire, 10ème colloque Gretsi Nice 20-24 Mai 1985 p41-45.

[2] Grisel R., de Brucq D., Le Cordier R., Elimination de raies spectrales par la méthode de Prony étendue. Traitement du Signal Vol 2 n°1 (1985) p 75-78.

[3] Hubin M., Aribert-Desjardins A., Laforie P., Gouault J., Capteur de température superficielle à tranducteur couche mince. Conférence Capteurs (1984), p 368-373.

[4] Monteil J.P., modélisation numérique et réalisation d'une sonde de température compensée, Thèse Docteur Ingénieur Paris (1978).

[5] Pitarque T., Alengrin G., Houri A., Algorithmes de filtrage et de lissage appliqués à l'extraction de potentiels évoqués, 10ème colloque Gretsi Nice 20-24 mai 1985 p 1095-1099.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SIGNAL AVERAGING USING SHAPE CLASSIFICATION : APPLICATION TO HIGH RESOLUTION E.C.G.

Sérgio JESUS[+], Hervé RIX[+] and André VARENNE[++]

+ Laboratoire Signaux et Systemes, U.A. 814 CNRS, University of Nice, 41 Bd. Napoléon III- 06041 Nice-cedex France.
++ Laboratoire de Cardiologie (CRECEC) CHU Pasteur, 30 Ave. Voie Romaine, Nice France.

We present a method for signal classification, using the measure of shape differences by means of the Distribution Function Method. The algorithm is applied to ECG signal (P waves) An averaged signal is obtained for each class : the alignment process makes use of the P waves themselves as triggering signals.

1. INTRODUCTION

The classical averaging technique makes it possible to extract a reccurent signal from noise, under the folowing assumptions :
i) the signal to recover is identically reproduced from one realization to another;
ii) all the realizations are rigorously aligned;
iii) the additive noise is zero-mean, white and independant of the signal.

So the i^{th} realization of signal s(t), corrupted by the noise sequence $n_i(t)$, can be written in the form :

(1) $y_i(t) = s(t) + n_i(t)$

Averaging N realizations gives :

(2) $\bar{y}(t) = s(t) + \bar{n}(t)$

The signal-to-noise ratio (S/N) is increased of 10xp dB if $N = 10^p$.

In real cases of application like biomedical signals, these conditions are not strictly verified : so it is necessary to take into account deviations from ideal case. We have studied this problem in the analysis of High Resolution Electrocardiograms (HRECG) obtained on body surface [1]. The aim is to estimate either the fine structure of classical waves (P,Q,R,S,T) or micropotentials, invisible on a unique beat. At this level of detail, from one beat to another, we have to take into account variations of time interval lengths, e.g. PR or ST, and variations in shape of each wave. Therefore, we must have a triggering algorithm as precise as possible, abble to align directly the P waves (or T waves) and we must average signals having practically the same shape. A triggering algorithm has been provided by the authors [2] and checked on HRECG 's [3,4]. The aim of this paper is to propose a shape classification algorithm and then average signals (P waves) inside a same class, using our triggering method.

2. CLUSTERING AND AVERAGING SIGNALS

The ECG signal, coming from some lead on body surface,is sent to an apparatus abble to give HRECG 's (i.e. higly amplified ECG 's) : the amplification factor is chosen in function of the part of the signal to study. The digitized signal (sampling rate = 1000 Hz) is then processed by a computer. Currently the same signal goes through two channels with different amplification and filtering specifications : one for the alignment process (lower amplification) and the other for the averaging process [4]. In the present work we insert a clustering process before triggering and averaging inside a same class.

2.1. Classification

The chosen algorithm for classification needs no a priori information. It has been derived from the "K-means algorithm" [5] : the calculation of cluster centres is modified. In order to avoid the bias introduced by the jitter phenomenon the cluster centre is not a mean signal ; the centre is the nearest element (or one of the nearest elements) to all the others in the class. In addition time computation is rather reduced by this choice. The similarity criterium is given by the Distribution Function Method (DFM) [6], which has been successfully applied in analytical chemistry [7,8].

Recalling the DFM principle :

Let s(t) and v(t) be two signals assumed to be positive (e.g. the absolute value of two P waves) and S(t) and V(t) their normalized integrals. Function $t' = \phi(t)$ defined by

(3) $S(t) = V(t')$

characterizes the difference in shape between v and s by way of its deviation from linearity. From a set of discrete values (t_j, t'_j) (j=1 to n) we ca compute the coefficients a(s,v) and b(s,v) of the regression line of t'_j against t_j of the form :

(4) $t' = a\ t + b$

Let us call $\Delta(s,v)$ the quantity defined by :

(5) $$\Delta^2(s,v) = \frac{1}{N} \sum_{j=1}^{N} \left(\phi(t_j) - a(s,v)t_j - b(s,v) \right)^2$$

To measure shape similarity between s and v we chose the symetrical quantity :

(6) $\Delta^*(s,v) = 0.5\ (\Delta(s,v) + \Delta(v,s))$

2.2. Alignment and averaging

After classification, signals belonging to a same class are algned with the signal designed as class centre. The algorithm performs the coincidence of the mean times of the signals to be averaged [2].

Recalling the method :

With the same notations as before, for s(t) and v(t) having their support in [a , b] interval, one estimates :

(7) $$Q(\tau_i) = \int_a^b (S(t) - V(t - \tau_i))dt$$

for a series of given shifts τ_i .Without noise, and for s and v having the same shape, i.e. :

(8) $v(t) = k\ s(t-d)$ (k, d constants)

function Q( ) has the form :

(9) $Q(\tau) = d + \tau$

So, $Q(\tau)$ is a straight line with zero-crossing when $\tau = -d$. In presence of noise, d is estimated by the zero-crossing of the regression line of $Q(\tau_i)$ versus τ_i. Then averaging is done using the estimated shifts.

3. RESULTS

The real data come from the cardiology department (Hopital Pasteur, Nice) and have been recorded with a PANCARDIOGRAPH BIOSIGNAL which is an acquisition device coupled to a microcomputer. In our case the recorded signal has been digitized (1000 Hz) and processed on a minicomputer (LSI 4/90 COMPUTER AUTOMATION) at the laboratory.

3.1. Checking the method

In order to verify the running of our algorithm we used a series of T waves for which clustering was obvious by visual inspection. This series of T waves (like those of fig. 1) comes from an exercise ECG : the isoelectric line variations create a variable flattening from one wave to another. The number of classes has been successively fixed to 2, 3 and 4 : in each case, we verified that shape classification coincided with the degree of flattening, and that the obtained classes were independant of the initial choice for the class centres.

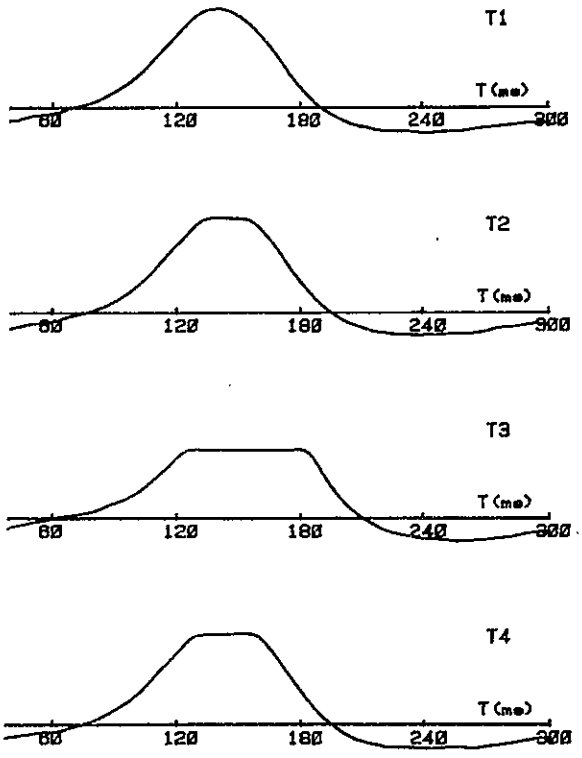


Fig. 1.

3.2. Application to P waves

The algorithm has been checked on P waves coming from a healthy man and where visual classification was practically impossible. Figures 2 and 3 show averaged waves respectively for a resting ECG and for a light exercise ECG. The number of classes was 2 and the total number of

beats was 80 in each case. The convergence of the algorithm towards stable classes was practically independant of the initial choice of class centres ; it lead to 34 and 56 elements for resting ECG (fig. 2) and to 24 and 56 elements for exercise ECG (fig. 3). In order to have a significant comparison between the two classes, we show averaged corresponding to the same beat number. One can see on fig. 2(a) and fig. 2(b) a difference between the mean P waveforms. In fig. 3(a) and fig. 3(b) it is chiefly obvious in regions before P (level height) and after P. These examples show that small variations in shape can be detected by our algorithm.

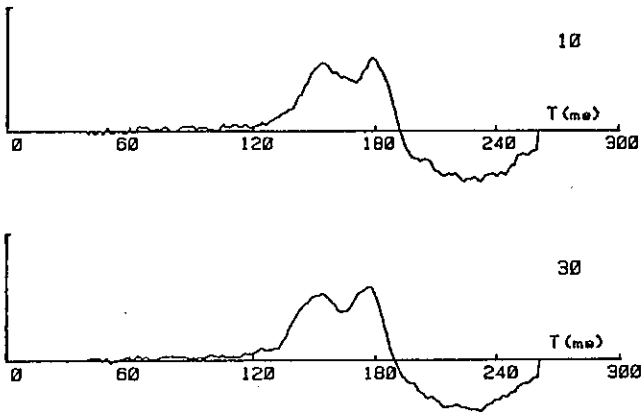


Fig. 2.A.

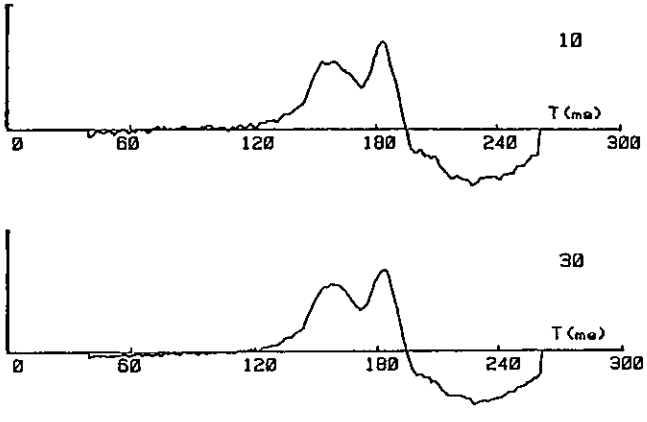


Fig. 2.B.

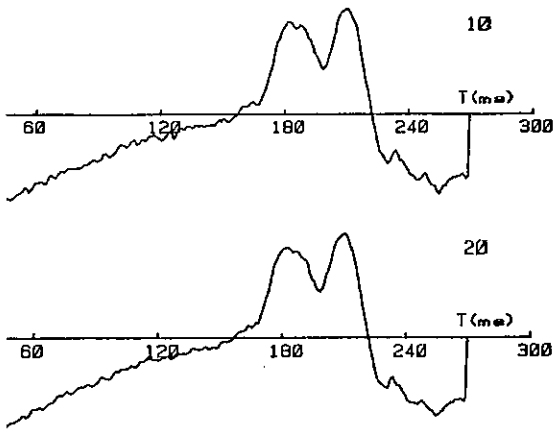


Fig. 3.A.

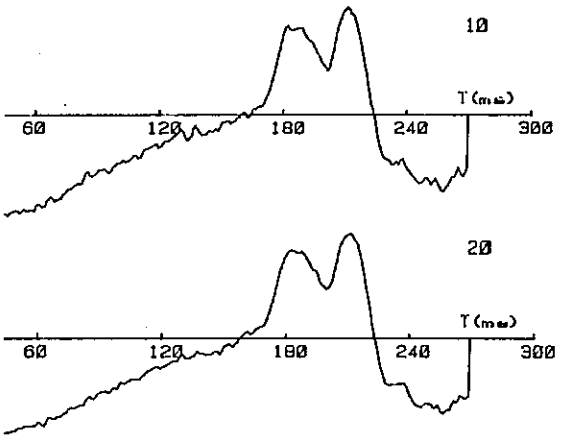


Fig. 3.B.

4. CONCLUSIONS

We have presented an analysis procedure suited to study recurrent physiological signals e.g. ECG 's. The procedure combines a new triggering algorithm, reducing PR and ST jitter, to a clustering algorithm utilizing a shape similarity measure. This procedure has to be connected to that using PR interval length as classification criterion [9]. From the few presented results, obtained with a healthy man where differences in ECG shape are small, one can anticipate more evident results in pathologic cases.

REFERENCES

[1] Jesus, S., Doctoral Thesis, Univ. of Nice, in print.
[2] Rix, H. and Jesus, S., C.R. Acad. Sc Paris t.299 II 8 (1984) 399.
[3] Jesus, S. and Rix, H., Proc. of the XIV ICMBE and VII ICMP, Med. and Biol. Eng. and Computing vol. 23 suppl. part 2 (1985) 1462.
[4] Jesus, S. and Rix, H., ECG Analysis: improvement of signal averaging and comparison with a beat to beat approach using adaptive identification in : Luque, A., Figueiras, A.R. and Delgado, J.M.R. (eds), MELECON'85, vol. 1 : Bioengeneering (Elsevier Sc. Publ. B.V. North-Holland, IEEE, 1985) pp. 135-138.
[5] Tou, J.T. and Gonzalez, R.C., Pattern recognition principles (Addison-Wesley, Massachusetts,USA, 1974)
[6] Rix, H. and Malengé, J.P., IEEE Trans. Syst. Man and Cybern., 10 2 (1980) 90.
[7] Rix, H. and Malengé, J.P., J. of High Resolution Chromatography and Chrom. Commmunications, 3 4 (1980) 172.
[8] Rix, H., J. of Chromatography, 204 4 (1981) 163.
[9] Ros, H.H., Koeleman,A.S.M., Hand, R.C. and Flowers,N.C., Proc. of The XIV ICMBE and VII ICMP, Med. and Biol. Eng. and Computing, vol. 23 suppl. part 2 (1985) 1459.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A RAPID ANGIOGRAPHIC TECHNIQUE TO MEASURE RELATIVE CORONARY BLOOD FLOW

J. van Ommeren, M.Sc., F. Zijlstra, M.D., P.W. Serruys, M.D., J.H.C. Reiber, Ph.D.,

Laboratory for Clinical and Experimental Image Processing, Thoraxcenter, Erasmus University and University Hospital Dijkzigt, Rotterdam, The Netherlands.

Coronary cineangiograms acquired during a cardiac catheterization procedure provide information about the location and anatomic severity of obstructions in the coronary arterial tree. In this paper a rapid angiographic technique is described that allows the assessment of additional clinically relevant information from the cineangiograms, being the regional coronary blood flow measurement.
Regional coronary blood flow is determined from the temporal changes in the brightness levels in a user-defined myocardial region-of-interest in a cineangiogram. Functional parameters for the calculation of blood flow are presented in parametric images. A time-parameter image is used to define contrast medium appearance. A maximum density image provides information for the computation of regional vascular volume. From the mean density and the appearance time, the regional coronary blood flow can be calculated. Acquiring and analyzing these image data at the control state and at maximal flow, allows the measurement of coronary flow reserve being a measure for the functional significance of a coronary obstruction.

1. INTRODUCTION

Coronary angiography is the only technique that allows the high resolution visualization of the coronary anatomy in men. In our laboratory quantitative techniques have been developed for the accurate and reproducible analysis of the dimensions of coronary arterial segments from 35 mm cineframes on the Cardiovascular Angiography Analysis System (CAAS) (1-3). In addition to the accurate anatomic description of the severity of coronary obstructions, it is of great interest to obtain information about the functional significance of such obstruction, i.e. the question is posed: to what extent does the obstruction limit the flow through that artery?
During the last five years techniques have been developed at several institutes for the determination of time and flow parameters in digital angiography (4-9). By means of these parameters the functional significance of anatomically defined coronary obstructions can be determined.
In this paper a digital cineangiographic technique is described, which allows the quantitation of relative regional coronary blood flow on the basis of myocardial contrast perfusion measurements. This technique was introduced by R.A. Vogel, et al. (5,6). Although this technique is usually applied to video images of the coronary arterial system acquired directly from the image intensifier of the X-ray system, we have applied the technique to 35 mm cineframes.

For the computation of relative regional coronary blood flow, ECG-gated end-diastolic (ED) images are acquired from successive cardiac cycles. Following digitization of the image series and background subtraction, the appearance time of the contrast medium bolus for each picture element (pixel), as well as the maximum density per pixel is determined over the entire series of images. The time values are color coded and stored in an appearance time picture. The density values are stored in a contrast density image. For a user-defined region of interest the relative regional coronary blood flow is then calculated as the quotient of the mean maximal density and the mean appearance time for that particular myocardial region. The functional significance of a coronary obstruction can be assessed from two relative regional coronary blood flow measurements, one under normal flow conditions (baseline), the other under maximal flow (hyperemic) conditions. The quotient of flow during hyperemia and at baseline is the socalled Coronary Flow Reserve (CFR)-ratio. This is a measure for the capacity of the coronary arteries.

2. METHODS

2.1. Angiographic image acquisition

Coronary cineangiograms are obtained with a 35 mm film camera mounted on the X-ray image intensifier. The filmspeed for coronary cineangiograms is taken at 25 frames/s. Images are

acquired after ECG-triggered injection of a nonionic contrast medium using a Medrad Mark IV infusion pump. During image acquisition atrial cardiac pacing is performed at a rate just above the spontaneous heart rate. For the left coronary artery 7 ml was injected at a flow rate of 4 ml/sec in a left anterior oblique projection. For the right coronary artery 5 ml is injected at a flow rate of 3 ml/sec in a left or right anterior oblique projection. The angiogram is repeated 30 sec after a bolus injection of 10 mg papaverine into the coronary artery.

2.2. Cineframe digitization

For the quantitation of the relative coronary blood flow, five to eight ED-cineframes are selected for digitization from successive cardiac cycles. These cineframes are digitized at a resolution of 512 x 512 pixels and 256 gray levels. The digitized images are corrected for the dark current of the video camera. Logarithmic mask-mode background subtraction is applied to the image subset to

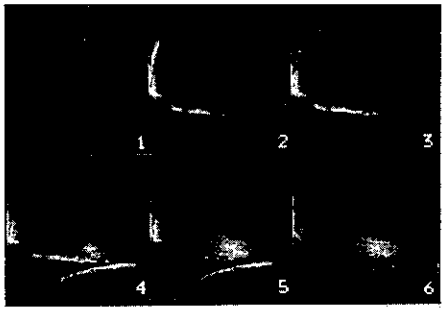


Fig. 1. End-diastolic cineframes, digitized in 512 x 512 eight-bit matrices from 6 consecutive heart beats following contrast injection. Stationary background structures were eliminated by means of logarithmic mask-mode background subtraction using the ED-cineframe acquired prior to contrast administration as a mask.

eliminate non-contrast medium densities. The last ED-cineframe prior to the contrast administration is chosen as the mask. The intensity level in the background subtracted cineangiograms is proportional to the irradiated amount of contrast material (10,11). An example of a series of 6 consecutive background subtracted ED-cineframes is shown in figure 1.

2.3. Time parameter extraction

From the sequence of background subtracted cineframes a contrast medium appearance time picture is generated, using a fixed threshold (12%) in pixel brightness level. The individual pixels in this image are color coded, based on the sequence number of the heart cycle in which the pixel intensity for the first time exceeds the threshold, starting from the beginning of the ECG-triggered contrast injection. Red was assigned to the pixels whose intensity surpassed the threshold during the first post-injection cycle, yellow for the second cycle, white for the third, green for the fourth, and so on.
For the calculation of relative regional blood flow, within a user-defined myocardial region of interest (ROI) two parameters are required, i.e. relative regional vascular volume and mean contrast appearance time.

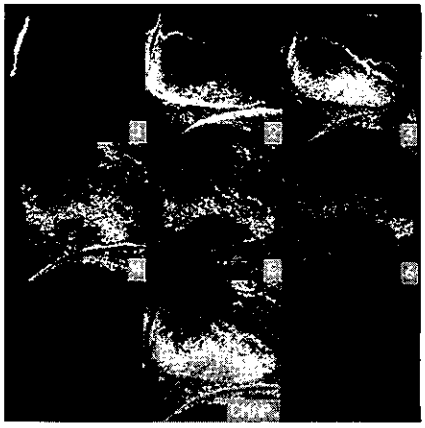


Fig. 2. Contrast medium appearance in 6 consecutive cardiac cycles from the example of Fig. 1. The first two cycles show the arterial phase, while the other cycles show the myocardial perfusion phase. The 6 individual cycles are finally combined into the last image, being the color-coded contrast medium appearance picture (only shown in black/white).

Contrast medium appearance in the first post-injection cycle counts as 0.5 cycle, in the second as 1.5 cycle, and so on. The contrast medium appearance in consecutive heart cycles is demonstrated in Figure 2.

2.4. Regional vascular volume determination

The relative regional vascular volume, the second parameter required in the calculation of relative regional coronary blood flow, is calculated from a maximum intensity image, which is to be generated from the sequence of background subtracted cineangiograms. Each picture element in this image represents the maximal pixel intensity level found within the series of background subtracted cineframes. As a result, the maximum intensity image contains information on the maximum contrast medium concentration within the displayed vessels as it occurred during the acquisition period. The maximum intensity image generated from the image series of Figure 1 is demonstrated in Figure 3.
This maximum intensity image shows the distri-

Fig. 3. Maximum intensity image, generated from the image series of Figure 1. Each individual pixel intensity in this image represents the maximum intensity for that particular pixel as it occurred over the entire image series.

bution of contrast agent over the cardiovascular system. As stated before (10,11) the intensity value in background subtracted cineangiograms is proportional to the transradiated amount of contrast material within the vessels. Under the assumption of homogeneous mixing of the contrast medium with the blood, the regional vascular volume V for a user-defined ROI is proportional to the mean radiographic density within that ROI:

$$V = k\int_{ROI} D(p)\, dp = k\bar{D} ,$$

where k is a radiographic constant and D (p) the radiographic density per pixel and $\bar{D}$ the mean radiographic density within the ROI.

2.5. Relative regional flow

Next, the color coded time parameter image, shown as a grey-tone image in Figure 2, is combined with the maximum intensity image, resulting in a dual parameter image, which contains both time and density information. In this dual parameter image appearance time is color coded and contrast medium accumulation is represented by the individual pixel intensity values.
Regional flow values can then be determined quantitatively using the following videodensitometric principle:

$$Q = V/\bar{T} ,$$

where Q is the regional flow, V the regional vascular volume, and $\bar{T}$ the mean transit time.

We are interested in the ratio of the coronary flow under normal (baseline) conditions and under maximal flow (hyperemic) conditions, for the determination of the socalled coronary flow reserve factor. As flow ratios are determined, only relative and not absolute regional flow values for baseline and hyperemic conditions are required. Therefore, coronary flow reserve (CFR) is defined as the quotient of relative hyperemic and baseline flows Q_h and Q_b, respectively:

$$CFR = \frac{Q_h}{Q_b} = \frac{V_h \cdot \overline{T_b}}{\overline{T_h} \cdot V_b} = \frac{\overline{D_h} \cdot \overline{T_b}}{\overline{T_h} \cdot \overline{D_b}}$$

Mean contrast density and appearance time are computed within a user-defined ROI. The ROI is chosen in such a way that the large epicardial arteries, the aortic root and the coronary sinus are excluded from the analysis.

3. CLINICAL RESULTS

The technique was applied to seventeen coronary arteries of patients with single vessel coronary artery disease (CAD) and 6 coronary arteries of patients without CAD. The 17 coronary artery lesions were all single discrete stenoses in the proximal parts of the vessels before any sidebranch occurred. The anatomic severity of the coronary obstructions was determined with the CAAS; the minimal cross-sectional area (MLCA, mm^2) and the area stenosis (AS,%) were assessed for each stenosis from an average of 2.3 angiographic projections. The relation between CFR and MLCA was best described by the quadratic equation:

$$CFR = 0.28 + 0.91\ MLCA - 0.039\ (MLCA)^2,\ r=0.92,$$

and the relation between CFR and AS by:

$$CFR = 5.0 - 3.3(AS \times 10^{-2}) - 1.3(AS \times 10^{-2})^2,\ r=0.92.$$

Table 1: Relation between quantitative assessed coronary artery dimensions and CFR.

	CAD		normals
	severe N = 11	moderate N = 6	N = 6
MLCA	$<2\ mm^2$	$2\text{-}4.5\ mm^2$	$>4.5\ mm^2$
AS	>70%	50-70%	<50%
CFR (mean ± SD)	1.0 ± 0.3	2.6 ± 0.7 p=0.001	5.0 ± 1.0 $p<10^{-4}$

The investigated vessels were divided into three categories on the basis of both AS and MLCA (Table I). The 6 normal coronary arteries were compared with the 6 coronary arteries with an AS between 50% and 70% and a MLCA between 2 and 4.5 mm^2 (moderate CAD) and with the 11 coronary arteries with an AS in excess of 70% and a MLCA less than 2 mm^2 (severe CAD). The vessels with severe CAD had a mean CFR of 1.0 (s.d. ± 0.3) and differed highly

significantly (p= 0.001) from the CFR of the vessels with moderate CAD, who had a mean CFR of 2.6 (s.d. ± 0.7). The difference between the normal vessels (CFR = 5.0 ± 1.0) and the vessels with moderate CAD was also highly significant ($p < 10^{-4}$).

4. VALUE AND LIMITATIONS

With the technique described in this paper it is possible to study the physiologic significance of coronary lesions (Table 1). The choice of a myocardial region of interest however is restricted. In the two-dimensional image there does not exist a one-to-one relation between a coronary segment and the corresponding myocardial region that is perfused by that segment. Overprojections with other regions may occur. In the selection of an ROI overlap with epicardial vessels must be avoided. At the present time we are studying the possibilities of three-dimensional reconstruction of the myocard from two orhtogonal projections to overcome such problems.

5. CONCLUSION

In conclusion, a technique for the assessment of relative regional coronary blood flow and coronary flow reserve from 35 mm cineangiograms has been implemented. By this approach the relation between the anatomic severity of a coronary obstruction and its functional significance can be studied. From the results obtained sofar, it may be concluded that the reduction in CFR for a stenosis can be predicted with reasonable accuracy by quantitative assessment of the coronary artery dimensions. Further validation studies will be carried out to define exactly the value and limitations of the technique for the measurement of coronary flow reserve.

ACKNOWLEDGEMENTS

The authors wish to thank Mrs. S.M. Spierdijk and Mrs. M.J. Kanters-Stam for their secretarial assistance in the preparation of this manuscript.

REFERENCES

1. Reiber JHC, Kooijman CJ, Slager CJ, Gerbrands JJ, Schuurbiers JCH, Boer A den, Wijns W, Serruys PW, Hugenholtz PG.
Coronary artery dimensions from cineangiograms. Methodology and validation of a computer assisted analysis procedure.
IEEE Trans Med Imaging MI-3, 1984: 131-141.
2. Reiber JHC, Serruys PW, Kooijman CJ, Wijns W, Slager CJ, Gerbrands JJ, Schuurbiers JCH, Boer A den, Hugenholtz PG.
Assessment of short-, medium- and long-term variations in arterial dimensions from computer-assisted quantitation of coronary cineangiograms.
Circulation 71, 1985: 280-288.
3. Reiber JHC, Kooijman CJ, Slager CJ, Gerbrands JJ, Schuurbiers JCH, Boer A den, Wijns W, Serruys PW.
Computer assisted analysis of the severity of obstructions from coronary cineangiograms: a methodological review.
Automedica 5, 1984: 219-238.
4. Hahne HJ.
Time and flow parameter extraction in digital angiography: Principles and Methods.
Herz 10, 1985: 220-227.
5. Vogel R, LeFree M, Bates E, O'Neill W, Foster R, Kirlin P, Smith D, Pitt B.
Application of digital techniques to selective coronary arteriography: use of myocardial contrast appearance time to measure coronary flow reserve.
Am Heart J 107, 1984: 153-164.
6. Hodgson JMcB, LeGrand V, Bates ER, Mancini GBJ, Aueron FM, O'Neill WW, Simon SB, Beaumon GJ, LeFree MT, Vogel RA.
Validation in dogs of a rapid digital angiographic technique to measure relative coronary blood flow during routine cardiac catheterization.
Am J Cardiol 55, 1985: 188-193.
7. Smith HC, Robb RA, Ritman EL.
Roentgen videodensitometric assessment of myocardial blood flow: clinical applications.
In: Roëntgen-Video-Techniques for Dynamic Studies of Structure: Function of the Heart and Circulation. PH Heintzen, JH Bürsch (Eds). Georg Thieme Publishers, Stuttgart, 1978: 39-48.
8. Bürsch JH.
Densitometric studies in digital subtraction angiography: assessment of pulmonary and myocardial perfusion.
Herz 10, 1985: 208-214.
9. Ratib O, Chappuis F, Rutishauser W.
Digital Angiographic technique for the quantitative assessment of myocardial perfusion.
Annales de Radiologie 28, 1985: 193-197.
10. Reiber JHC, Slager CJ, Schuurbiers JCH, Boer A den, Gerbrands JJ, Troost GJ, Scholts B, Kooijman CJ, Serruys PW.
Transfer functions of the X-ray-cine-video chain applied to digital processing of coronary cineangiograms.
In: Digital Imaging in Cardiovascular Radiology. PH Heintzen, R Brennecke (Eds). Georg Thieme Verlag, Stuttgart, 1983: 89-104.
11. Kooijman CJ, Kalberg R, Slager CJ, Tijdens FO, Plas J van der, Reiber JHC.
Densitometric analysis of coronary arteries.
This volume.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AN ADAPTIVE METHOD FOR ECoG SIGNAL FILTERING

Antoni GRZANKA

Institute of Fundamental Electronics,
Technical University of Warsaw,
00-665 Warsaw, Nowowiejska 15/19, Poland*

The methods of selecting the linear filtration to be suitable to measured signals have been considered. The practical algorithm of adaptive filtration has been developed and tested on real ECoG signals.

1. INTRODUCTION

The electrocochleographic (ECoG) evoked response is an electrical answer generated by the cochlea nerve to specially formed, short acoustic stimulus [2]. Like other electrophisiological evoked responses this signal begins at the moment of stimulation and disappears quickly. So, it can be considered as a signal of finite time duration.

Because of presence of spontaneouse nerve action not related to the stimulus, the results of succesive measurements are not repeatable. Practically, the signal should be treated as a realisation of a random process and statistical approach to extraction of information must be applied. The simplest (but only hypothetical) model of the signal assumes that the observed signal $y(t)$ is the sum of the real evoked response $r(t)$ and a random component $n(t)$ with zero mean value.

$$y(t) = r(t-t_0) + n(t); \text{ for } t>t_0 \quad (1.1)$$

where:
t_0 denotes the moment of stimulation,
$n(t)$ represents both spontaneouse nerve action and measuring noise.

The $y(t)$ processing aimes at calculation of aproximate waveform of $r(t)$ or, in other words, decrease of the noise level in the observed signal $y(t)$. Adoption of a specific estimation method depends on our knowledge of the signal. A popular technique consists in repeated $y_j(t)$ measurements under the same stimulation. The response $r(t)$ is calculated in terms of averaging of $y_j(t)$ for each time point separately. Elementary statistics indicates that if both $r(t)$ and covariance function of $n(t)$ are constant and noise components $n_i(t)$, $n_j(t)$ of two different responses are uncorrelated, then averaging is the optimal method in any sense.

These conditions, however, are very strong with reference to the facts. Hearing, like other senses, adapts oneself to stimulus volume, therefore the assumption of constant response is not exactly valid. Investigation of adaptation phenomenon requires a method giving different results for each response.

If some quantitative data for $r(t)$ variation were available, a Bayesian method [1] of filtration would be adopted. In fact, a response depends on individual characteristic of a patient and both a stimulus volume and its repetition frequency (see Fig. 1), so there is no a priori information of kind mentioned above.

Therefore, we have reconsidered possibilities of formal linear operations to be suitable for filtration. According to this concept, the operator will be able to be individually matched with gathered real data by selection (optimization). The mean square error has been proposed as a gauge of fitting.

It will be demonstrated in the begining of Section 2, that selecting process from the full class of linear operators will provide only amplitude of responses being differentiate. We have removed this "over-estimation" by limiting number of degrees of freedom in fitting

* This research was supported by PW 06.9.

procedure. The selection has been reduced to the class of wighted windows in domain of settled orthonormal transform. This concept has been considered in detail in the second part of Section 2.

Basing on theoretical results, the algorithm has been worked out using the method of limited optimization (Section 3). The reader interested in results of the ECoG signal processing only may wish to proceed to Section 4.

2. THEORETICAL BACKGROUND

Given a set of responses as a sequence of vectors, Y_j; $j=1..L$, of sampled data assume that each vector is composed of real response, R_j, and additive random part, N_j. Each term is a realisation of adequate vector random variable denoted by Y, R and N respectively. The noise is of zero mean value, so mean of R (denoted by M) is equal to mean of Y:

$$M = E(R) = E(Y) \tag{2.1}$$

Consider a linear operator represented by A matrix of KxK elements, where K is a length of signals. Transforming Y by A gives a resultant vector variable Y':

$$Y' = AY \tag{2.2}$$

We desire to obtain the A matrix that will give the best matching of Y' with a real response. By choise the mean square error as a measure of fitting we obtain:

$$e = E(\|R-Y'\|2) \tag{2.3}$$

We assume both the variance of R and covariance of R and N to be small in comparision with square of mean. Then approximately:

$$e = E(\|M-Y'\|2) \tag{2.4}$$

This relation will be the basis for our analysis. Below, we shall present effects of the error minimization for two classes of A matrices.

2.1. Full matrix optimization

Minimization of mean square error brings the following equation for each i-th row of A matrix:

$$C\,A_i = E(m_i\ M) \tag{2.5}$$

where C is the matrix of second join moments of Y with:

$$c_{ik} = E(y_i\ y_k) \tag{2.6}$$

(Lower case indexed leters denote elements of adequate vectors and matrices).

Finally, the A has following structure:

$$A = \begin{bmatrix} m_1\ M^T C^{-1} \\ m_2\ M^T C^{-1} \\ \dots\dots\dots \\ m_K\ M^T C^{-1} \end{bmatrix} \tag{2.7}$$

Replacing random variables in Eq. (2.2) by their realisations we obtain:

$$\begin{gathered} Y_j' = a_j\ M \\ a_j = M^T\ C^{-1}\ Y_j; \quad j=1 \dots L. \end{gathered} \tag{2.8}$$

Notice that the "filtered" responses, Y_j, differe from each other only by scalar a_j. So this method, howevwr very interesting, could be applicable to signals with only response amplitude variability.

2.2. Limited optimization

In order to receive individual shapes of Y_j' we will make some restriction for A in form of :

$$A = T^{-1}\ D\ T \tag{2.9}$$

where:
T - matrix of a orthonormal transform,
D - a diagonal matrix being searched for.

On the ground of the Parselval's theorem square norm is invariable under any orthonormal transform, so:

$$\|M-T^{-1}DTY\|2 = \|TM-DTY\|2 = \|M_T-DY_T\|2 \tag{2.10}$$

The T-indexed leters denote results of transformation by T.

Any linear operation adheres to the terms that a random vector is of zaro mean. Applying this for N vector gives the following solution of optimization:

$$d_i = \frac{m_{Ti}2}{c_{Ti}} \tag{2.11}$$

where $c_{Ti} = E(y_{Ti}2)$.
Note that $0 \langle d_i \langle 1$.

The filter coefficients can be easily calculated from a number of measured signals, Y_j. Adequate estimators are given by:

$$\hat{m}_{Ti} = \frac{1}{L}\sum_{j=1}^{L} y_{ji}; \quad \hat{c}_{Ti} = \frac{1}{L}\sum_{j=1}^{L} y_{ji}2 \tag{2.12}$$

Some sophisticated applications including estimation of necessary number of data or tests of significance,

require more detail analysis. The asymptotic value of di variance can be easily obtained if we take in consideration that statistics

$$\hat{m}_{Ti} \text{ and } \hat{s}_{Ti} = \hat{c}_{Ti} - \hat{m}_{Ti}^2$$

are independent and their variances decrease with number of data; and then:

$$V(\hat{d}_i) = \left(\frac{\partial d_i}{\partial m_{Ti}}\right)^2 V(\hat{m}_{Ti}) + \left(\frac{\partial d_i}{\partial s_{Ti}}\right)^2 V(\hat{s}_{Ti}) \qquad (2.13)$$

For Gaussian distributed noise this gives following expression for the variance:

$$V(\hat{d}_i) = \frac{2}{L} d_i (2-d_i)(1-d_i)^2 \leq 1/2L \qquad (2.14)$$

with maximum for $d_i = 1-\sqrt{2}/2 \approx 0.29$

This result could be obtained by analysis of the likelihood function.

The distribution of d_i for $m_i = 0$ is connected with t-Student distribution with L-1 degrees of freedom:

$$\hat{d}_i = \frac{t^2_{L-1}}{L-1+t^2_{L-1}} \qquad (2.15)$$

The test for stating d_i to be significant (different from zero) is equivalent in numbers to well known t-Student test for m_i average with unknown variance.

3. IMPLEMENTATION

The algorithm based on described method consists of two passes. During the first pass, optimal filter is being chosen. The final filtration is performed during the second one. Both passes need any response to be available, so the signal must be digitized and stored in the memory. Two versions of the algorithm will be presented; the less-memory consuming and less-time consuming algorithms.

3.1. Less-memory consuming version

Pass 1. Each response, one after another, is tranformed by T completing two vectors of sums. The first vector contains ordinary sums of transformed responses, the second one contains sum of squares. These data are sufficient to compute diagonal of D as the filter. In general, the filter dumps of a signal, so it is convenient to multiply all filter coefficients by a normalizing constant.

Pass 2. A response to be filtered is transformed, then the result is multiplied by diagonal (element by element). Finally, the inverse transform is applied giving filtered respnse.

3.2. Less-time consuming version

If there is enough memory in the computer, transformed responses can be saved during the first pass. So, computation of simple transform will not be necessery during the second pass.

4. EXPERIMENT

4.1. Material

The recodings used to demonstrate the adaptive filtration were selected in the ENT Clinic of the Institute of Surgery of the Medical Academy of Warsaw. The patients were lying on bed in a sound-treated room and the evoked potentials were obtained by transtympanal technique. The stimuli consisted of clicks were produced by applying 0.1 ms rectangular pulses of alternate polarity to TDH-39 earphone. Various interstimuli intervals (ISI) and intensities of stimulus were applied. The ISI was set at 50 or 100 ms; the stimulus level was chosen as 110, 100, 90 or 80dB peak SPL (sound pressure level).

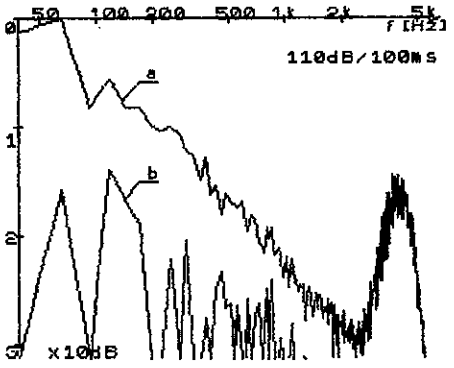
Data epochs of 19.2ms were sampled in 256 points using 13.3kHz sampling frequency and 8-bits word. 47 sweeps were acquired and stored in computer memory. Off-line, the signal to noise ratio was calculated as being at -15dB level.

4.2. Results

The presented results were obtained using the Fourier transform as T. Fig. 1 shows the averaged squre of transformed signals (a) and absolute value of transformed average (b) for two stimulus levels: 110dB (upper) and 80dB (lower). Note that the Figure 1 does not show the spectrum of signal but the results of transformation, so two components, cosinusoidal and sinusoidal, have been marked for each frequency.

This plot demonstrates the noise level falls down with slope approx. 20dB/dec. Rising values for higher frequencies are related with so-called stimulus artifact, which disturbes the useful information. The alternate polarity stimulation couses compensation of artifact in averaged response.

The ratio of functions presented on Figure 1 constitutes the filter



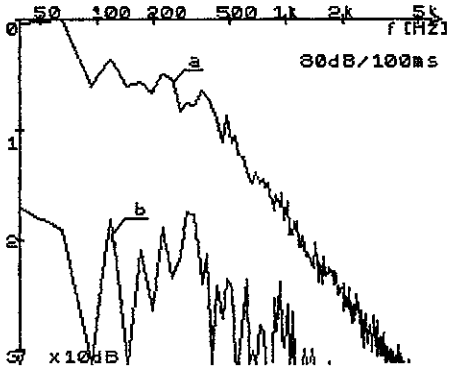


Figure 1

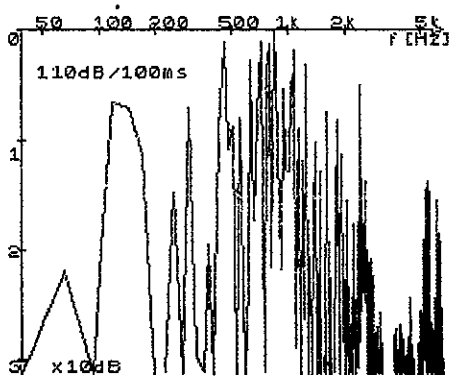


Figure 2

coefficients. The relative attenuation of the filter has been shown on Figure 2 in the same convention as above data.

The examplary result of final processing have been demonstrated on Figure 3. Two favourable effects of the filtration may be observed: the stimulus artifact cancelation and the action potential (AP) extraction. It is visible that the waveform of similar shape like AP appears on a random place in the signal. This phenomenon has been confirmed using correlation method.

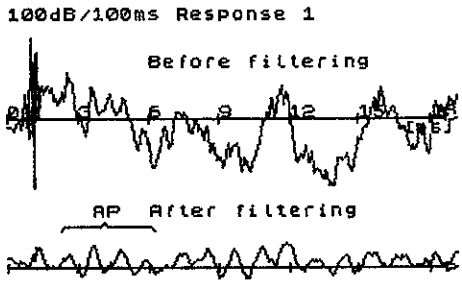


Figure 3

The Hadamard (and equivalent Walsh) transform has been tested as T without satisfactionary results.

5. SUMMARY

In order to individualize the waveforms of single evoked responses with decreased noise level, we have developed the adaptive method of selecting a linear operator from the class of filters in the settled orthonormal transform domain. The filter is determined by only K coefficients, where K denotes the lenght of single response. For each group of responses, the filter can be easily found by transforming input data and creating the simple sum and sum of squares for all signals. The final filtration consists of three stages: simple transformation of required response, multiplying by filter coefficients and inverse transformation. This way is similar to usually applied filtration algorithm, but notice that in our method not only amplitude of a signal is being modified but its phase as well. The experiments that have been carried out to test ECoG signal prove that this method is of great usefulness in the first stage of the signal examination, when there is no a priori information and various concepts of the signal nature must be considered.

ACKNOWLEDGEMENTS

We should like to thank Mr K. Kochanek for the signals measurement.

REFERENCES

[1] Anderson, B.D.O. and Moor, J.B., Optimal filtering (New Jersey 1979).

[2] Teas, D.C., Eldredge, D.H. and Davis H., Cochlear responses to acoustic transients: An interpretation of whole-nerve compound action potentials, in J. Acoust. Soc. Am. 34 (1962), pp. 1438-1459.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

3-D DIGITAL FILTERING OF BIOMEDICAL IMAGES

V.Cappellini
Dipartimento di Ingegneria Elettronica, University of Florence
and IROE-C.N.R., Florence, Italy

R.Carlà
Istituto di Ricerca sulle Onde Elettromagnetiche (IROE),
National Research Council (CNR), Florence, Italy

M.Melani
Dipartimento di Ingegneria Elettronica, University of Florence,
Florence, Italy

In the biomedical area there is often the clinical necessity to examine sections of the human body along directions in which an image acquisition could not be performed: to this end a set of data of the three-dimensional (3-D) representation of the organ analyzed is required. Moreover it is very important to extend to 3-D space the techniques already developed and currently used in 2-D space for analysis and information extraction purposes. In this paper the problem of the interpolation and representation of 3-D images from a small set of 2-D sections is considered first; some preliminary results of 3-D filtering of the 3-D reconstructed images are then presented by using a 3-D FIR zero-phase filter.

1. INTRODUCTION

Digital image processing plays a relevant role in the biomedical area. Many digital image processing techniques have been developed with the purpose of obtaining higher quality images more useful for the physicians interpretation and diagnosis. In the biomedical area there is also an increasing interest and necessity of examining multiple sections of human body, in particular along directions in which there is no possibility of performing a direct image acquisition. Moreover a representation of a 3-D structure gives a better idea of shape, dimension and position as well as more accurate localization of possible anomalies, and hence a more precise diagnosis. Therefore it is very important to extend to the 3-D space the techniques already developed and currently used in the 2-D space for analysis and information extraction purposes [1], [2]. In this paper the problem of the reconstruction and representation of 3-D images from a small set of 2-D sections is considered first; some preliminary results of 3-D digital filtering of the 3-D reconstructed images are then presented, which were obtained by using a 3-D FIR (Finite Impulsive Response) zero phase filter, already extensively tested in the 2-D domain on Computerized Axial Tomography (CAT) images.

2. IMAGE INTERPOLATION AND REPRESENTATION

A 3-D image is a solid to each point of which a given gray level is associated with each point of this solid. It can be thought of as a sequence of parallel sections of the anatomic portion examined, each normal to a particular direction (e.g. the z-axis of a coordinate system), with a fixed distance between two adjacent sections.

In order to obtain a set of data with the same definition along all the three axes, the sampling period Dz between two adjacent sections must be equal to the spatial resolution in each section. In this way each "voxel" or volume element contains the information of a cubic cell, the whose side length represents the resolution of the 3-D image. The 3-D interpolation is in general required because a limited number of sections or images is available along the axis orthogonal to the image planes.

It is possible to obtain N sections from the M

original sections through interpolation algorithms (linear, cubic splines, etc.), and consequently to achieve the same definition along all the three axes, and voxels effectively of cubic shape. Linear digital interpolation seems to be adequate for many purposes. A main problem that arises in processing 3-D images, is the image representation on a 2-D plane of the reconstructed 3-D information. In general, the method used must meet many requirements such as short processing time, flexibility, ease of use, and efficiency of presentation of the shape, size and interrelationship of the various structures. Many useful axonometric representation and particularly perspective projection methods can be used in order to obtain a global outlook of the structure examined, even if these display methods involve very complex data processing procedures which are needed for hidden-object removal, shading, transparency, projection direction, etc., and the results often require a more complex analysis for a clear identification of some details of the displayed structure.

A very simple but efficient display method of a 3-D image is a sequence of successive slides of its sections, which are taken out parallel to any axis choosen. This method is particularly efficient when there is the necessity to evaluate accurately the shape and the size of the structure analyzed and the details of possible pathologic conditions. Moreover, the selection of a set of sections normal to a particular direction may enhance the representation of some features which are not so apparent in the examined organ along other directions and not so well defined in a perspective display.

Some results of a 3-D image reconstruction by interpolation of a limited set of sections have been obtained by processing only two CAT images of human skull. A small area of 32 by 32 pixels has been selected on the two originals (Fig.1). Fig. 2a represents the sequence of successive slides reconstructed, parallel to the original (x,y) plane, whereas the slides of Fig. 2b are parallel to an orthogonal plane (y,z). The sequence must be viewed from left to right and from the bottom to the top. Although the limited set of original images, the display method makes evident the shape evolution of the structures in the scene.

3. 3-D DIGITAL FILTERING

3-D frequency analysis performed on the original images or on the interpolated 3-D images, is very useful to know the frequency extensions along the three axes and to define the most suitable filtering operations to be applied.To this purpose a 3-D FFT routine has been used.

Because the difficulty to have enough CAT images of the same person, tests have been performed on a suitable wooden model constituted of nine sections and then processed to reconstruct 32 sections of 32 by 32 pixels each. The reconstructed 3-D model has been processed by means of a 3-D filter that is an extension in the 3-D domain of a FIR zero phase filter already used on CAT images in the 2-D space [3], [4]. Such a filter, of parabolic type, has a frequency response constituted by the interconnection of two parabolas having opposite concavity, and may be defined, in 1-D space (Fig.3), by the following equations (w=angular frequency)

$$y_1 = f_1(w) = dw^2 + f \qquad (1)$$
$$y_2 = f_2(w) = aw^2 + bw + c$$

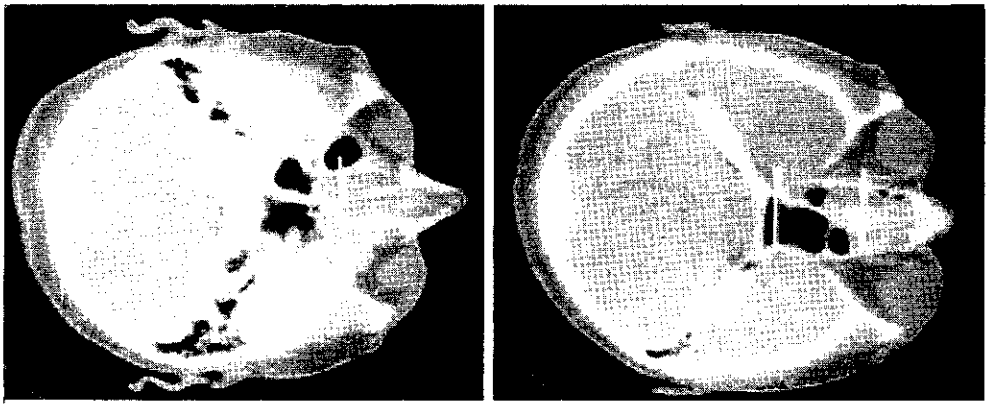


Fig. 1 - CAT images of human skull of the same person

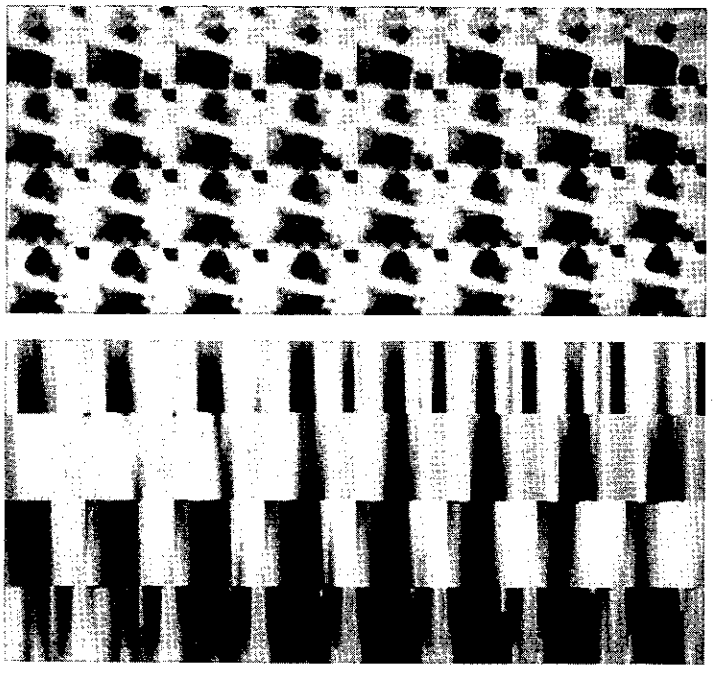


Fig.2 - Sequence of slides interpolated:
a) parallel to (x,y) plane (top)
b) parallel to (y,z) plane (bottom)

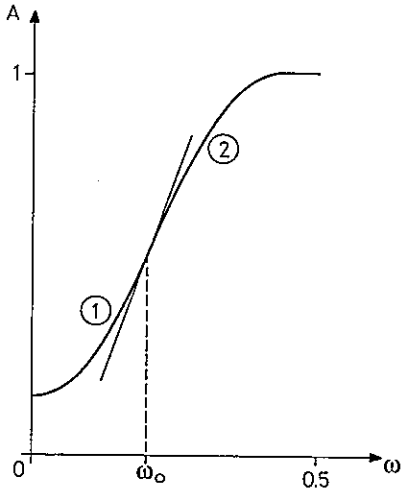


Fig.3 - 1-D frequency response of the designed parabolic filter

For higher design simplicity, practically,such a curve has been defined by choosing only two parameters, the "slope" and the "origin", and by deducing the other coefficients by these ones. The "origin" parameter represents the value of frequency response at w = 0 and therefore determines the performance of the filter at low and very low frequencies. The "slope" parameter denotes the value of the first derivative at the interconnection point between the two parabolas and controls the filter behaviour at higher frequencies.

Assuming the continuity both of the curve and of the first derivative at the interconnection point, and the opposite concavity of the two parabolas, the values of a, b, c, d coefficients and the interconnection point w_o are obtained by means of simple geometric and algebraic considerations; by solving the obtained system of five equations and five unknown variables, we have:

$$a = \frac{-(y')^2}{2(1-f)} \qquad b = 2y' \qquad (2)$$

$$c = 2f-1 \qquad d = -a$$

In Fig. 4a, 4b and 4c some results are reported which were obtained by applying the filter to a 2-D CAT image of a human skull. Fig. 4b shows the result obtained by processing the original image (Fig. 4a) with filter parameters suitable for edge detection, while the slope and origin values of Fig. 4c were chosen in order to obtain an enhancement of the skull features. The filter is particularly efficient on medium-consistency and hard tissues. In 3-D space the frequency response of such a filter is obtained by a 3-D rotation of the generation curve described. It has a spheric symmetry and,being a 4-dimension solid, it cannot be displayed. The developed 3-D parabolic filter has been

Fig.4a- Original CAT image

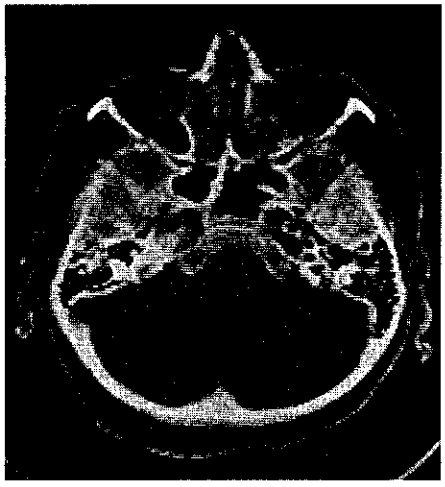


Fig. 4c - 2-D parabolic filtering: origin= 0.25, slope = 4.0

tested on the wooden model with parameter values suitable for contrast enhancement and edge detection.

The filter utilization with parameters suitable for enhancement gives interesting results, not so good results were obtained when using the filter as edge detector owing to the "noise" frequency content of the image.

Fig.4b - 2-D parabolic filtering: origin=0.025, slope = 4.0

4. CONCLUSIONS

The reported results confirm the interest and utility of the used interpolation and representation method; this is particularly true in all that cases in which there is the necessity to examine sections of human body along particular directions. The results obtained by application of the 3-D parabolic filter to the interpolated wooden model, extend digital filtering techniques, already well known and tested in the 2-D domain, into the 3-D space. Further applications of the developed 3-D digital techniques to ultrasonic images are under development.

REFERENCES

[1] Herman,G.T., Udupa,J.K., Display of 3-D Digital Images: Computational Foundations and Medical Applications, IEEE Computer Graphics and Applications, (1983), 39-46.

[2] Webber,R.L.,Nagel,R.N., Three Dimensional Enhancement of Two-Dimensional Images, Journal of Clinical Engineering, (1980), 41-50.

[3] Cappellini,V., Carlà,R., Melani,M., Implementation of a 2-D FIR Digital Filter of Parabolic Type for Biomedical Applications, Proc. of the Internat. Conf. on Digital Signal Processing, Sept. 5-8,Florence, (1984).

[4] Cappellini,V., Constantinides,A.G., Emiliani,P.L., Digital Filtering and Their Applications, Academic Press, London, New York, (1978).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

PARAMETRIC SPECTRAL ANALYSIS OF HEART RATE VARIABILITY; APPLICATION OF KNOWLEDGE BASED MODEL ORDER SELECTION

Otto Rompelman and Richard H.J. Derkx

Delft University of Technology, Delft, Netherlands

1. PROBLEM

Spontaneous fluctuations in heart rate (Heart Rate Variability or HRV) reflect the continuous interaction of the autonomic nervous system and the cardiovascular system. Spectral analysis of HRV shows two dominant regions of activity due to respiration (frequency equal to the respiratory rate) and to oscillations of the baroreceptor reflex control loop (frequency about 0.1 Hz) (fig. 1) [1]. The latter frequency is hardly dependent on subject, age or sex and is mainly determined by the total delay of this control loop. In diabetics with autonomic neuropathy this loop delay is increased resulting in a decreased frequency of oscillation [2]. Therefore the assessment of this frequency may supply useful diagnostic information.

2. BACKGROUNDS

HRV-information can be obtained non-invasively from the electrocardiogram (ECG). First the ECG is converted into an event series, the events being the QRS-complexes. The intervals of this event series are characterized by its low coefficient of variation, CV (order: 0.15) and relatively slow variations. Consequently, since HRV-information is band limited to 0.5 Hz, the event series may be passed through a low pass filter in order to generate a proper HRV-signal [3]. In the case of diabetic neuropathy there are a number of difficulties encountered when trying to carry out spectral analysis of HRV. First the CV of the intervals is reduced (order: 0.05). Hence a very accurate QRS-detection is required in order to avoid distortion of the event process due to random shifts of the events. Secondly, since mainly the para-sympathetic pathways are affected, in particular the spectral components above 0.04 Hz are suppressed. Assuming that the frequency component to be assessed (0.1 Hz) is shifted to lower values this component may be obscured by the rather large and slow variations in heart rate. This latter problem is even more severe if only short data segments (e.g. a few minutes) are available as is often the case in polyclinical practice.

3. METHOD

It was decided to apply AR-spectral analysis to the HRV-data since it is claimed that this method yields a higher spectral resolution than FFT-methods, in particular when dealing with short data segments. A general problem in this approach is the choice of the proper model order p. Though usually a model order criterion is applied (e.g. Akaike's Final Prediction Error criterion), we introduced an alternative approach based on prior knowledge. This knowledge implies the postulated presence of a spectral peak in the range of 0.04 - 0.1 Hz, representing the slowed down oscillations of the baroreceptor control loop. The problem is the assessment of the actual value of this frequency. A series of 40 AR-spectra were derived from an HRV-signal of about 200 sec duration. The AR-coefficients were estimated using the Burg-algorithm. For each hence obtained spectrum the frequency corresponding to the maximum within the range of 0.05 - 0.1 Hz was determined whereafter a plot was made of this dominant frequency as a function of the p. If this curve showed a plateau it was assumed that this plateau corresponds to the frequency of interest. The rationale of this reasoning is the assumption that in this area the estimated frequency is not dependent on p but purely on the data.

4. RESULTS AND DISCUSSION

In fig. 2 some results are shown of one example of the analysis of HRV obtained from a diabetic. Fig.'s 2a, 2b and 2c show the AR-spectra for p=10, 30 and 45 respectively. For p=10 no spectral peak is found in the region of interest. For p=30 a definite peak is present, whereas at order 45 two adjacent peaks are found which is the well-known phenomenon of line splitting. Fig. 2d shows the plot of the assessed frequency as a function of p. Indeed a plateau is present indicating that in this range the value of p is rather indifferent.

After having analysed the HRV data of a number of patients in this way we came to the conclusion that there is a considerable amount of overlap in the location of the plateaus. This indicates that we may end up with one advisable order for the AR-model when applying spectral analysis for determining the degree of autonomic neuropathy on the basis of HRV-analysis.

REFERENCES

[1] R.I. Kitney, O. Rompelman, The beat-by-beat investigation of cardiovascular function, Clarendon Press, Oxford, 1986.

[2] T.J. van den Akker, A.S.M. Koeleman, L.A.H. Hogenhuis, O. Rompelman, Heart rate variability and blood pressure oscillations in diabetics with autonomic neuropathy, Automedica, Vol. 4, 201-208, 1983.

[3] O. Rompelman, A.J.R.M. Coenen, R.I. Kitney, Measurement of heart rate variability: Part I - Comparative study of heart rate variability analysis methods, Med. & Biol. Eng. & Comp., Vol. 15, 233-239, 1977.

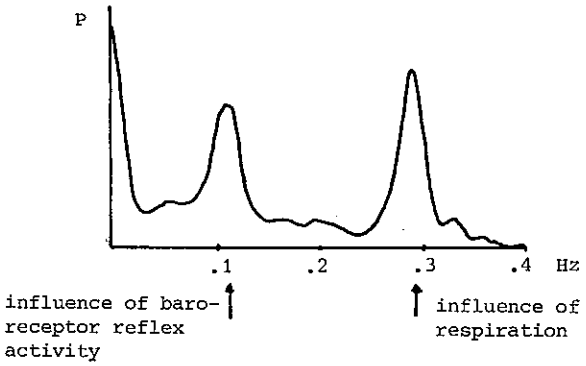


Figure 1

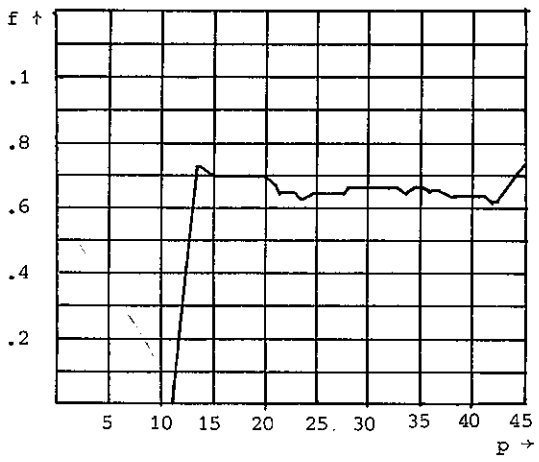


Figure 2d.

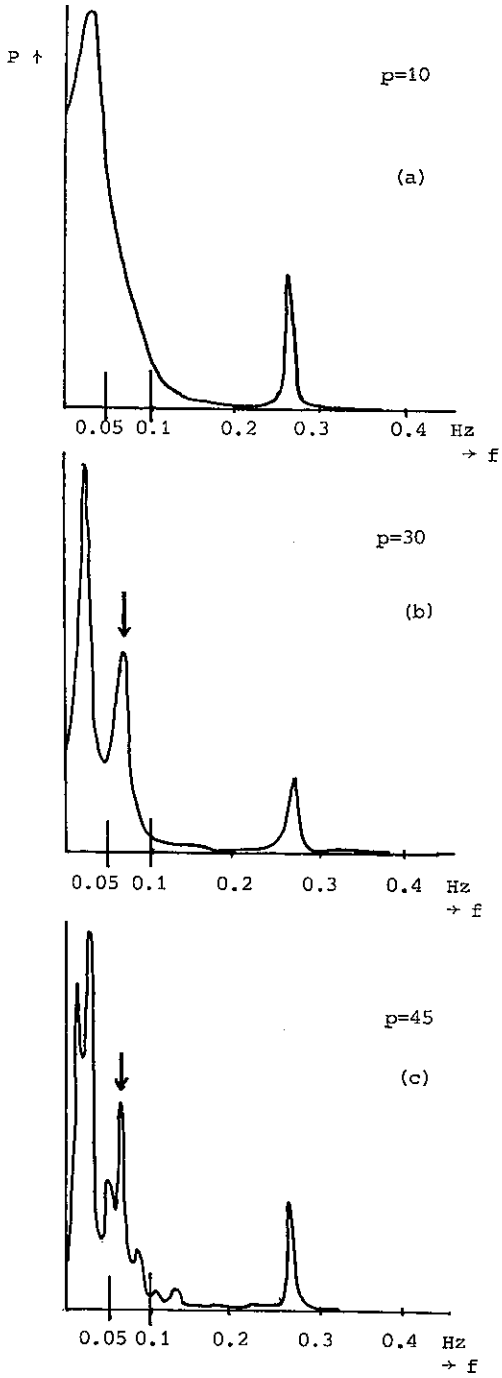


Figure 2 (a-c)

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DYNAMIC CHARACTERISTICS OF THE HUMAN HEART RATE, USING A PSEUDO RANDOM BINARY WORK LOAD

M.J. Coëmet-Penning, Ir M. Woerlee, and Prof. dr. I.T. Young

Pattern Recognition Group, Department of Applied Physics, Delft University of Technology, Delft, and Work Physiology Section, PTT Medical Department, The Hague, The Netherlands

A method for determination of the impulse response of the system controlling the heart rate during physical exercise is proposed. It makes use of a pseudo random binary work load superposed on a constant level of physical exercise, combined with the cross-correlation method for system identification. This enables complete determination of the dynamic characteristics of this system at a specific level in one single experiment on an ergometric bicycle.
Suitable values of the parameters of the maximum length binary sequence are investigated. This method will be used in an attempt to design a new method for the determination of the anaerobic threshold.

1. INTRODUCTION

1.1. Physiological framework of this investigation

It is important in work physiology to measure the fluctuations in the physical fitness of people.
Both excessive work loads and a general decrease in physical fitness among the population may cause problems during work.
During physical exercise, two situations can be distinguished:
- Below a particular level, the energy necessary for the exercise is produced by the aerobic metabolism. Such a level of physical exercise can be performed for a long time.
- Above this level the aerobic metabolism provides insufficient energy and thus energy must be supplied by the anaerobic metabolism. This kind of exercise can only be maintained for a restricted time and afterwards a recovery period is required.

In physiology, this level is called the Anaerobic Threshold (AT). It differs from one person to another and depends on physical fitness.
A practical method for determination of the AT using an ergometric bicycle in a laboratory situation has been used for many years, but it has the disadvantage of being very sensitive to physiological noise. If the subject is not accustomed to doing physical exercises on an ergometric bicycle, the results are often inaccurate.

1.2. System analysis approach

Because of the shortcomings of the current method for determination of the AT, an approach in terms of system analysis is chosen. The purpose of this investigation is to study the low dynamic characteristics of the systems controlling the heart rate, ventilation, O_2 intake, and CO_2 delivery below and above the AT, and to design on the basis of these results a new method for the determination of the AT which is less sensitive to noise influences.
The previous physiological description is interpreted as follows:
1. Below the AT: one LTI system H1 representing the complex physiological mechanisms of the aerobic metabolism.
2. Above the AT: two LTI systems in parallel. The same H1 as below the AT, and a second system H2 representing the mechanisms of the anaerobic metabolism.

The first step is to determine the impulse response of the investigated systems at different levels of physical exercise, using a black box approach, and to examine whether this model fits experimental data.
A method has been designed and will be discussed in this paper. Until now it is only implemented for the system controlling the heart rate. It makes use of pseudo random binary test signals superposed on a constant level. The cross-correlation method is used to determine the impulse response of the system concerned.

2. THEORETICAL DESCRIPTION OF THE METHOD

2.1. System identification using random signals

One of the fields where random signal testing is used, is the identification of the characteristics of control systems, as described in details by Davies [1].
Physiological systems have the same requirements, namely: identification must be performed during normal operation. The test signal must be applied together with the normal operating signal and its amplitude must be small enough to ensure that the system is not disturbed too much with respect to its normal operating level. Another reason for keeping the amplitude of the test signal small is that the system must remain linear in the studied conditions, with a view to reliable application of

system analysis theories.
The essence of random signal testing is that a random signal is applied together with the normal operating signal of the system under investigation. The cross-correlation of the input of the system and its output is calculated. It is an estimate of the system impulse response.

2.2. System identification from cross-correlation function

The cross-correlation function $Rxy(\tau)$ of the input $x(t)$ and the output $y(t)$ of a LTI system $h(t)$ is equal to the time response of the system with the autocorrelation function $Rxx(\tau)$ as input:

$$Rxy(\tau) = h(\tau) * Rxx(\tau)$$

The Fourier Transform then gives the cross power density spectrum $Sxy(\omega)$:

$$Sxy(\omega) = H(\omega) \cdot Sxx(\omega)$$

And the system's frequency response function can be determined by:

$$H(\omega) = \frac{Sxy(\omega)}{Sxx(\omega)} \quad \text{for } Sxx(\omega) \neq 0 \qquad (2.2.1)$$

An arbitrary input signal will often be too weak to permit the use of this method, but a convenient input test signal may be added to the normal input signal of the system under investigation.
Using a input test signal uncorrelated with the original input signal will directly give the system's frequency response function.
A convenient input test signal is white noise, but it has the disadvantage that it requires a long measuring time.
This drawback may be overcome using pseudo random noise.

2.3. System identification using pseudo random binary noise

The maximum length binary sequence (m.l.b.s.), also called pseudo random binary sequence (p.r.b.s.), is such a pseudo random noise signal. In an important sense it acts like band limited white noise.

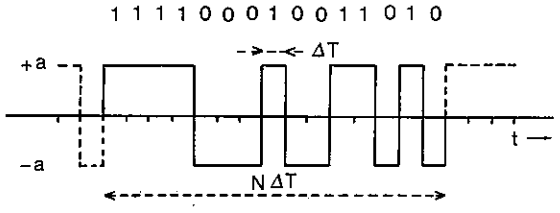


Figure 2.1: Discrete and analog m.l.b.s.

It is a periodic signal which alternates between two levels $x(t) = \pm a$ according to a particular pattern. It can easily be generated with a shift register as described in Davies [1] or Korn [2], or with a computer algorithm. This produces a sequence of N logic states 0 or 1. N is the length of the sequence; it is equal to 2^n-1, where n is called the sequence length generator.
To make it a continuous signal, a time interval ΔT and an amplitude a are chosen. A D/A converter can e.g. be used to achieve that each logic state 0 gives a corresponding value for the associated time signal -a, and each logic state 1 gives a value +a (see figure 2.1).
The autocorrelation function of an m.l.b.s. is an approximation of a pulse with a small d.c. component of $-a^2/N$ as shown in figure 2.2.

$$Rxx(\tau) = \begin{cases} a^2\left\{1 - \frac{|\tau|}{\Delta T}\frac{N+1}{N}\right\} & \text{for } 0 \leqslant \tau < \Delta T \\ -\frac{a^2}{N} & \text{for } \tau \geqslant \Delta T \end{cases}$$

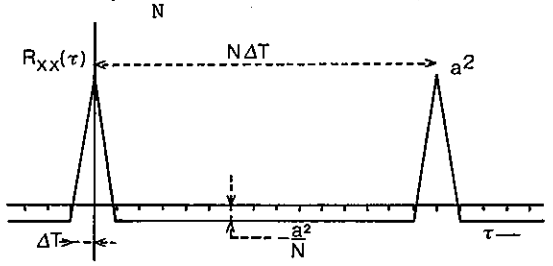


Figure 2.2: Autocorrelation function of an m.l.b.s.

Its power density spectrum is a line spectrum and has an envelope with a shape $(\sin(x)/x)^2$ (figure 2.3). The first zero is reached at $2\Pi/\Delta T$. The distance between two lines in the spectrum is $2\Pi/N\Delta T$.

$$Sxx(\omega) = \frac{a^2(N+1)\,\Delta T}{N} \sum_{r=1}^{N} \left\{\frac{\sin(r\Pi/N)}{r\Pi/N}\right\}^2$$

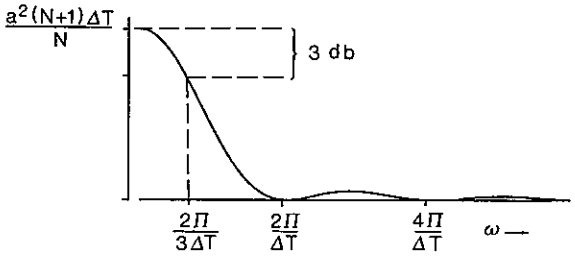


Figure 2.3: Envelope of the power density spectrum of an m.l.b.s.

The -3 db point is at the frequency given by $Sxx(\omega) = 0.7071$ which is approximately $r = N/3$ or $\omega = 2\Pi/3\Delta T$.
ΔT, the smallest time element of the sequence, determines the width of the first lob of the power density spectrum. N determines the number of spectral lines in this first lob, and together with ΔT, they determine the resolution in the frequency domain $2\Pi/N\Delta T$ and the duration of the sequence in the time domain $N\Delta T$.

Assuming that the pulse response has decayed to zero in a time less than $N\Delta T$:

$$Rxy(\tau) = \frac{a^2(N+1)\Delta T}{N} h(\tau) - \frac{a^2}{N} \int_0^{N\Delta T} h(s)\, ds$$

The second term on the right hand side is constant and small for a large value of N. By either assuming that this second term is insignificant, or biasing the cross-correlation with this constant, it can be seen that the cross-correlation is directly proportional to the impulse response of the system.
An important practical point is that a first sequence called the 'settling sequence' is required before the method can be used, to ensure that the transient following the application of the first m.l.b.s. has died out. Only the second and possibly following sequences can be used to determine the impulse response of the system under study. By using more sequences, and averaging the impulse responses obtained, the influence of noise at the output of the system is attenuated.
ΔT has to be chosen a factor smaller than the smallest relevant time constant of the system. $N\Delta T$ has to be chosen large enough to ensure that the impulse response of the system has decayed to zero within this time.

3. REALIZATION OF THE EXPERIMENTAL SITUATION

A Digital MINC-computer system with a 64K memory, using Fortran 4 and Real-11/MNC subprograms is used to communicate with laboratory instruments:
- Digital-to-analog data transfer: It drives the load of the ergometric bicycle, Lode-Ergometer 'Corival-300'. It also produces two pulses at a constant zero level for recognition of the second and possibly following sequences of the experiment. These pulses are stored together with the subject's electrocardiogram on a cassette tape.
- Analog-to-digital data transfer: to gather data during the experiment.
- Real-time clock with two Schmitt Triggers, used to measure off-line the heart rate beat to beat during the relevant part of the experiment. Subsequently, interpolation is performed to obtain equidistant samples.

While the subject performs the pseudo random binary work load pattern on an ergometric bicycle in an upright position with a pedaling rate of approximately 60 rotations per minute, the electrocardiogram is recorded on a cassette tape by an electrocardiogram recorder, 'Siemens - Siretape-C', consisting of a portable recording unit and a display-replay unit.

The data set obtained is transferred via a telephone line to a DEC10 computer system, where it is processed. A Fortran computer program using the Fortran Nag Library for Fourier Transform calculates the power density spectrum and autocorrelation of the input test signal, and the cross power density spectrum and cross-correlation of input and output. It also performs digital filter operations, and calculates the estimated impulse response of the system over one or more sequences, using formula (2.2.1).

4. CHOICE OF THE PARAMETERS OF THE M.L.B.S., AND RESULTS OBTAINED

4.1. Choice development, first approach

The choice of the parameters ΔT, N, and a of the m.l.b.s. is always an optimalization problem and a compromise between the requirements: little disturbance of the system, and high signal-to-noise ratio. In this particular study, physiological constraints must also be taken into account. The duration of the experiment may not be too long, because of the requirement of time invariance of the system during the experiment. Besides, ΔT should preferably not be less than one second, because the pedaling rate is 60 rpm. In that way the load is kept constant for at least the duration of one rotation.
Since the power density spectrum of the heart rate does not exceed about 0.5 Hz (see Rompelman [3]), a value of $\Delta T = 1$ s is chosen. Taking into account the allowable length of an experiment and the fact that the expected value of the time constant of the system is in the order of one minute, a choice of $N = 255$ is made. This gives a distance between two spectral lines of 0.0039 Hz and an experiment duration of 4 min 15 s per sequence. A sampling frequency of 8 Hz is chosen.

4.2. Results and conclusions of the first approach

Six experiments are carried out with three subjects, two males and one female. As regards the estimated frequency responses, the results are in agreement with those found by others, e.g. Rompelman [4], and Kamphuis [5]. A first peak appearing at very low frequencies is usually attributed to the thermoregulation system, a peak at about 0.1 Hz to the blood pressure control system, and a peak between 0.2 Hz and 0.4 Hz to the respiration. An example is shown in figure 4.1. But the estimated impulse responses are completely unusable for parameter estimation because of the occurrence of oscillations of relatively high frequency. This phenomenon is also described by Bakker [6] and attributed to the blood pressure control system.
The low dynamic properties of the system controlling the heart rate are obscured by the influence of the blood pressure and respiration.

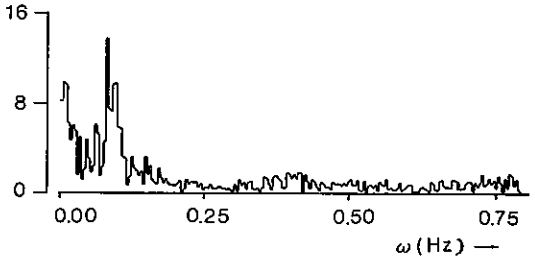


Figure 4.1: Module of the frequency response of the heart rate control system

4.3. Choice adaptation: second approach

Because the low dynamic properties of the system have to be investigated, the blood pressure and respiration control systems are rather disturbing factors.
The requirement that $1/\Delta T$ must be higher than the highest frequency in the system studied, is based on the fact that all frequencies found belong to this system. In this case, frequencies above approximately 0.1 Hz are influences of other regulation systems on the system studied and they tend to mask its response. For this reason, a new value of ΔT = 10 s is chosen. This gives a width of the first lob of 0.1 Hz. To keep the duration of the experiments short enough, is chosen for N = 31. A sampling frequency of 8 Hz is retained to avoid aliasing of the system's frequency response. With these new values of the parameters, the power density spectrum of the m.l.b.s. acts like a low pass filter and the cross power density spectrum is an estimate of only part of the total frequency response of the system. The part of the frequency response of the system occurring at frequencies higher than 0.1 Hz is sacrified so as to eliminate the disturbing influences of the blood pressure and respiration control systems.

4.4. Results and conclusions of the second approach

With one subject several experiments are performed at different exercise levels and using different amplitudes of the m.l.b.s..
Figure 4.2 shows the cross power density spectrum and cross-correlation function obtained with one m.l.b.s. above the AT. A trend is now clearly visible.

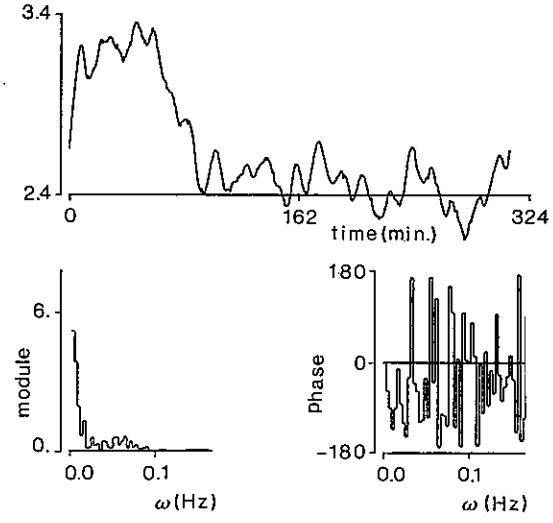


Figure 4.2: Cross power density spectrum and cross-correlation obtained with one m.l.b.s. above the AT

Estimated impulse responses of the system below and above the AT using the average obtained with two m.l.b.s. are shown in figure 4.3. It can be seen that using two sequences decreases the influence of noise.

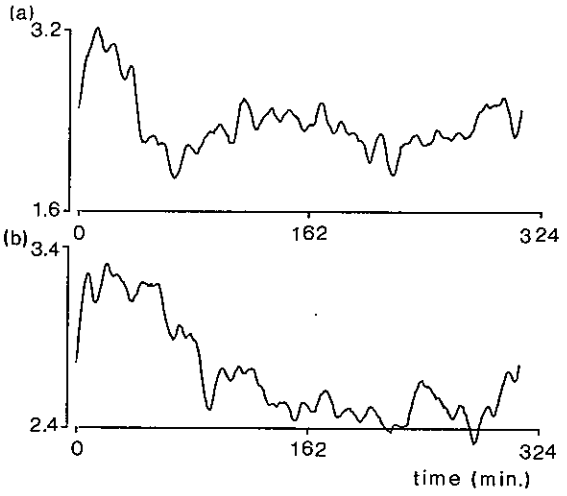


Figure 4.3: Estimated impulse responses: a. below the AT, b. above the AT

5. CONCLUSION

Combination of the cross-correlation method and the use of a pseudo randomly varying work load superposed on a constant level is suitable for determination of the low dynamic characteristics of the system regulating the heart rate during exercise.
Implementation for the systems regulating the ventilation, O_2intake, and CO_2delivery will take place in the near future. The hypothesis that the systems are different below and above the AT will be tested. If the difference is found to exist, a new method will be proposed for determination of the AT using the results obtained.

REFERENCES

[1] Davies W.D.T., System Identification for Self-Adaptive Control
Wiley-Interscience (1970)
[2] Korn G.A., Random-process simulation and measurements. McGraw-Hill (1966)
[3] Rompelman O., A.J.R.M.Coenen , R.I.Kitney Measurement of heart rate variability
Med.& Biol.Eng.& Comput., 1977.15.233-239
[4] Rompelman O., J.B.I.M.Snijders, C.J.van Spronsen, A fast algorithm for the calculation of heart rate variability spectra
International Workshop on the analysis of heart rate variability and blood pressure fluctions. Delft University of Technology
[5] Kamphuis A., Mentale belasting in een codeertaak. Tijdschrift voor Ergonomie
Jrg 10 - No1 maart 1985.
[6] Bakker H.K., R.S.Struikenkamp, and G.A. de Vies, Dynamics of ventilation, heart rate, and gas exchange: sinusoidal and impulse work loads in man.
J.Appl. Physiol., 48(2): 289-301, 1980

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

SEGMENTATION AND CLUSTER ANALYSIS OF TWO-DIMENSIONAL ELECTROPHORESIS IMAGES

S.B.Serpico, G.Vernazza, P.Antognoli, A.Bozzo

Dept. of Biophysical and Electronic Eng.
Via all' Opera Pia 11a, 16145 Genova, Italy

The technique of 2D gel electrophoresis has proved a powerful tool for the differentiation of proteins in the biological area. A full exploitation of the information requires computerized image analysis techniques. Some important aspects are investigated in the following, such as segmentation, cluster analysis, and matching of two gels.

1. INTRODUCTION

The technique of 2D gel electrophoresis has proved a powerful tool for the differentiation of proteins in the biological area. Some thousands of spots can be identified in a gel, each one representing a different protein; the position of a spot and its optical density are connected to the kind and quantity of protein present in the sample [1,2].
In recent years, suitable equipments have been developed to obtain reproducible and standardized gels. However, many problems remain to be solved in order to achieve accurate results by an automatic analysis. Noise effects, spot overlapping, scale distortion, non-linearity of the densitometric response of a film, appropriate spot detection, protein grouping, and matching between gels of similar samples are among the main topics to be considered.
The purpose of this paper is to analyze some basic aspects of these problems, such as the segmentation phase to achieve reliable results in spot area detection; the development of cluster analysis for the detection of the main protein groupings; and a technique for matching images obtained from similar biological samples.

2. SEGMENTATION PROCESS

Some typical 2D electrophoresis images were obtained from autoradiographic film plates by means of a TV camera (Plumbicon). Such images were stored as a matrix of 512x512x8 bits. Each frame (Fig.1) corresponded to 1/16 of the whole gel, so that a 2048x2048 equivalent spatial resolution was utilized (corresponding to 100 μm).
Spot peak points were detected using a technique based on the zero-crossing of the first derivative [3].
As regards the segmentation phase, classical techniques are not suitable for the present application for two reasons: i) variation in gray levels are very smooth near spot edges; ii) the global gray level histogram does not exhibit any sharp valley (threshold) separating

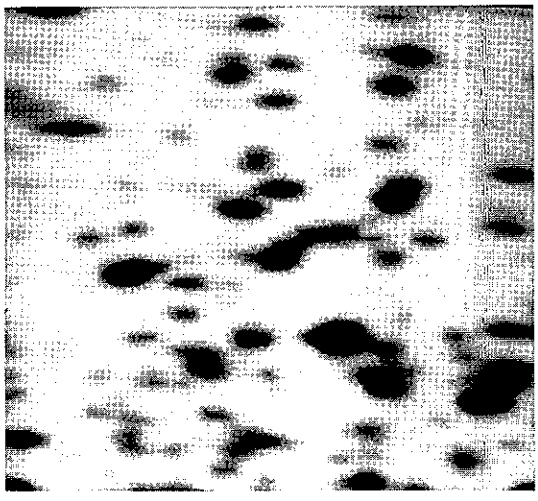


Fig.1. Typical gel image employed in this study (spatial sampling is 100 μm).

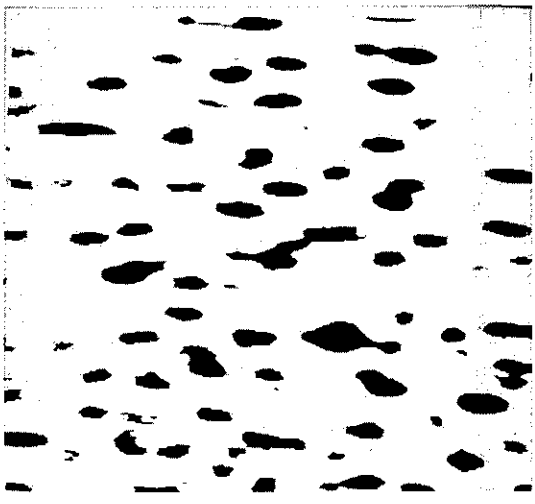


Fig.2. Result of the segmentation process applied to the image in Fig.1.

spots and background.
The gray level distribution in a generic spot delimits a (3D) volume, the external surface of which can be considered as the 3D shape of the spot. A preliminary analysis shows that the gray-level contour lines are elliptical. The 3D shape can be of variable height and width, but it is always of the same characteristic type.
On the basis of this information, a particular procedure for testing the gray level distribution around each point has been devised, which discriminates between background and spot points. The procedure consists of the following steps:
- smoothing of the original image to reduce noise effects;
- three concentric ellipses of different sizes (E1,E2,E3, in decreasing order) are located around each pixel;
- the mean values of the gray levels on their contours are evaluated (L(E1),L(E2),L(E3));
- pixels belong to spots if the following relation holds:

$$(L(E1)>L(E2)+\varepsilon) \text{ AND } (L(E2)>L(E3)+\varepsilon) \quad (1)$$

(where ε depends on the noise standard deviation and has a filtering effect; typical value: 2 to 3).
Otherwise, pixels are assigned to the background.

As a matter of fact, either of disequations (1) is sufficient, however both of them should be employed for a more reliable spot segmentation.
Some typical spots were considered and for each of them various areas were calculated using different ellipse sizes. An interactive segmentation was performed on such spots. The selected ellipse sizes were such as to give very similar results to those of the manual segmentation.
This procedure has been applied to the image in Fig.1, and the results are shown in Fig.2.

3. CLUSTER ANALYSIS

Electrophoresis images contain a large amount of information about the protein content of an analyzed sample [4].
The coordinates of each spot allow proteins to be identified on the basis of two physical characteristics (for instance, pH value and molecular weight). From the optical density the protein content in the sample can be derived (for accurate measurements, calibrated strips should be employed).
In addition to this basic information, other features can be extracted and employed to characterize biological samples. Geometric parameters, like area, perimeter , and shape factors, are typical examples of such features.
Considering the feature space, it is possible to evaluate natural clusters in spot distribution [5], and analyze their biological meaning, in normal and pathological situations.
As a first application, a cluster analysis based on a graphic method (Minimal Spanning Tree (MST)) has been carried out. The starting point is the minimal tree in the feature space; then all branches longer than a threshold are cut, giving rise to different subtrees, i.e. 'clusters'.
Fig.3 shows the application of MST to the spots of Fig.1. In this simple case, the coordinates of the peak points are used as features; the threshold value is 1.3 times the mean branch length.

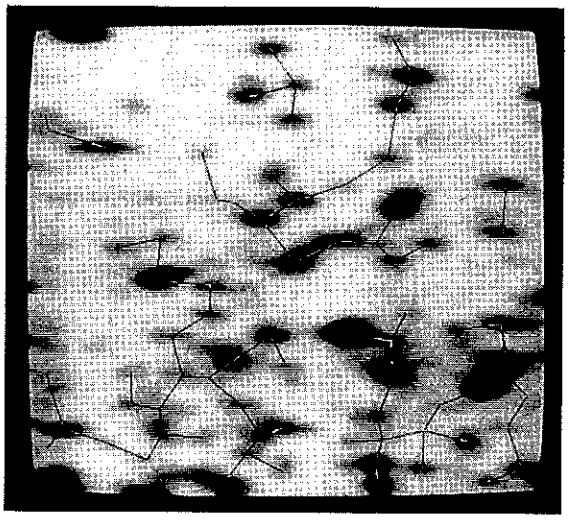


Fig.3. Spot clusters in the original image (MST technique).

4. MATCHING OF TWO ELECTROPHORESIS IMAGES

In analyzing electrophoresis images, it is often useful to make a comparison between a reference image and an investigated one.
A direct comparison of any two gel images is not feasible, because various kinds of distortion and noise affect spot features.
Images of the same sample clearly show this effect. Such a situation has been used as the first meaningful test for a match procedure.
We suggest to adapt a technique commonly used in computer vision for optical flow calculation and motion analysis [6,7]. This way, the problem can be splitted into three steps:
i) find appropriate match points in the two frames;
ii) calculate the match probability for each pair of points belonging to the two frames;
iii) solve uncertain situations and isolated points.

This method is particularly suitable for our application, since spot peak points can be employed as proper and meaningful match points.
Besides two heuristic rules can be applied: corresponding points in two frames cannot move beyond a threshold (T1); even if the global movement between two frames can be elastic, in

small areas such movement are almost rigid.
The number of rows and columns in the matrix of match probabilities is equal to the number of spots in the first and second frames, respectively. The movement related to each element is first calculated and then, by applying the first heuristic rule, most of the elements in each row are deleted, so reducing processing time. Besides, a no-match probability element is added to each row to avoid that small match probabilities grow for normalization effect.
Starting probabilities are computed on the basis of similarities between the feature vectors of each couple of matching spots, whereas no-match probabilities are inversely proportional to the number of elements in their rows. Normalization to 1 is then applied to the rows.
Match probabilities are iteratively updated according to the following relations:

$$\|v_{ij} - v_{kl}\| < Dv_{max} \qquad (2)$$

where vij and vkl are the movement between a point of the first frame (Pk and Pi) and a point of the second one (Pj and Pl); Dvmax is the threshold for an almost rigid local movement (second heuristic rule).

$$Q_{ij}^{n-1} = \sum_k \sum_l P_{kl}^{n-1} \qquad (3)$$

where k is such that Pk and Pi are neighbors; l is such that kl satisfies (2). Pk and Pi belong to the first frame, and they are neighbors if the distance between them is below a threshold (T2).

$$\hat{P}_{ij}^{n} = (A*P_{ij}^{n-1} + B*Q_{ij}^{n-1}) \qquad (4)$$

where A and B are two coefficients which weigh the probability (P_{ij}^{n-1}) of the previous step and the updating part (Q_{ij}^{n-1}), respectively. P_{ij}^{n} at the n-th iteration is obtained by normalizing $\hat{P}_{ij}^{n}$ resulting from (4).
The superscripts (n, n-1) stand for the iteration number.
Only the normalization operation is applied to no-match probabilities.
In this way, match probabilities increase if in the neighboring area there are matches characterized by similar movements and high probabilities. Otherwise, they are reduced by normalization for one of the following reasons: another match element is growing in the same row, or the no-match element will grow, as it is not reduced by the coefficient A (which should be taken less than 1).
Most situations are solved by this procedure, i.e., the right match and no-match probabilities converge to 1, while the wrong ones approach 0.
The third step is the newly developed part. It uses the results of the previous one, and iteratively updates the probability matrix according to the following relations:

$$C_k = 1 - N_k \qquad (5)$$

Nk is the no-match probability and Ck is the sum of the match probabilities in the k-th row.

$$\hat{R}_{ij}^{n-1} = \sum_k \sum_l 0.5 + (P_{kl}^{n-1} - C_k/2) \qquad (6)$$

where k is such that Pk and Pi are neighbors AND Ck is greater than Tk; l is such that kl satisfies (2).

$$R_{ij}^{n-1} = \hat{R}_{ij}^{n-1} / (\sum_k 1) \qquad (7)$$

where k is such that Pk and Pi are neighbors AND Ck is greater than Tk.

$$\hat{P}_{ij}^{n} = E*P_{ij}^{n-1} + F*R_{ij}^{n-1} \qquad (8)$$

where E and F are constants (with the same meaning as A and B).
The condition about k in (6) and (7) selects the voting rows whose total match probability Ck is beyond the threshold Tk.
No-match probability updating and normalization are the same as in step two.
The idea is that a correct match vij should collect most of the available votes (Ck) from its reliable neighbors.
After the third step, match and no-match probabilities should converge to their final values. Some of them could maintain intermediate values in the range from 0 to 1, due to really doubtful situations.
Finally, isolated points (i.e. points of the first frame with no neighbors) are considered. For each of them (Pi), the nearest point (Pk) in the same frame is found among those which exhibit high match probabilities (Pk s.t. there exists $P_{kl} \cong 1$); then the distance (Dik) is computed. Points (Pj) which could match Pi are searched for in the second frame.
The right match (Pj) is selected according to the following relations:

$$\|v_{ij} - v_{kl}\| < (D_{ik} * Dv_{max})/T2 \qquad (9)$$

$$\|v_{ij} - v_{kl}\| = \min_n \|v_{in} - v_{kl}\| \qquad (10)$$

In (9) Dvmax is increased proportionally to Dik.
The above method was applied to two situations: one was created by rigidly shifting an electrophoresis film between two successive acquisitions; the other was the comparison of two gels obtained from the same sample.
35 and 53 spot peak points were selected for

such situations, respectively. In both cases, two errors remained to be corrected after the two first steps, which were accurately recovered by the third one.
In Fig.3 the movement vectors obtained for the second experiment are shown. They depict the complex non-linear distortion effect due to differences in the preparation of the two gels. Ten iterations of each step were sufficient, even if probability values can be considered really stable only after 15 iterations.
The processing time on our HP1000 computer was 25s for 15 iterations in the second step, and 23s for the same number of iterations in the third step.

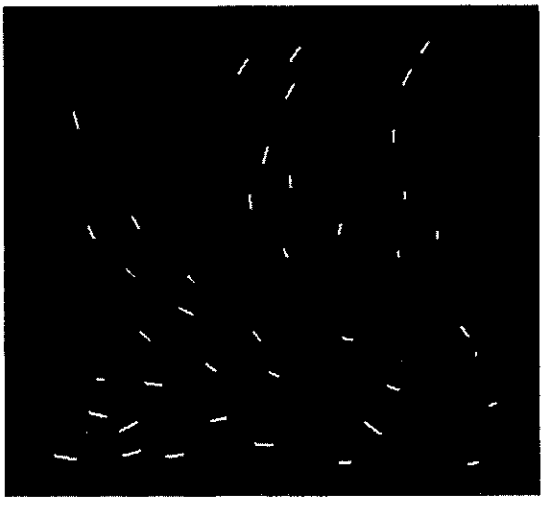


Fig.4. White lines depicts the match vectors. Deformation lines due to complex phenomena are evident.

5. DISCUSSION

Computerized techniques are the basic approach to exploit systematically the information provided by electrophoresis images in clinical, biological, and biotechnological applications.
The proposed segmentation method can be regarded as the combination of spatial filtering (with particular templates), and a thresholding process. The results depends on the Laplacian of the gray-level distribution, but a higher percentage of the total spot volume was extracted, as compared with second derivative zero-crossing techniques.
Obviously, the ellipse sizes are very important parameters which affect the final results. Further investigations could be carried out by employing ellipses of different sizes and by combining the results.
A preliminary example of spot clustering has been presented. Feature selection and an analysis of the results based on biological interpretation of the spot clusters should allow further information to be extracted.
Promising results have been obtained by the proposed technique for gel image matching. In fact, the basic method for optical flow calculation is also suitable for the present application, while the newly developed part increases its reliability.
Applications of the technique to similar (but not the same) biological samples will be the goal of our next investigation.
A useful facility should lie in the assignment of species spot numbers on the basis of the results of the matching between a labelled reference image and unknown ones.

ACKNOWLEDGMENTS

The authors thank the Interdisciplinary Laboratory of Nuclear Medicine, Erasmus Hospital, University of Brussels (Dr. Lecoq), and, the Biochemical and General Physiology Dept., University of Milan (Dr. Alberghina) for supplying the 2D electrophoresis gels.

REFERENCES

[1] O' Farrel, P.O., High resolution two-dimensional electrophoresis of proteins, J.Biol.Chem. 250, pp.4007-4021 (1975).

[2] Garrels, J.J.,Quantitative two-dimensional gel electrophoresis of proteins methods, Enzimology 100, pp.411-423, vol.28 (1983).

[3] Vernazza, G., Serpico, S. B., Giusto, D. and Caredda, A., Computerized analysis of two-dimensional electrophoretic images, SPIE 2nd Int. Techn. Symp. on Opt. and El.Opt. Appl. Sc. and Eng., Dec.1985, Cannes (France).

[4] Celis, J.E. and Bravo, R., Two-dimensional gel electrophoresis of proteins (Academic Press, London, 1984).

[5] Duda, R. O. and Hart, P. E., Pattern classification and scene analysis (John Wiley, New York, 1973).

[6] Ballard, D.H. and Brown, C.M.,. Computer Vision (Prentice Hall, New York, 1982).

[7] Barnard, S. T. and Thompson, W. B., Disparity analysis of images, Technical Report 79-1, Comp. Sc. Dept., Univ. Minnesota, Jan. 1979.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

A FAST ALGORITHM FOR KALMAN FILTERING OF KINEMATIC QUANTITIES

S. Fioretti, L. Jetto and T. Leo

Dept. Electr. and Autom. University of Ancona,
via Brecce Bianche, 60131 Ancona, Italy.

1. INTRODUCTION

The numerical differentiation of a signal is a critical point in Biomechanics and in many other experimental fields. Dynamical analysis of human body segmental movement requires accurate knowledge of velocities and accelerations starting from position data of point body landmarks. It is well known that the derivative operator amplifies the noise corrupting the signal especially for its highest frequency components.
Moreover the estimation of kinematic quantities, as the parameters of the istantaneous axis of rotation, is seriously affected by unaccuracy on the signal knowledge [1].
In [2] a lower bound on the derivatives variance is given with respect to an ideal differentiator.
In order to reduce the noise variance on the signal and on the estimates of its derivatives, two ways may be pursued:
1) to increase the experimental accuracy
2) to adopt more refined filtering and differentiation techniques.
They both have been pursued.
TV or other optoelectronic systems are now widely adopted in biomechanic laboratories. Their accuracy is higher than current stereophotographic techniques. Moreover the statistical characteristics of some of them are gaussian and stationary [3].
Many filtering and differentiation techniques have been proposed in biomechanical literature. They are based on frequency domain techniques as the FIR or IIR filters [4] or on time methods as polynomial and spline approximation [5,6].
In the present contribution a simple Kalman filter based technique is proposed in order to improve already available derivative estimates.

2. METHOD

It has already been experienced that a Kalman filter based on a local Taylor series expansion of the signal and its derivatives can be successfully employed in problems of signal restoring from noisy observed data [7].
We recall that the model has the following structure:

$$X_{k+1} = AX_k + W_k$$
$$Y_{k+1} = CX_{k+1} + V_{k+1}$$

where:

$$A = \begin{bmatrix} 1 & T & T^2/2 & \dots & T^h/h! \\ 0 & 1 & T & \dots & T^{h-1}/(h-1)! \\ & & & \dots & \\ 0 & 0 & & \dots & 1 \end{bmatrix}$$

$$X_{k+1} = (x^{(0)}, x^{(1)}, \dots , x^{(h)})^T_{k+1}$$
$$C = (1, 0, \dots , 0)$$

and W_k and V_k are white, zero mean gaussian noises.
The extension to the contemporary estimation of a signal and its derivatives is a difficult task, due to problems of numerical instability. A possible method to overcome this inconvenient is to use already available derivative estimates as noisy observations in order to improve their accuracy.
The signal to noise ratio improvement (SNRI) introduced by the filter can be estimated as:

$$SNRI = R\,(C\,P_{\infty}\,C^T)^{-1} \qquad (1)$$

where R is the variance of V and P_{∞} is the steady state solution of the Riccati equation.

It can be proven that this figure is always greater than 1. In fact it is easy to verify that:

$$C\,P_{k|k}\,C^T = a - a^2/(a+R) \qquad \text{for every } k \quad (2)$$

where $P_{k|k}$ is the estimate error covariance matrix at step k, and

$$a = \sum_{i=1}^{N+1}\sum_{j=1}^{N+1} (T^{i-1}/(i-1)!)p_{i,j}(k-1|k-1)(T^{j-1}/(j-1)!) \qquad (3)$$

Hence $C\,P_{k|k}\,C^T < R$.

It is also important to know an estimate of the model error due to the local Taylor approximation. An upper bound for this estimate can be obtained under the following general hypothesis on the signal spectrum:

$X(j\omega) = 0 \;\forall \omega \geq \bar{\omega}$

$|X(j\omega)| \leq k \;\forall \omega < \bar{\omega}$

For signal sampled at frequency 1/T it is then possible to write:

$$x(t+T) = x(t) + \sum_{i=1}^{\infty} F^{-1}[(j\omega)^i X(j\omega)]\,T^i/i! \qquad (4)$$

where

$$F^{-1}[(j\omega)^i X(j\omega)] = (2\pi)^{-1}\int_{-\infty}^{\infty} (j\omega)^i X(j\omega) e^{j\omega t} d\omega =$$

$$= (2\pi)^{-1}\int_{0}^{\infty}[(j\omega)^i X(j\omega) e^{j\omega t} + (-j\omega)^i X(-j\omega) e^{-j\omega t}]\, d\omega$$

Moreover

$$|F^{-1}[(j\omega)^i X(j\omega)]| \leq (2\pi)^{-1}\int_{0}^{\infty} |(j\omega)^i X(j\omega) e^{j\omega t} + (-j\omega)^i X(-j\omega) e^{-j\omega t}|\, d\omega \leq$$

$$\leq (2\pi)^{-1}\left\{\int_{0}^{\infty} |(j\omega)^i X(j\omega) e^{j\omega t}|\, d\omega + \int_{0}^{\infty} |(-j\omega)^i X(-j\omega) e^{-j\omega t}|\, d\omega\right\} \leq k\pi^{-1}\int_{0}^{\bar{\omega}} \omega^i d\omega = k\pi^{-1}\bar{\omega}^{i+1}(i+1)^{-1}$$

Truncating the Taylor series expansion (4) at the N-th order, the truncation error h_N is given by

$$h_N \leq \sum_{i=N+1}^{\infty} (T^i/i!)\, k\pi^{-1}\bar{\omega}^{i+1}(i+1)^{-1} =$$

$$= k\pi^{-1}\sum_{j=0}^{\infty} (T^{N+1+j}/(N+1+j)!)\,\bar{\omega}^{N+2+j}/(N+2+j) \leq$$

$$\leq k\pi^{-1}(T^{N+1}\bar{\omega}^{N+2}/(N+2)!)\sum_{j=0}^{\infty} (\bar{\omega}T)^j/j!$$

Hence:

$$h_N \leq k\pi^{-1}(T^{N+1}\bar{\omega}^{N+2}/(N+2)!)\, e^{\bar{\omega}T} \qquad (5)$$

Unequality (5) may be utilized as a project criterium to define the variance of the model error Q in the Kalman filter equations.

3. RESULTS

Kinematic quantities relevant to human movement analysis have been simulated fitting accurate experimental data by orthogonal polynomials. Data are positions of point body landmarks obtained by CoSTEL automatic data acquisition system [8]. Simulated data have been corrupted by gaussian noise with standard deviation equal to the accuracy of the experimental technique.

Estimates of velocity and acceleration have been obtained by means of orthogonal polynomials (OP) and five point second order data fit (5PDF) methods. These estimates have been considered as fictitious observations for our Kalman filter.

It is possible to consider three zero order models for the displacement, velocity and acceleration signals respectively. In this case for the steady state solution of the Riccati equation and for the error model we have respectively:

$$P_{\infty} = (-Q + \sqrt{Q^2 + 4RQ})/2$$

and

$$h_0 \leq (K\,T\,\bar{\omega}^2)\, e^{\bar{\omega}T}/2\pi$$

In fig. 1 are reported the results relative to the velocity and in fig. 2 those relative to the acceleration. The SNRI expressed in dB for the two cases is reported in the following table

	velocity	acceler.
5PDF	2.09	7.36
OP	1.73	0.63

4. DISCUSSION

A fast technique has been proposed in order to improve the accuracy of already available derivative estimates. Experimental results show that this is possible even in the extremely simple case of zero order models. Higher order models do not produce appreciably different estimates. The improvement obtainable depends

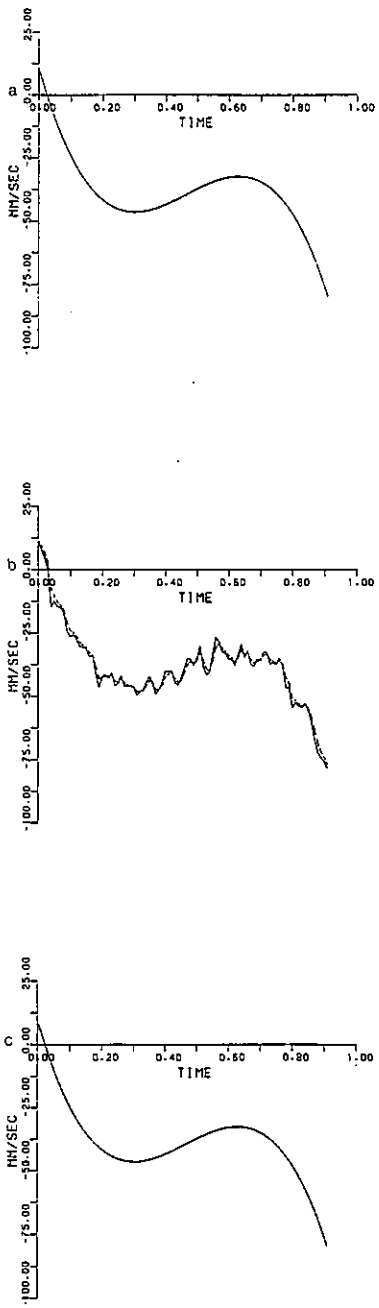


Figure 1: a) simulated velocity. b) estimates of velocity by 5PDF algorithm and Kalman filter. c) estimates of velocity by OP algorithm and Kalman filter. Kalman estimates are dashed lines.

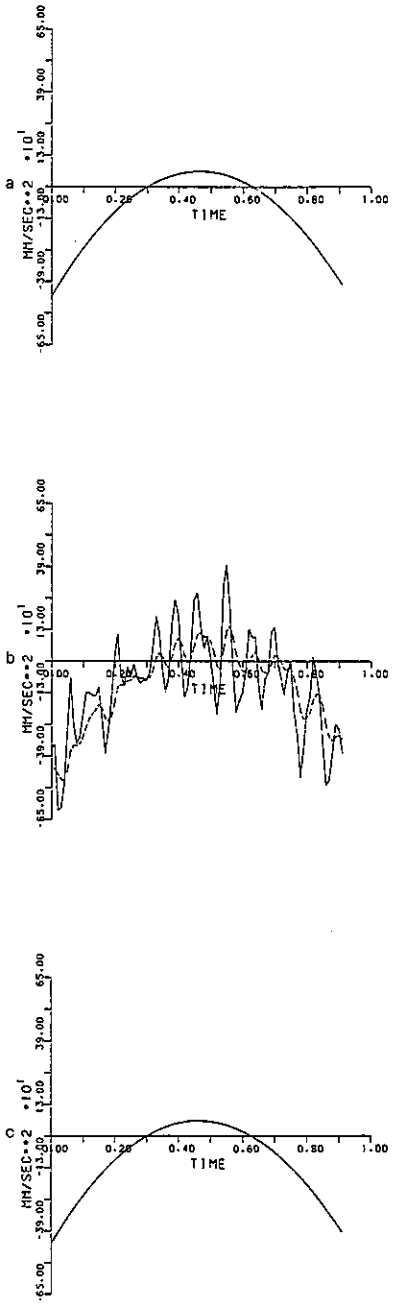


Figure 2: a) simulated acceleration. b) estimates of acceleration by 5PDF algorithm and Kalman filter. c) estimates of acceleration by OP algorithm and Kalman filter. Kalman estimates are dashed lines.

on the initial accuracy of the estimates: as the accuracy increases the SNRI decreases.From (2) derives that R represents the upper bound for P_{∞} so that in any case the filter may introduce an improvement of the signal to noise ratio.
It is clear that relations (1) and (2) represent a tool to "a priori" decide on the opportunity of using the method described.
In general a sequentially correlated incertitude is introduced by the preliminary derivative estimation procedures. The degree of correlation depends on the particular technique used. If it is sufficiently low it can be neglected without an appreciable difference in the filter performance. Otherwise it is opportune to apply decorrelation procedures which usually involve an augmented state vector in the Kalman filter equations [9].

REFERENCES

[1] Woltring H.J., Huiskes R., De Lange A. and Veldpaus F.E. (1985): Finite centroid and helical axis estimation from noisy landmark measurements in the study of human joint kinematics. J. Biomechanics, 18, 379-389.
[2] Lanshammar H. (1982): On precision limits for derivatives numerically calculated from noisy data. J. Biomechanics, 15, 459-470.
[3] Cappozzo A., Leo T. and Macellari V. (1983): The CoSTEL kinematics monitoring system: performance and use in human movement measurements. Int. Ser. Biomech., V. 4B, H. Matsui and K. Kobayashi Eds.
[4] Lesh M.D., Mansour J.M. and Simon S.R. (1979): A gait analysis subsystem for smoothing and differentiation of human motion data. J. Biomechanical Eng., 10, 205-212.
[5] Pezzack J.C., Norman R.W. and Winter D.A. (1977): An assessment of derivative determining techniques used for motion analysis. J. Biomechanics, 10, 377-382.
[6] Wood G.A. and Jennings L.S. (1979): On the use of spline functions for data smoothing. J. Biomechanics, 12, 477-479.
[7] Jetto L. (1985): Small computer procedure for optimal filtering of haemodynamic data. Med. and Biol. Engng. & Comput., 23, 203-208.
[8] Macellari V. (1983): CoSTEL: a computer peripheral remote sensing device for 3-dimensional monitoring of human locomotion. Med. Biol. Engng. & Comput., 21, 311-318.
[9] Jazwinski A.H. (1970): Stochastic processes and filtering theory. Academic Press, New York.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

THALLIUM-201 TOMOGRAPHY; DEVELOPMENTS TOWARDS QUANTITATIVE ANALYSIS

A.E.M. Reijs*, J.H.C. Reiber, P. Fioretti, J.J. Gerbrands**, M.L. Simoons, P.P.M. Kooij*

Thoraxcenter and Dept. Nucl. Med.*, Erasmus University Rotterdam, and ** Information Theory Group, Delft University of Technology, The Netherlands.

Thallium-201 (Tl-201) tomography allows the computation of the three-dimensional (3D) distribution of TL-201 within the myocardium. In this paper our approaches towards quantitative analysis of the tomograms are described. Following standard transaxial reconstruction slices perpendicular and parallel to the long axis of the left ventricle (LV) are reconstructed. The orientation of the (LV) is determined by having the operator indicate the LV long axis in two orthogonal projection images. Algorithms have been developed for the detection of the 3D inner and outer boundaries of the LV myocardial wall with an algorithm based on maximal boundary strength criteria using a spherical coordinate system. On the basis of circumferential profiles computed within the oblique and sagittal cross sections the boundaries of perfusion defects can be determined. From these data volume and mass of myocardial regions with defects can be computed.

1. INTRODUCTION

Thallium-201 early and late postexercise planar scintigraphy is now widely used for the detection and evaluation of patients with coronary artery disease. Software packages with different degrees of sophistication for the quantitative analysis of the resulting images have been developed and are now available for most nuclear medicine computer systems (1,2). However, despite the objective analyses of the myocardial perfusion images, the planar technique is limited by the fact that the two-dimensional images are projections of three-dimensional structures, with the consequence that overlap of normal, ischemic and infarcted areas may occur in the images. As a result, the correlation between the location of perfusion defects and the site of the obstructions in the coronary arteries responsible for the perfusion defects has remained relatively poor. Tomographic approaches have been proposed to circumvent these limitations. Through tomographic reconstruction methods, the myocardial Tl-201 distribution in cross sections of the heart can be assessed with a high target-to-background ratio. The goal of our study has been the computation of the location, volume and mass of ischemic and infarcted areas in the myocardial muscle by quantitative analysis of early and late postexercise thallium-201 tomograms acquired with a rotating gamma camera. In this paper we will present and discuss our approaches and developments towards such analysis. Another method for the quantification of the relative 3D distribution of Tl-201 in the myocardium has been described by Garcia et al. (3).

2. METHODS

2.1. Data Acquisition

The Tl-tomograms are acquired using standard techniques. The patient performs a maximal or symptom limited exercise test on a bicycle ergometer. One minute prior to maximal exercise 1.5 mCi Tl-201 is administered intravenously.
In our institute a Siemens Gammasonics single-head Rota-camera equipped with 37 photomultiplier tubes, a 3/8-inch NaI(thallium) crystal, and a low-energy all-purpose collimator is used. A 20% window is centered on the 80 and 167 KeV photopeaks of the isotope. Thirty projections with 6° increments (180° scanning) are acquired with an acquisition time of 1 min per projection. The digitizing matrix (64x64, word mode) is selected in the mid portion of the image by using a zoom factor of $\sqrt{2}$.
For purposes of comparing the new tomographic results with the results from our established quantitative approach for planar Tl-201 scintigrams (2), two planar views (700,000 counts full field) in the anterior and LA045 views are obtained with the Rota-camera immediately prior to the tomographic acquisition. After 4 hours late postexercise imaging is performed in the same sequence. The acquisition and processing of the images are performed on a DEC Gamma-11 computer system.

2.2. Data Analysis

After collection of all planar and tomographic views from the early and late postexercise studies, the data analysis is performed. The planar images are analyzed quantitatively using our software package for the analysis of

planar scintigrams (2).
A number of the basic ideas that we have developed for the quantitative analysis of the planar scintigrams can be applied to the three-dimensional tomograms. Therefore, we propose to implement the following steps for the quantitative analysis of Tl-201 tomograms:

1. reconstruction of the transversal slices with the SPETS-package (Karolinska Hospital);
2. determination of the LV long axis from two orthogonal projections;
3. reconstruction of oblique slices perpendicular to the LV long axis and sagittal slices parallel to the long axis;
4. contour detection of the endocardial and epicardial boundaries of the LV activity structure in the oblique slices;
5. computation of circumferential profiles in oblique slices;
6. construction of a socalled Bull's Eye Display.
7. determination of the boundaries of the Tl-201 perfusion defects in the oblique slices;
8. computation of location, volume and mass of regions with perfusion defects.

Until to-date we have implemented the first four steps, step 6 and the last step of this proposed analysis procedure; these will be described in some detail hereafter.

2.2.1. Three-dimensional reconstruction

For the three-dimensional reconstruction of the transaxial tomograms we use the commercially available SPETS-package which is based on a filtered backprojection technique. The resulting tomograms are directed parallel or perpendicular to the rotational axis of the camera.

2.2.2. Determination of the left ventricular long axis from two orthogonal views

For purposes of comparing circumferential profiles with normal ranges, a standardization procedure must be defined for the slices in which these profiles are to be assessed. It will be clear that oblique slices perpendicular to the LV long axis and sagittal slices parallel to the long axis are to be preferred. Such tomograms can be interpreted more easily and in an intuitively appealing manner. The three-dimensional direction of the LV long axis can be determined from two orthogonal views. The coordinate system (X,Y,Z,O) that is implicitly defined by the reconstruction of the transaxial slices is not very suitable for carrying out the rotation. The origin O of this coordinate system is defined at the right lower corner of the cube of 64x64xn pixels, with n the number of reconstructed transversal slices (Fig. 1). To obtain oblique tomograms a new coordinate system must be defined with respect to the patient's body.
To describe the rotation in simple mathematical terms the origin O' of the new coordinate system must be defined at the point of rotation. An appropriate choice is to define the origin of the new coordinate system at the midpoint of the LV long axis. This new left-hand coordinate system (X',Y',Z',O') is then defined with respect to the body as follows: (Fig 1).

- the origin O' of the coordinate system lies at the midpoint of the LV long axis; this origin is also the point of rotation.
- the positive X'-axis points in the direction of the right arm of the patient.
- the positive Y'-axis points to the back of the patient.
- the positive Z'-axis points to the head of the patient.

Since the origins O' and O of the new and old coordinate systems, respectively, do not coincide, both a rotation and a translation are required. A translation vector $\vec{s}$ can be defined from O to O' (Fig. 1). Let $\vec{p}'$ be a vector in the new coordinate system (X',Y',Z',O') to a point P in a plane perpendicular to the Z-axis, then the coordinates of the vector $\vec{p}$ after transformation in the original system (X,Y,Z,O) can be computed as follows:

$$\vec{p} = [R] . \vec{p}' + \vec{s}, \qquad (1)$$

where $[R]$ is the rotation matrix. This matrix $[R]$ defines rotations over angles α and β, α being the angle between the long-axis and the

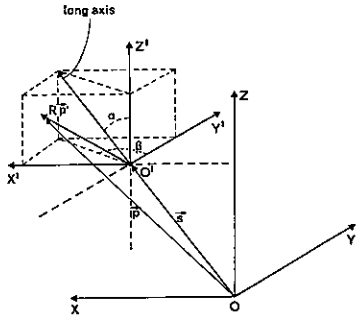


Fig. 1. Definition of the two coordinate systems (X,Y,Z,O) and (X',Y',Z',O').

positive Z'-axis and β the angle between the projection of the long-axis on the X'Y'-plane and the positive Y'-axis, respectively. The rotation matrix $[R]$ is defined as follows:

$$[R] = [R]_{-\beta} [R]_{-\alpha} [R]_{\beta} \qquad (2)$$

The rotation over β is only necessary to have an well defined rotation over α. For a left-hand system the rotation matrices $[R]_{\alpha}$ and $[R]_{\beta}$, define a rotation in a clockwise direction around the X'- and Z'-axes, respectively, as viewed from a point along the positive axis

facing the origin,

$$[R]_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} [R]_\beta = \begin{bmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

The angles α and β can be determined from the projections of the long axis on the XOZ- and the YOZ-plane. In each of the two orthogonal views the user defines with a joystick the center and the direction of the LV long axis. From these two indicated lines, the 3D direction of the LV long axis can be computed.

3. RECONSTRUCTION OF THE SLICES PERPENDICULAR AND PARALLEL TO THE LV LONG AXIS

Now that the 3D direction of the LV long axis is known, the activity values for all pixels in the new coordinate system (X',Y',Z',0') can be determined by applying formula (1). If the computed coordinates in the (X,Y,Z,0) system are noninteger, a linear 3D interpolation between the values of the eight neighboring points is determined. After the oblique slices are copied to a SPETS-reconstruction file, the SPETS-package is used to reconstruct slices parallel (sagittal) to the long axis.

4. CONTOUR DETECTION OF THE LEFT VENTRICULAR INNER AND OUTER BOUNDARIES

We were interested in developing a 3D contour detection algorithm that should satisfy the following three requirements:

a. the algorithm should find both the inner and the outer boundaries of the myocardium;
b. the algorithm should be sufficiently robust so that the boundaries of the myocardial wall can also be detected automatically at the locations of perfusion defects (at least for regions with decreased, not absent uptake);
c. the algorithm should also work satisfactory in images with poor signal/noise ratios.

Based on our excellent experiences with a 2D contour detection algorithm that we have implemented and used routinely for Tl-201 planar and Tc-99m gated blood pool scintigrams, we decided to attempt to expand this algorithm to three dimensions (4-6). In short, the 3D-image was transformed into a spherical coordinate system with respect to the approximated LV center. By resampling the 3D-data along radial lines, a total of 64 halfplanes were defined. By applying for each pixel the first derivative with respect to the radial distance R, a cost matrix could be computed for each half plane. The endo- and epicardial boundaries were defined by searching for the maximal and minimal cost paths in these cost matrices, respectively. Finally, a retransformation of the detected boundary positions to the (X,Y,Z,0) coordinate system was performed. The different steps have been described in more detail elsewhere (7). Fig 2 is an example of the detected endo- and epicardial boundaries in an oblique slice of a study.

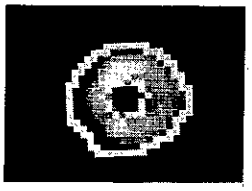


Fig. 2. Example of detected inner and outer boundaries in an oblique section of a patient study.

From evaluation studies with phantom studies of the LV and with patient studies it became apparent that the pseudo 3D contour detection technique is hampered by a number of problems caused by the transformations between the spherical and cartesian coordinate systems and especially by the fact that these transformations require resampling. Although solutions may be found to handle these problems, it will increase the complexity and thus computation time of the contour detection technique. For that reason we have decided not to further investigate the pseudo 3D contour detection technique, but rather apply the minimal cost contour detection technique to the oblique cross sections, thereby making use of expectation windows to limit the search from one cross section to the other.

6. BULL'S EYE DISPLAY

To quantitatively determine the location and severity of perfusion defects in the oblique and sagittal slices, circumferential profiles are computed in each slice (step 5). A circumferential profile is defined by the maximal activity levels along 60 radial segments within the detected endo- and epicardial boundaries of the slice. By comparing corresponding circumferential profiles from the early and late postexercise studies, the location and size of perfusion defects can initially be marked by the user of the system and in the near future be determined automatically. To present the circumferential profiles of the different slices and subsequently derived parameters, such as size of the perfusion defects, thallium-washouts etc. in a readily interpretable manner, a socalled Bull's Eye Display has been developed (3).

7. CALCULATION OF THE LOCATION, VOLUME AND MASS OF AREAS WITH PERFUSION DEFECTS

Once the endo- and epicardial boundaries and the radial segments with perfusion defects are known (step 4,5) the volume and mass of these defects can be computed. Because the endo- and epicardial boundaries and the size of the defects are more easily defined in the polar- than in the cartesian-domain the volume calculation is performed in the polar domain as well. The endo- and epi-cardial boundaries are represented in the polar domain by the detec-

ted contour paths(7) and the size of the defect by two radial angles α_1 and α_2 (Fig 3). The angular resolution depends on the angular sampling$\Delta\varphi$, while the maximal length of radius R is defined by the matrix size. Typical values for the total number of angles and the maximal value of R are 60 and 32, respectively.

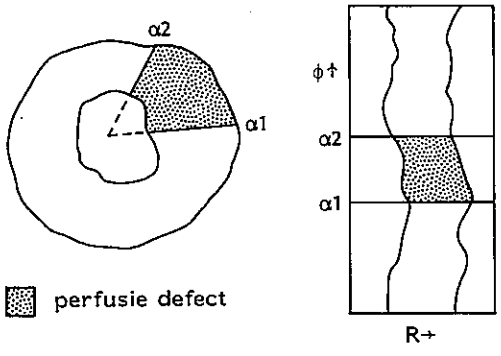


Fig. 3. Oblique slice with perfusion defect in the cartesian and polar domains.

The calculation of the volume of one particular oblique slice proceeds as follows: the boundaries in the polar domain are scanned from α_1 to α_2 (Fig. 3), being multiples of$\Delta\varphi$. The total volume of a perfusion defect in an oblique slice can be defined by summation of circular segments:

$$\Delta V = \pi d . \frac{\Delta\varphi}{360^0} \sum_{i=\alpha_1}^{\alpha_2} (R^2_{epi}(i) - R^2_{endo}(i)) \quad (4)$$

with

ΔV = volume of perfusion defect (cm^3)
d = slice thickness (cm)
α_1 = angle at which perfusion defect starts(1,2.....60)
α_2 = angle at which perfusion defect ends(1,2.....60)
R_{epi}= radius epicardial boundary (cm) (1,2...Rmax)
R_{endo}= radius endocardial boundary (cm) (1,2...Rmax)
$\Delta\varphi$ = angular sampling (degrees).

For each slice the volumes of one or more areas with perfusion defects can be calculated with formula 4. After all the areas with a defect have been calculated and it has been decided which slice areas belong to one great perfusion defect, the total volume of this defect is simply the sum of the individual slice areas. The total volume of the LV can be calculated by summing the volumes between the endo-and epicardial boundaries (i=1,2.....360/$\Delta\varphi$). Finally volume of the defects can be expressed in percentages of the total myocardial volume. The mass of perfusion defects can simply be determined from the calculated volumes by multiplying with the specific weight of the myocardium (1.05 g/cm^3).

CONCLUSION

In this paper our approaches towards the quantitative analysis of Tl-201 tomograms have been described. At the present time, the two-dimensional contour detection technique in the individual oblique cross sections is being implemented and evaluated. In addition, the software routines for the circumferential profiles are being completed, as well as the Bull's Eye Display. Also, planar and tomographic early and late postexercise thallium-201 tomograms are being acquired in a clinical research study. The oblique and sagittal cross sections are interpreted visually; a scoring system has been developed that allows the assessment of the size and severity of perfusion defects from the visually interpreted results. The tomographic diagnosis will be compared with the quantitative results from the planar thallium-201 scintigraphy.

Acknowledgements

The authors wish to thank Mrs. S.M. Spierdijk and Mrs. M.J. Kanters-Stam for their secretarial assistance in the preparation of this manuscript. This project has been supported in part by the Dutch Heart Foundation under grant NHS 82043 and 84074.

References

1. Garcia E, Maddahi J, Berman D, Waxman A. Space/time quantitation of thallium-201 myocardial scintigraphy. J. Nucl. Med. 22: 309-17.1981
2. Reiber JHC, Lie SP, Simoons ML, Wijns W, Gerbrands JJ. Computer quantitation location, extent and type of thallium-201 myocardial perfusion abnormalities. In Proc. First Intern. Symp. on Medical Imaging and Image Interpretation ISMIII - 1982, 123-28. IEEE Cat. No. CH1804-4.
3. Garcia EV, Van Train K, Maddahi J et al. Quantification of rotational thallium-201 myocardial tomography. J. Nucl. Med. 26: 17-26. 1985.
4. Blokland K. Three-D contour detection in emission-tomographic images of the heart. M.Sc. Thesis, Delft University of Technology, 1982 (in Dutch).
5. Gerbrands JJ, Hoek C, Reiber JHC, Lie SP, Simoons ML. Minimum cost contour detection in technetium-99m gated cardiac blood pool scintigrams. Comp. Cardiol.: 281-84. 1982.
6. Reiber JHC, Lie SP, Simoons ML, et al. Clinical validation of fully automated computation of ejection fraction from gated equilibrium blood-pool scintigrams. J Nucl Med 24 : 1099-1107. 1983.
7. Reiber JHC, Reijs AEM, Gerbrands JJ, Simoons ML, Kooij PPM. Thallium-201 tomography: Developments towards quantitative analysis. Comp Cardiol 1985 (in press).

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DENSITOMETRIC ANALYSIS OF CORONARY ARTERIES

C.J. Kooijman, M.Sc., R. Kalberg, M.Sc., C.J. Slager, F.O. Tijdens, BSc, J. van der Plas, B. Sc., J.H.C. Reiber, Ph.D.

Laboratory for Clinical and Experimental Image Processing, Thoraxcenter, Erasmus University Rotterdam.

Quantitation of the severity of coronary obstructions from automatically detected contours is limited to the assessment of the obstruction diameter and percent diameter stenosis in that particular view. Even if an obstruction is analyzed from two views, the computed cross-sectional area does not provide a reliable measure for asymmetric lesions. A densitometric technique is described which attempts to provide measures about cross-sectional areas from a single view on the basis of the measured brightness levels within the arterial segment. Phantom studies show that the accuracy of the assessment of percent area stenosis of obstructions equals 2.79% (s.d. 1.76%) with a computed densitometric transfer function.

1. INTRODUCTION

Previous studies on the hemodynamic effects of obstructions in an artery have shown that the most critical determinant of the severity is the minimal luminal cross-sectional area (1). Assessment of the cross-sectional area and percent area reduction in a stenotic area from the diameter measurements obtained from a single angiographic view assumes a circular cross section, an assumption that is not always correct (2). Even a technique of quantitating area stenosis from 2 orthogonal views and computing the area based on an elliptical model would fail to describe an asymmetric lesion accurately.
However, if a relation between the local thickness of the irradiated object and the resulting brightness level in the angiographic image could be established for each individual pixel, the true luminal cross sections of a contrast-filled coronary artery could be computed, even from a single X-ray projection. This approach is being referred to as the densitometric analysis technique. We have expanded our routine computer-based analysis procedures for the assessment of the arterial dimensions from 35mm coronary cineangiograms with the densitometric analysis of coronary obstructions. It is the purpose of this paper to briefly describe the architecture of the Cardiovascular Angiography Analysis System (CAAS) and the analysis procedure for the assessment of the arterial dimensions, followed by a presentation of the basic principles underlying the densitometric technique, and the implementation of the densitometry. Finally, some initial results are presented and discussed.

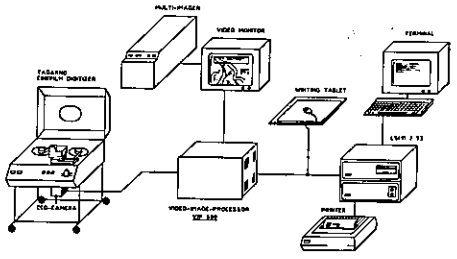


Fig. 1. Block diagram of the CAAS.

2. ARCHITECTURE OF THE CAAS

The CAAS consists of five basic components: a cinefilm digitizer, a video image processor (VIP-500), an LSI 11/73 minicomputer with terminal, which runs under the multi-user/-multi-tasking RSX-11M Operating System, a writing tablet, and application software packages (3-5). A block diagram of the system is given in Fig. 1. A selected region-of-interest in a cineframe to be analyzed is digitized by a charge-coupled device (CCD-camera), stored in the video image processor and displayed on a video monitor. Digital image data are transferred to the host computer for subsequent analysis. At several crucial points in the data acquisition and analysis procedures user interaction by means of the writing tablet is possible. A video imager is used for documentation purposes.
A very important component in the entire image digitization and analysis chain is the cine-digitizer, which consists of a standard cine-projector modified for high-resolution digitization of a selected cineframe (6,7). An array of LED's is used as the monochromatic light source. Any 6.9 by 6.9 mm area in a selected cineframe (18x24 mm) can be digitized by a CCD

camera with a resolution of 512 by 512 pixels and 8 bits of grey levels.

3. CONTOUR DETECTION AND ANALYSIS

The contour detection procedure has been described extensively elsewhere (3-5). It basically consists of the following steps: 1) computation of the calibration factor by means of the contrast catheter displayed in the image; 2) automated boundary detection of the arterial segment; 3) correction of the contour positions for pincushion distortion from the image intensifier; 4) computation of absolute and relative parameters describing the severity of a coronary obstruction.
The result of such an analysis is presented in Figure 2. The obstruction diameter and the percent diameter stenosis with respect to a user-defined reference point were found to be 1.29 mm^2 and 67%, respectively.

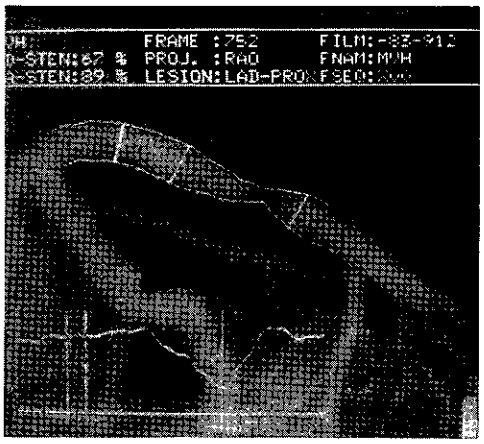


Fig. 2. Example of the assessment of the stenosis severity in terms of diameter measurements in this view.

4. BASIC PRINCIPLES DENSITOMETRIC ANALYSIS PROCEDURE

Sofar we have only analyzed the arterial segment in terms of changes in arterial dimensions in a particular angiographic view. However, there is much more information available that could be applied advantageously, namely the contrast density within the arterial segment.
Constitution of the relationship between irradiated object path lengths and brightness values requires detailed analysis of the complete X-ray/cine/image digitization chain (6). A simple block diagram of this chain is presented in Figure 3. In a simplified approach, we are only interested in the static properties of the system. For the first part of the chain from the X-ray source to the output of the image intensifier we use Lambert-Beer's Law for the X-ray absorption and apply certain

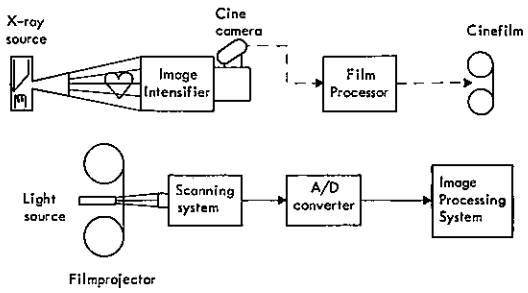


Fig. 3. Block diagram of the X-ray/cine/image digitization chain.

models for the X-ray source and the image intensifier. For the second part of the chain from the output of the image intensifier up to the brightness values in the digital image a transfer function is used. Two approaches to obtain this transfer function have been implemented and evaluated.
The first approach is based on a theoretical description of the individual transfer functions of the components in the cinefilm/image digitization chain (6). This analysis leads to a logarithmic transfer function. The second approach is based on the actual measurement of this transfer function from 21 calibrated density frames, which are processed photographically simultaneously with the rest of the coronary cinefilm, so that the film development process is identical for both the calibration frames and the clinical coronary cineframes. These 21 density frames are exposed homogeneously with a sensitometer having the same color temperature as the output screen of the image intensifier. By means of this calibration procedure many nonlinear effects as well as the daily changes in the cinefilm processing and the film digitization system are taken into account. The measurement procedure for the transfer function is explained in section 5, Measurements.
The percentage cross-sectional area reduction is then obtained as follows (4,6). First, the contours of the arterial segment are detected automatically. Then, the brightness profile along a scanline perpendicular to the local centerline direction is transformed into an exposure profile by means of the selected transfer function. The background contribution is estimated by computing the linear regression line through the background points directly left and right of the detected contours. Subtraction of this background portion from the absorption profile within the arterial contours yields the net cross-sectional absorption profile. Integration of this function results in a measure for the cross-sectional area at the particular scanline. Applying the same procedure for each scanline results in the densitometric area function for the segment analyzed.

5. MEASUREMENTS

In this section a number of measurement and validation procedures are described in more detail. The first, and probably the most important one, is the measurement of the transfer function of the cinefilm/image digitization chain. This transfer function is measured by means of 21 calibrated density frames. These calibration frames are exposed according to an exponential function:

$$E(n) = K.2^{-n},$$

where E(n) is the light exposure level, K a constant and n the frame sequence number ($0 \leq n \leq 20$).
Because of the large range of density levels that must be digitized, a special procedure has been developed for the assessment of the transfer function. Each of the calibrated density frames is digitized with the CCD-camera using a suitable LED light intensity. With suitable we mean that the brightness values in the digitized image are in the range of 200 to 255. For the frames with higher density (darker images) this objective cannot be met anymore, because the LED light intensity will become the limiting factor; for these frames the maximally available light intensity is used.
For each frame a mean brightness value is measured within a rectangle of 20 by 20 pixels in the center of the digitized image. Each of the recorded brightness values V_m is then transformed into a new brightness value V_d which would have been obtained if the particular density frame was digitized with a so-called reference LED light value. The resulting brightness value V_d therefore is calculated according to:

$$V_d = V_m \cdot \frac{Pref}{P_o}$$

where V_m is the measured brightness value, P_o the used light intensity and P_{ref} the reference light intensity. The light intensity is given in dimensionless units between 0 and 127. P_{ref} is usually taken as 64. An example of a transfer function obtained by the described method is given in Fig. 4. On the horizontal axis the logarithm of the exposure of the cineframe is given and on the vertical axis the transformed brightness value, which is a measure for the opacity of the film. The homogeneity of this imaging chain was measured by selecting an appropriate LED light value and digitizing an image without a cineframe positioned between light source and camera. The digitized matrix was divided into submatrices of 20x20 pixels, resulting in 25 submatrices in the vertical and horizontal directions. For each submatrix the average brightness value of the 20x20 pixels was computed and the results displayed in a pseudo three-dimensional representation (Fig. 5). Analysis of the obtained data shows that the inhomogeneity is better than 5%.

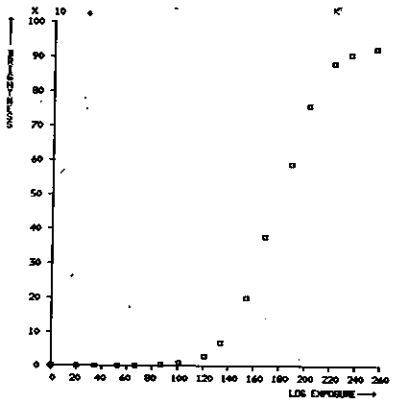


Fig. 4. Example of transfer function of cinefilm/image digitization chain.

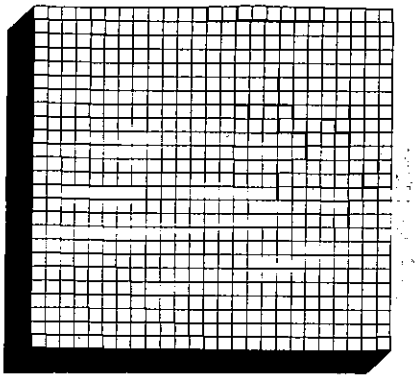


Fig. 5. Pseudo three-dimensional representation of the homogeneity of the image digitization chain.

The homogeneity of the entire X-ray/cine/image digitization chain has been measured by filming copper plates of varying thicknesses at various kilovoltages. Fig. 6 shows an example of an irradiated copper plate of 0.5 mm filmed at 62 kV. By comparing Fig. 5 with Fig. 6, it may be concluded that the distortion due to vignetting, veiling glare and other disturbing factors in the X-ray chain is much larger than the distortion due to the inhomogeneity of the cine-digitizer. Therefore, correction for the inhomogeneity of the cine-digitizer does not seem necessary as long as no corrections are applied for vignetting, etc. To validate the densitometric method described a cinefilm of a perspex model of coronary obstructions filled with different concentrations (50% and 100%) of contrast agent and filmed with a 7" image intensifier at different kV's was analyzed. The perspex model has 17 different diameters ranging from 0.3 mm to 6 mm. From this model percentages densitometric area stenosis as described in section 4 were assessed. Using the densitometric transfer function a mean absolute error between the measured and true

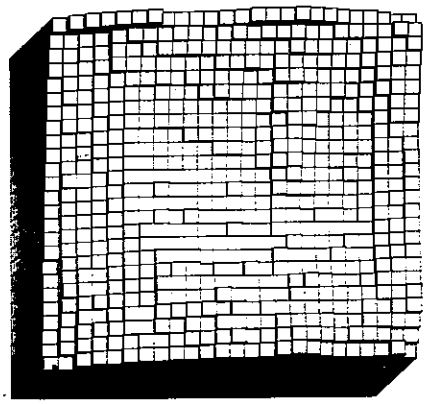


Fig. 6. Pseudo 3D representation of the homogeneity of the entire X-ray/cine/image digitization chain.

percentages area stenosis of 2.79% (s.d. 1.76%) resulted. For the logarithmic transfer function, a mean absolute error of 5.64% (s.d. 3.05%) resulted. To investigate the spatial dependency of this method the perspex model was also placed in the four corners of the field of view. The mean absolute difference in results between the measurements over the four corners, carried out at different kV's was 5.22% (s.d. 3.00%). The kiloVoltage dependency was also assessed. Preliminary results show that higher kiloVoltages lead to more accurate results.

6. DISCUSSION

In section 5 we have described a simple technique for the assessment of the transfer function of the cinefilm/image digitization chain and thus for the densitometric analysis technique. A more sophisticated densitometric analysis procedure should take into account additional spatially variant effects in the image intensifier and the one in the optical chain of the cine-digitizer, as well as other disturbing nonlinear effects. For example, the curvature of the input screen of the image intensifier results in a nonlinear geometric distortion (pincushion distortion) and in a nonuniform detection efficiency over the input intensifier screen (vignetting).
Other disturbing effects which have to be taken into account are beam hardening, X-ray scattering and veiling glare. The beam hardening results in different attenuation coefficients for different medium thicknesses. Scattered radiation disturbs the exponential absorption law and produces a spatially nonuniform opacification of the film resulting in loss of contrast. Veiling glare is an additional image distortion component from optical scattering in the image intensifier. A technique using a digital convolution algorithm has been proprosed by Shaw et al. to approximate and correct for the scatter and glare (7).

7. CONCLUSIONS

Preliminary results justify the following conclusions for the described simplified densitometric analysis technique:

1. Administration of 50% contrast gives better results than for 100% contrast.
2. The densitometric transfer function gives better results then the logarithmic transformation, especially for the larger cross-sectional areas.
3. Filming at higher kV's leads to more accurate results, which however is less clear at arteries filled with 50% contrast agent.
4. Spatial dependency was better than 5%, even without correction.

Most of these observations can be explained by the beam-hardening phenomenon. Further developments at present are directed towards more extensive validations of the densitometric procedures and the development of techniques to compute absolute cross-sectional area by either an internal or external calibration procedure.

ACKNOWLEDGEMENTS
The authors wish to thank Mrs. M.J. Kanters-Stam and Mrs. S.M. Spierdijk for their secretarial assistance in the preparation of this manuscript.

REFERENCES

1. Lipscomb K, Hooten S. Effect of stenotic dimensions and blood flow on the hemodynamic significance of model coronary arterial stenoses. Am. J. Cardiol. 42, 1978: 781-792.
2. Vlodaver Z, Edwards JE. Pathology of coronary atherosclerosis. Prog. Cardiovasc. Dis. 14, 1971: 256-274.
3. Reiber JHC, Serruys PW, Kooijman CJ,et al. Assessment of short-, medium-, and long-term variations in arterial dimensions from computer-assisted quantitation of coronary cineangiograms. Circulation 71, 1985: 280-288.
4. Reiber JHC, Serruys PW, Slager CJ. Quantitative coronary and left ventricular cineangiography: methodology and clinical applications. Martinus Nijhoff Publishers. Dordrecht/Boston/Lancaster, 1986.
5. Reiber JHC, Kooijman CJ, Slager CJ et al. Taking a quantitative approach to cineangiogram analysis. Diagnostic Imaging, April 1985: 87-89.
6. Reiber JHC, Slager CJ, Schuurbiers JCH, et al. Transfer functions of the X-ray-Cine-Video Chain applied to digital processing of coronary cineangiograms. In: Digital Imaging in Cardiovascular Radiology. P.H. Heintzen, R. Brennecke (Eds.). Georg Thieme Verlag Stuttgart, 1983: 89-104.
7. Shaw CG, Ergun DI, Van Lysel MS, et al. Quantitation techniques in digital subtraction videoangiography. In: Digital Radiography. W.R. Brody (Ed.). SPIE 314, 1982: 121-129.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

AUTOMATED DETECTION OF LEFT VENTRICULAR BOUNDARIES FROM 35mm CONTRAST CINE-ANGIOGRAMS.

P.J. v. Leeuwen, M. Sc., J.H.C. Reiber, Ph.D.

Laboratory for Clinical and Experimental Image Processing, Thoraxenter, Erasmus University, Rotterdam.

A new method is described for the automated detection of left ventricular (LV) contours in contrast cineangiograms in the RAO projection. The method requires the manual definition of three reference points, two at the aortic valve and one at the apex. Next, a model, derived from a learning set of manually drawn contours, is fit through these points and edge features are extracted along scanlines perpendicular to the local model boundary direction. The edge detection method is based on dynamic programming techniques, thus allowing the determination of local contour points to be influenced by the entire global border path. A preliminary qualitative evaluation study showed that in 85-90% of the ED-frames and in 70-80% of the ES-frames the LV boundary could be detected fully automatically. The time required to detect an ED- or ES-contour in a 512x512 image is 10 seconds. On the basis of these preliminary data it may be concluded that reliable automated detection of LV-boundaries in a routinely acceptable processing time is feasible.

1. INTRODUCTION

The nature and extent of left ventricular wall motion abnormalities can be assessed accurately by the analysis of the left ventricular roentgen contrast cineangiograms obtained during cardiac catheterization. Conventional assessment of global and regional left ventricular function from cineangiograms is performed either by visual inspection or by manual tracing of the end-diastolic (ED) and end-systolic (ES) cineframes. The technique of visual interpretation is hampered by considerable inter- and intraobserver variabilities (1-3).

The manual tracing method is tedious, time-consuming and also restricted by definite intra- and interobserver variabilities. (4-6). From the ED- and ES-boundaries, global and regional ejection fractions, being measures for LV-function can be computed. Analysis of instantaneous regional LV-wall motion requires the delineation of the boundaries in each frame over one or more cardiac cycles, which is not very practical if it needs to be performed entirely manually.

To increase the reproducibility of the assessment of the left ventricular function and to make frame-to-frame analysis feasible, we are in the process of developing a software-based technique for automated boundary detection of the left ventricular angiograms. Our experiences with a hard-wired left ventricular angioprocessing system, the Contouromat, have been very positive and have shown that reproducible automated contour detection of the left ventricular boundary is feasible (7,8).

The first phase of this project involves the analysis of only the ED- and ES-boundaries, while the frame-to-frame boundary detection technique will be implemented in the second phase. In this paper the basic principles of the first phase of this project will be described.

The LV-boundary detection method that we have implemented is based on dynamic programming techniques and has been described by Kooijman et al (9). This algorithm has been applied successfully in the detection of the contours of coronary arterial segments and requires very little operator interaction.

The detection procedure has been implemented at the Laboratory for Clinical and Experimental Image Processing of the Thoraxcenter in Rotterdam as part of the PDP 11/44-based interactive Cardiovascular Angiography Analysis System (CAAS), which runs under the RSX11/M+ Operating System. A block diagram of this research system is given in Fig. 1; operator interaction is possible with a keyboard and a writing tablet.

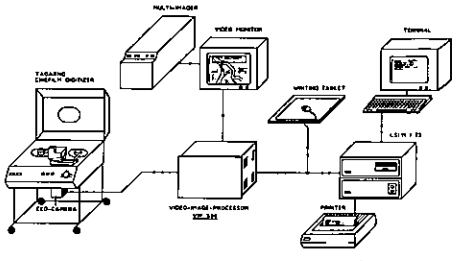


Fig. 1. Block diagram of the CAAS.

2. PROCEDURE

Single-plane left ventricular cineangiograms were acquired in the RAO (right anterior oblique)-view at a film speed of 25-50 frames/sec. To determine the left ventricular contour in a selected ED- or ES-frame of a 35 mm cinefilm, the image is converted into video format by means of the cine-video converter (CIVICO) of the CAAS, digitized and stored in the memory of the image processing system. Each digitized image consists of 512x512 pixels with 8 bits of grey levels (fig. 2).

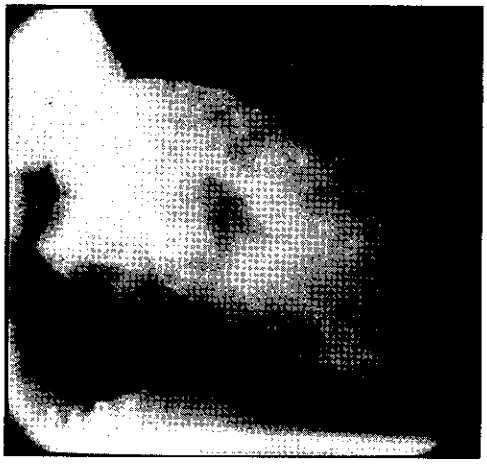


Fig. 2. Digitized image of the left ventricle in the ED-phase.

The method that has been developed to automatically detect the left ventricular border consists of three parts:

a. Extraction of image information along a given model into a resampled rectangular subimage.
b. Edge enhancement in the subimage to obtain a cost matrix.
c. Tracing a path of minimum global cost through the matrix using a dynamic search algorithm.

These three steps will be described in more detail in the following paragraphs.

Resampling procedure

For the automated left ventricular boundary detection edge information must be extracted from the original image in an efficient manner, i.e. unused information should be excluded as much as possible. For these purposes, a new subimage with the appropriate edge information must be created by resampling the original image. The subimage is a rectangular matrix, with every row being a resampled line (scanline) of the original image, preferably directed orthogonal to the expected local boundary direction. The most commonly used model to obtain such scanlines is a polar transformation defined with respect to an operator defined center point or to the densitometric center of the image. The greatest disadvantage of this model for LV-boundary detection purposes is the large deviation of the directions of the scanlines from orthogonality with the local LV-boundary direction at the apex and at the base, especially in ES- frames. To avoid such problems with the polar and other fixed model transformations, we propose a more flexible model. On the basis of a learn set of 10 ED- and ES-contours acquired from a set of 10 arbitrarily selected left ventricular cineangiograms, representative standard contours were created and stored on disk. The ED- and ES contours were manually drawn in the digitized images.

In a cineframe to be analyzed the operator indicates three LV-boundary positions, i.e. the two endpoints of the aortic valve and the apex. The model contour is fit through these three positions, resulting in a crude estimate of the LV-boundary in that particular frame. On the basis of this model reasonable approximations of the scanlines orthogonal to the local directions of the left ventricular contour to be determined can be obtained. The resulting scanlines then form the resampled subimage. A total of 128 scanlines are extracted along the LV circumference with a width of 64 pixels centered around the model contour point (fig. 3).

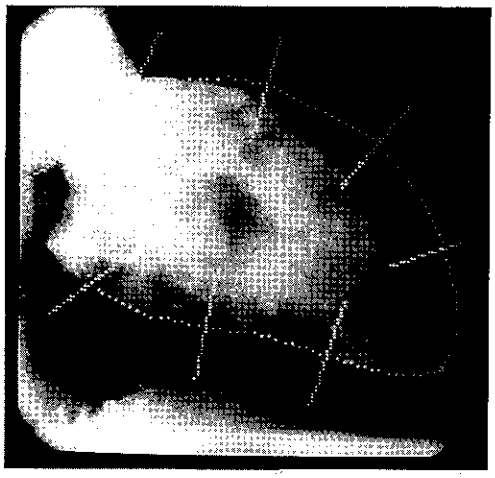


Fig. 3. Image of figure 2 showing the 3 reference points, the LV-model contour and 8 scanlines.

Edge Enhancement

Once a subimage has been extracted, a one dimensional gradient operator is applied to each scanline to enhance the edge-features, resulting in a socalled cost matrix. The gradient operator is application dependent; in our case, a weighted sum of modified first and second difference functions has been used. The gradient coefficients are selected on the basis of the densitometric profile of the edges being tracked. Fig 4 shows the brightness profile of one selected scanline plus the differ-

ence function.

Fig. 4. Example of the brightness profile of a scanline (A) and the derivative function (B).

Contour extraction

Given the cost matrix, an optimal global path through the matrix according to minimal cost criteria is determined with a dynamic search algorithm (9). In figure 5 the resampled image and the cost matrix both with and without the traced path superimposed are presented.

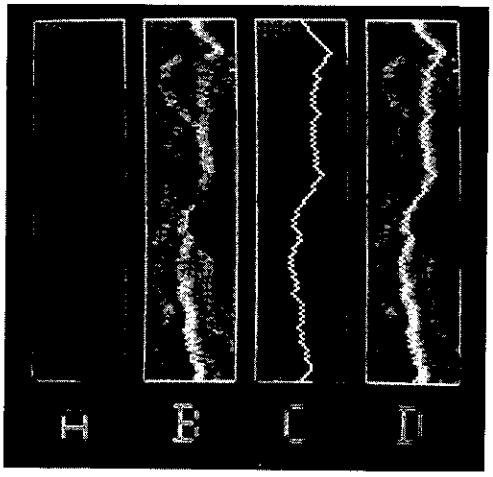


Fig. 5. This photograph shows the resampled image, and the cost matrix without (a,b) and with (c,d) the traced path superimposed.

As a next step, the minimal cost path is retransformed to the cartesian coordinate system. Finally, an interpolation and smoothing procedure is applied to the detected contour positions and the resulting contour is superimposed on the digitized left ventricular cineframe. The resulting contour for the image of figure 2 is shown in figure 6. This contour detection procedure is applied separately for the ED- and ES-cineframes. If necessary, the operator can make corrections to the portions of the contour that he does not accept as being the actual LV boundary.

Preliminary results of this version of the

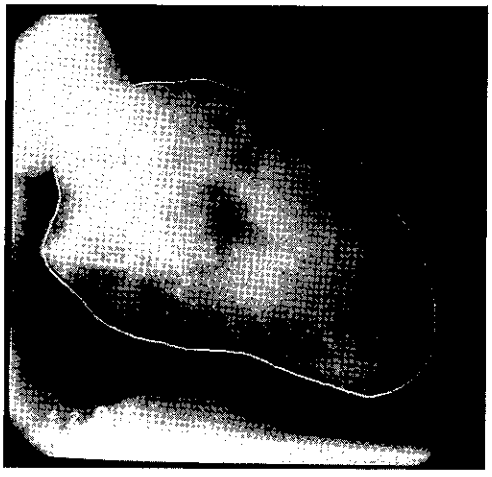


Fig. 6. Image of figure 2 with the resulting contour.

contour detection algorithm have been very promising. The use of the LV model as compared to a conventional polar transformation has greatly improved the success of the algorithm. However, in the basal contour parts close to the aortic valve detection errors could easily occur. These problems were caused by the fact that the model contour may deviate heavily from the actual LV boundary in these parts. A slight modification in the algorithm could solve these problems. In stead of detecting the complete contour in one step from the cost matrix, a step-wise approach has been taken. The resampling procedure now starts at the apex and proceeds on both the anterior and posterior sides of the ventricle along the model towards the aortic valve. After having resampled about two-thirds of the distance, the global optimal paths up to those anterior and posterior positions are tracked. Since we know that the model is represented by the middle vertical column of the resampled image, the deviation of the last position of the traced path with respect to the model, can be determined. If the traced path deviates more than a given interval from the model, the portion of the model between the current position and the aortic valve endpoint will be transformed and rotated to fit the expected boundary. Subsequently, the process of resampling and tracking is continued and finally the global optimal path is found. This modification improved the performance of the algorithm in the basal portions of the LV boundary greatly.

3. RESULTS

Presently the LV contour detection algorithm has been tested on a set of 20 ED- and 12-ES cineframes arbitrarily selected from a total

of 20 patient studies. The detected contours were examined visually. Of the 20 ED-contours, 17 could be regarded as totally correct, while 3 contours required a minor correction in the postero-basal region near the aortic valve.

Of the 12 ES-frames 7 contours were detected correctly. Of the remaining 5 contours, 3 required a minor correction near the mitral valve, one suffered an irregularity due to disturbing bone structures in the image and one LV contour was detected poorly due to the very low contrast in the apical and posterior parts of the contour.

Altogether, based on visual interpretation of the results in this limited evaluation study, it may be expected that with the present version of the software 85-90% of the ED-contours and 70-80% of the ES-contours will be detected correctly. In the remaining cases slight manual corrections may be necessary. More extensive quantitative evaluations are in progress.

The time required to detect an ED- or ES-boundary in a 512x512 image is 10 seconds; this is the true computation time assuming that the three reference points are available. It should be noted that about 40% of the time is required to send and receive image data between the image processor and the host computer.

4. CONCLUSIONS

In this paper a new algorithm for the automated detection of the left ventricular outline from 35 mm cineangiograms has been presented. On the basis of a preliminary evaluation, it could be shown that in 85%-90% of the ED-frames and in 70-80% of the ES frames, the LV contour could be detected automatically in a very acceptable processing time. To be applicable in routine daily practice, a success rate of 90% should be achieved. In addition, it should be preferable to define the reference points automatically. Therefore, further refinements will be applied.

Further developments are directed towards the frame-to-frame analysis. Since LV-cineangiograms are recorded at a rate varying from 25 to 50 frames/second, the displacement of the left ventricular boundary between two consecutive frames is usually so small that the corresponding positions in the next frame can be predicted reasonably well.

By using the information of the contour detected in the previous frame the algorithm is expected to prove very useful in assessing edges at dropout regions of the left ventricular cavity.

5. ACKNOWLEDGEMENTS

The authors wish to thank Mrs. M.S. Spierdijk and Mrs. M.J. Kanters-Stam for their secretarial assistance with the preparation of this manuscript.

References

1. Chaitman BR, DeMots H, Bristow JD, Rösch J, Rahimtoola SH.
Objective and subjective analysis of left ventricular angiograms.
Circulation 52, 1975: 420-425,.
2. Zir LM, Miller SW, Dinsomore RE, Gilbert JP, Harthorne JW.
Interobserver variability in coronary angiography.
Circulation 53, 1976: 627-632.
3. Rogers WJ, Smith LR, Hood WP, Mantle JA, Rackley CE, Russell RO.
Effect of filming projection and interobserver variability on angiographic biplane left ventricular volume determination.
Circulation 59, 1979: 96-104.
4. Sheehan FH, Stewart DK, Dodge HT, Mitten S, Bolson EL, Brown BG.
Variability in the measurement of regional left ventricular wall motion from contrast angiograms.
Circulation 68, 1983: 550-559.
5. Clayton PD, Klausner SC, Blair TJ, Jeppson GM, Liddle HV.
Sources and magnitude of variability in measurements of regional left ventricular function. In: Ventricular Wall Motion, U. Sigwart, P.H. Heintzen (Eds). Georg Thieme Verlag, Stuttgart/New York, 1984: 90-99.
6. Cohn PF, Levine JA; Bergeron GA, Gorlin R.
Reproducibility of the angiographic left ventricular ejection fraction in patients with coronary artery disease.
Am Heart J 88, 1974: 713-720.
7. Slager CJ, Reiber JHC, Schuurbiers JCH, Meester GT:
Contouromat - A hard-wired left ventricular angio processing system. I. Design and application.
Comp Biomed Res 11, 1978: 491-502.
8. Reiber JHC, Slager CJ, Schuurbiers JCH, Meester GT.
Contouromat - A hard-wired left ventricular angioprocessing system. II. Performance evaluation.
Comp Biomed Res 11, 1978: 503-523.
9. Kooijman CJ, Reiber JHC, Gerbrands JJ, Schuurbiers JCH, Slager CJ, Boer A den, Serruys PW.
Computer-aided quantitation of the severity of coronary obstructions from single view cineangiograms.
First IEEE Comp Soc Int Symp on Medical Imaging and Image Interpretation.
IEEE Cat 82, CH1804-4, 1982: 59-64.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

DEVELOPMENT OF A DIGITAL DIAGNOSTIC WORKSTATION FOR MEDICAL ULTRA-SOUND

J. Ridder*[1)], E. Hoyer*, A.J. Berkhout**

* Institute of Applied Physics TNO-TH
P.O. Box 155, 2600 AD Delft
** Laboratory of Seismics and Acoustics
Delft University of Technology
P.O. Box 5046, 2600 GA Delft

1. INTRODUCTION

During the last three years the Institute of Applied Physics, Delft University of Technology and the NV Optische Industrie Oude Delft (Oldelft), have been working together on the development of a digital diagnostic workstation for medical ultra-sound. At the start of the project the following research objectives were formulated:

- The system should be able to produce very accurate images of internal structures of the human body (e.q. liver, kidney, spline, etc.). The imaging principle is based on echo-acoustical techniques. The resolution of the system will be ca. 1-1.8 mm (-20 dB level).
- The system should allow the application of advanced signal analyses methods on the gathered data ("tissue characterization"). With these analyses methods structural and physical properties (features) of the medium could be determined. The actual values of the features can be used for the characterization of the medium (correlation with e.g. anatomical and pathological information of tissues).

Realization of both objectives will result in a dramatic improvement of the quality of the diagnostic capabilities of medical ultra-sound systems.

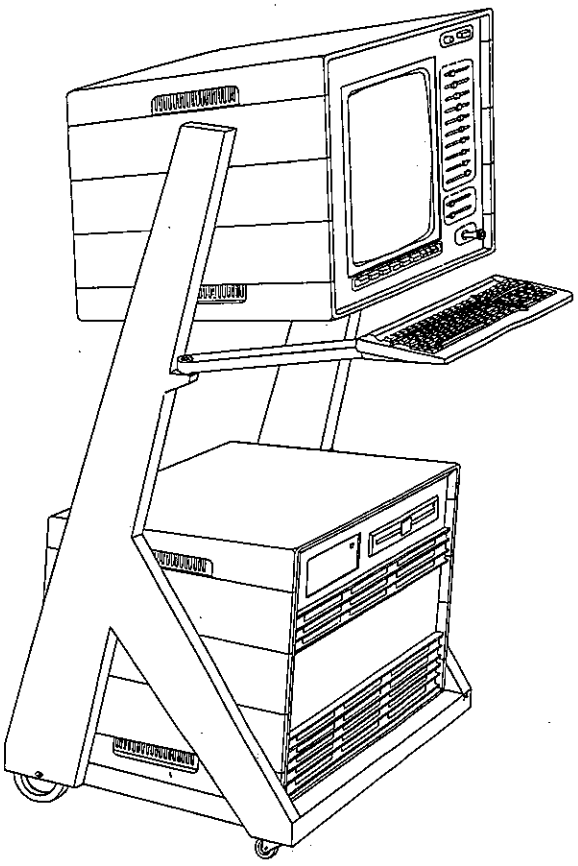


Fig. 1:
Workstation

2. INVERSION

Measured echo-acoustical signals are determined by:

- Data-acquisition method and data-acquisition system.
- Acoustical properties of the medium between the structures causing the echo's and the transducer (i.e. propagation effects).

1) Current address: Institute of Applied Geoscience, P.O. Box 285, 2600 AG Delft

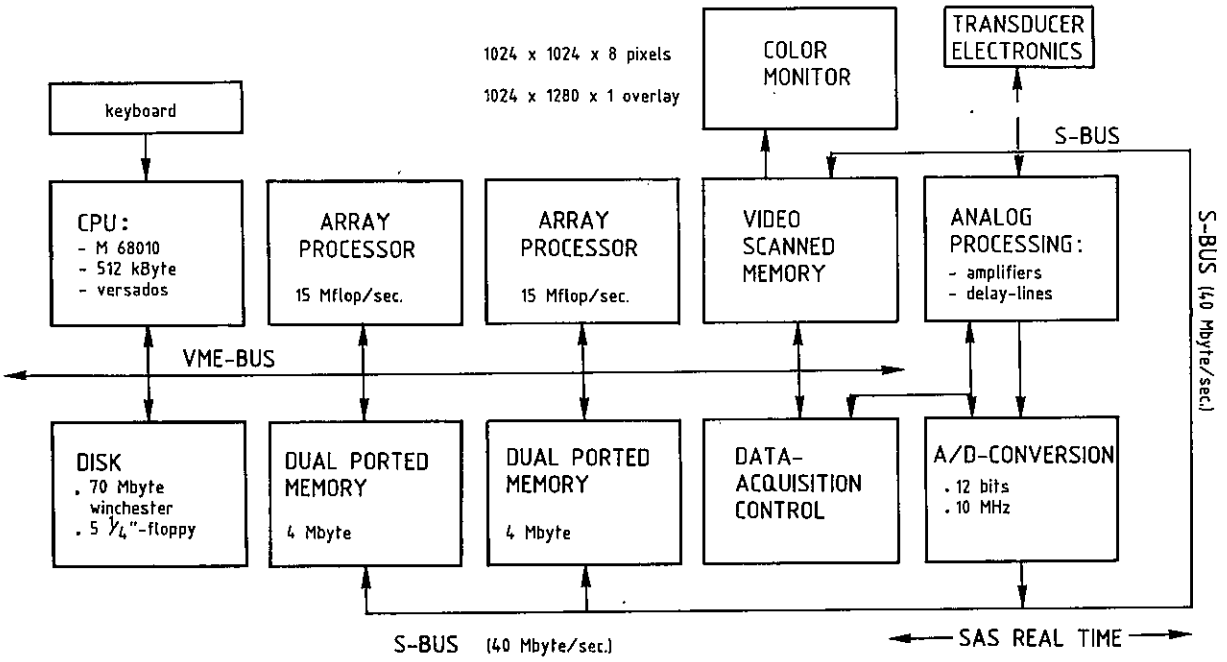


Fig. 2:
Functional blockdiagram of workstation

- Local acoustical properties of the structures causing the echo's. Examples of these local acoustical properties are: local impedance, local propagation velocity of sound, local density, etc.

So the following boundary conditions for successful acoustical imaging and analyses techniques can be formulated:
- The measured echo-signals should be corrected for distortion caused by data-acquisition method and data-acquisition system. An example of such a distortion is the angle dependent impulse response of transducers.
- Accurate correction (i.e. acoustical inversion) for the influence of propagation effects of the medium on both ultrasound pulse and echo's is a necessity.

In the developed workstation an accurate acoustic inversion technique has been implemented. This wave-equation based technique is called: Spatial de-phasing in the double Fourier domain (Berkhout, 1984). During the application of the acoustic inversion technique the data is also corrected for distortions caused by data-acquisition method and system.

3. FUNCTIONAL DESCRIPTION

Figure 1 shows the external of the workstation, while figure 2 consists of a blockdiagram of the system.

The workstation has two operational modes, i.e. the real-time mode and the inversion mode. The real-time mode will be used for the localization of structures. In the real-time mode the system is, with respect to signal processing, comparable with now commercial available linear array scanners. The acquired images are instantaneous displayed on the screen of the display subsystem. In real-time mode the system can produce 19 images per second. Each image consists of ca. 400 lines. After the user has localised a cross-section of particular interest he can switch the system to the inversion mode. In the inversion mode during 140 msec data will be gathered with almost omnidirectional beams at 512 different positions within the aperture area of the transducer. The amount of data acquired for one single image in the inversion mode is 4 Mbyte. This data is stored in a semi-conductor memory and processed according to the above mentioned inversion technique.

For the processing two (board level) array-processors (total performance of 30 Mflop) are available. After inversion the reconstructed image is displayed on the screen of the display subsystem. The total procedure, inclusive data-acquisition, inversion and display takes about 8 seconds. Both array-processors are also extensively used during the application of the algorithms for the determination of local acoustical parameters.

Berkhout, A.J. 1984, Seismic Migration-Imaging of Acoustic Energy by Wave Field Extrapolation, vol. 14A, Elsevier, Amsterdam and New York.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

Analysis of three-dimensional images of cell nuclei

B. Schmitt, G. Zinser, A. Erhardt, D. Komitowski, J. Bille*

German Cancer Research Center,
Institute of Experimental Pathology,
Im Neuenheimer Feld 280, D-6900 Heidelberg, FRG

* University of Heidelberg,
Institute of Applied Physics I,
Albert-Überle-Straße, D-6900 Heidelberg, FRG

Abstract

In histology image processing methods are a usefull tool to describe the nuclear structure. But in high resolution light microscopy a two-dimensional image of a cell nucleus forms an optical section. Because of the low focal depth it detects only a part of the nucleus. In addition it contains defocused projections of adjected focal planes. To overcome these problems a three-dimensional image is recorded as a focus series of two-dimensional images. The defocused projections are rejected by a reconstruction method <1>. By using the a priori information that the nucleus is more or less a sphere, the three-dimensional image is transformed in spherical coordinates. In this coordinate system it is posible to locate the boundary surface of the nucleus. The nuclear contour is transformed back to cartesian coordinates. It serves to segment the nucleus from its background. Within the volume of the nucleus local maximas of optical density, socalled chromatin particles, were detected which describes the internal structure of the nucleus.

Introduction

A threedimensional object, like a cell nucleus, exeedes the focal depth of a high resolution light microscopic two-dimensional image. Such images contain only a part of the information content in the whole nucleus. To describe the nuclear structur you must record a three-dimensional image by a series of focal planes. Each plane of such a three-dimensional image, however, is still distorted by defocused projections from the whole nucleus. This distortions can be removed by using a reconstruction method discribed earlier <1>. The result is a three-dimensional image with a good spatial resolution, especially perpendicular to the optical axis.
In digital image analysis, the segmentation of a region of interest is an important step. It causes, however, various technical problems. Because nuclear structures are so complex, exact segmentation meets with considerable difficulties related to the optical properties of these images. Especially one of the greatest problems is, that the variation of the optical density along the border of the nucleus is similar to the variation inside the nucleus. There is no posibility to segment the nucleus with a global threshold method. So a new method is created by using the a priori information that the nucleus is a sphere or an ellipsoid.

Image recording and reconstruction

Three dimensional images of Feulgen-stained cell nuclei are recorded as focus-series of 64 two-dimensional images using a microscope 'Axiomat' (Zeiss) with an objective 100X, numerical aperture 1.3 and an plumbicon scanner. The video signal is digitalized into 64 grey levels by an image analysis system 'Quantimet 720' (Cambridge Instruments, Inc). The sample spacing is .125 µm perpendicular and .25 µm parallel to the optical axis. The images are stored on magnetic tape by a 'pdp 11/34' computer for offline processing with a 'VAX 750' computer (both digital equipment corporation). The images are reconstructed by filtering with effective invers optical transfer function <1>.

Nucleus segmentation

In order to segment the nucleus from its background we first discussed the posibility to transform every x-y-plane in polar coordinates. By doing this we get a cylindercoordinate system.

$IM(x,y,z) \rightarrow C(r,\varphi,z)$

In this cobrdinate system we search the boundary of the nucleus with a tracking algorithm described in <2>. But there we have a big variation of the surface radius ($0 \leq r \leq r_{max}$) depending on the value of z. These property render the segmentation at the 'pole' zone ($r \rightarrow 0$) of the nucleus.
Therefore we developed a new segmentation method described as follows: The three-dimensional image IM (x,y,z) is transformed from cartesian coordinates (CCS) to spherical coordinates (SCS).

$IM(x,y,z) \rightarrow S(r,\varphi,\vartheta)$

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

NOISE CAUSED BY SAMPLING-TIME JITTER WITH APPLICATIONS TO SAMPLING-FREQUENCY CONVERSION

Gérard Verkroost

Technische Hogeschool Eindhoven, Afdeling der Elektrotechniek
Postbus 513
5600 MB Eindhoven, The Netherlands

1. INTRODUCTION

In the processing of digital signals it is sometimes necessary to change the sampling frequency of a signal from a given frequency $F_1 = 1/T_1$ to a different one $F_2 = 1/T_2$. If the ratio F_2/F_1 is rational, this can be done by interpolation and decimation [1]. If, however, two digital systems have to be interfaced and these systems have independent clocks, the ratio F_2/F_1 will be irrational. In that case the input signal $x_1(k/F_1)$ is interpolated to get an intermediate signal $x_i(k/F_i)$, where the intermediate frequency F_i is an integer multiple of F_1 ($F_i = LF_1$). The output signal $x_2(k/F_2)$ is then determined using this intermediate signal.

Now let t_2 be a time instant where an output sample is needed, according to the time grid belong to F_2. However, in general, t_2 will not fit in the intermediate time grid and a sample belonging to t_2 is not available. If t_i is the time instant in the intermediate time grid which is nearest to t_2 the available sample $x_i(t_i)$ can be used as an approximation for the output sample $x_2(t_2)$ [2]. This approximation introduces an error that will be investigated.

Let τ be the time difference between t_2 and t_i. With T_i denoting the sampling period of the intermediate signal this time error τ is bounded by

$$-\frac{T_i}{2} \le \tau < \frac{T_i}{2} . \qquad (1)$$

For each output sample the time error τ will be different. So τ fluctuates; this phenomenon is called "time jitter". Because of the irrational ratio F_2/F_i this time error will have a uniform probability density distribution over the interval given in (1).

The time jitter described above introduces an error in the output signal $x_2(n/F_2)$. This error can be made arbitrarily small by choosing a sufficiently small T_i. A smaller T_i, however, involves more interpolation computations and thus makes a realization more costly. So the question arises: what is the maximal T_i to fulfil certain requirements regarding the quality of the output signal? This question is investigated in this paper.

The jitter phenomenon is not only encountered in the problem sketched above; it is also encountered in other parts of the signal prcoessing proces. For example, the non-constant delay in an A/D convertor causes jitter. Jitter is also introduced by the instability of the sampling clock. With minor modifications the theory, given here, also applies to these and similar situations where jitter is encountered.

2. ANALYSIS OF THE ERROR

To analyze the general effect of sampling-time jitter the sampling of a continuous signal x(t) is considered. The signal is described by its statistical properties and is supposed to be real, stationary and bandlimited with maximum frequency B. Its spectrum is S(f), where f is the frequency.

This signal x(t) may be a physical signal. However when, as described in the introduction, a change of sampling frequency has to be accomplished, x(t) is a fictitious signal, viz. the ideal reconstruction of the samples $x_i(kT_i)$. So if x(t) would be available, resampling it with the frequency F_2 would deliver output-samples with a correct value. Now let t_2 be a time instant where an output sample is needed. Instead of the unavailable $x(t_2)$ the nearest grid value $x(t_2+\tau)$ is attached to the sample. So an error

$$n = x(t_2+\tau) - x(t_2) \qquad (2)$$

is introduced in the output signal (See Fig. 1). Any time an output sample is taken in this way a different error n is added to the signal.

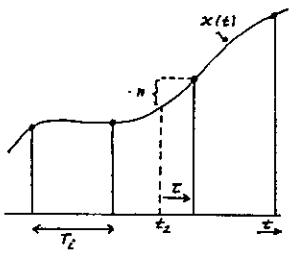


Fig. 1. The error signal n.

This disturbing signal n is called noise and its power P_n follows from (2) as

$$P_n(\tau) = E(n^2) = E\{x^2(t_2+\tau) + x^2(t_2) - 2x(t_2+\tau).x(t_2)\}$$
$$= 2\{R(0) - R(\tau)\}, \qquad (3)$$

where $R(\tau)$ is the autocorrelation function x(t), with

$$R(\tau) = E\{x(t_2+\tau)\ x(t_2)\} \qquad (4)$$

and $R(0) = P$ is the signal power of x(t).

In (3) it is stated in fact how P_n depends upon the possible change of x(t) during the time interval τ.

The autocorrelation function R(t) and the power spectrum S(f) are related according to

$$R(\tau) = \int_{-\infty}^{\infty} S(f)\, e^{2\pi j f\tau} df = 2 \int_{o}^{\infty} S(f)\cos(2\pi f\tau) df. \quad (5)$$

Substituting (5) in (3) leads to

$$P_n(\tau) = 4 \int_{o}^{\infty} S(f)\, \{1-\cos(2\pi f\tau)\} df. \quad (6)$$

As discussed in the introduction, τ is fluctuating between $-T_i/2$ and $T_i/2$ and its probability density function will be a constant $1/T_i$ in that interval. So not the noise power $P_n(\tau)$, given in (6), is of interest but its mean value $\bar{P}_n$ according to

$$\bar{P}_n = E\{P_n(\tau)\} = \int_{-T_i/2}^{T_i/2} \frac{1}{T_i} P_n(\tau) d\tau. \quad (7)$$

Substitution of (6) in (7) and integration over τ leads to

$$\bar{P}_n = 4 \int_{o}^{\infty} \left(1 - \frac{\sin(\pi f T_i)}{\pi f T_i}\right) S(f) df. \quad (8)$$

So given the spectrum S(f) of the input signal and given the intermediate sampling frequency $F_i = 1/T_i$, expression (8) determines the power of the noise added in the proces of "resampling".

3. A PRACTICAL APPROXIMATION

The spectrum S(f) is bandlimited with a maximum frequency B and $B < F_1/2$. For the intermediate signal, in most practical cases, $F_i >> F_1$ and therefore $B << F_i$ holds. In that case (8) can be approximated by a simpler equation using the fact that the integration extends from f = 0 to f = B and that in that interval $fT_i = f/F_i << 1$ holds. So expanding

$1 - \frac{\sin(\pi f T_i)}{\pi f T_i}$ in a Taylor series and only using

in a Taylor series and only using the first term (8) is approximated by

$$\bar{P}_n \simeq \frac{2\pi^2 T_i^2}{3} \int_{o}^{B} f^2 S(f) df. \quad (9)$$

Thus the noise power is, in practical cases, easily calculated using (9).

In the derivation of (9) the approximation

$1 - \frac{\sin x}{x} \simeq \frac{x^2}{6}$ is used, with $x = \pi f T_i$. But because

$1 - \frac{\sin x}{x} \leq \frac{x^2}{6}$ also holds and the spectrum S(f) is positive, the approximation (9) is also an upper-bound for $\bar{P}_n$.

REFERENCES

[1] Crochiere, R.E. and Rabiner, L.R., "Interpolation and Decimation of Digital Signals - A Tutorial Review", Proc. IEEE, vol. 69, no. 3, March 1981, pp. 300-331.

[2] Lagadec, R. and Kunz, H.O., "A Universal, Digital Sampling Frequency Converter For Digital Audio", IEEE, CH 1610-5/81.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

CONSTRAINED SIGNAL RECONSTRUCTION - A UNIFIED APPROACH

K. Stewart, T.S. Durrani

Signal Processing Group, Dept. Electronic and Electrical Eng.
University of Strathclyde, Glasgow,
Scotland, U.K.

Many current procedures aimed at the reconstruction of linearly degraded signals or images in noise may be expressed in terms of a general one-parameter family of functionals whose roots lie in regularisation and approximation theory. This paper illustrates the structure and limitations of such procedures by using a geometric (Hilbert space) model of the restoration problem, and discusses the principles of an analytical technique aimed at overcoming these limitations through the use of Generalised Cross-Validation and the singular value decomposition.

1.0 REGULARISATION IN RECONSTRUCTION

Linear reconstruction problems in signal and image processing may be posed in terms of the equation

$$g = Af + \varepsilon \qquad (1)$$

where g is an N-length vector of data points, f is the true solution to be recovered, A is the operator describing the degradation process to be inverted, and ε is a zero-mean white-noise process with auto-correlation matrix R. The purpose of a reconstruction algorithm is to design a linear or non-linear inverse operator B to recover an estimate $\hat{f}$ of f from only a knowledge of g, A and possibly some limited knowledge about the noise process ε. In (1) f is taken to be a member of F - an infinite dimensional (continuous signals) or M-dimensional (discrete signals, length M) Hilbert space over the real numbers, whilst g is an element of G - a corresponding N-dimensional space. The respective norms in F and G are the customary L_2 or ℓ_2 norms as appropriate.

A useful - and commonly used - method of solving (1) which originated in regularisation theory [1] is to estimate f by

$$\hat{f} = \underset{y \in F}{\text{minimiser}}\ \rho(g-Ay) + \alpha\Upsilon(y-f_o) \qquad (2)$$

where ρ and Υ are convex functionals over G and F (thereby assuring a unique solution estimate), α is the regularisation parameter, and f_o - often taken to be the zero vector - represents some a-priori estimate of the true solution f. By appropriate choice of F, G, A, ρ and Υ the general form of (2) can represent such diverse continuous and discrete signal restoration algorithms as minimum-norm bandlimited signal extrapolation [2], maximum entropy image restoration [3], and 'Marquardt' deconvolution schemes [4], along with many others.

This paper is concerned with a geometric analysis of the general characteristics of the algorithms described by (2). Also, in the interests of conciseness and simplicity, most attention is given to the discrete deconvolution case ($F \triangleq R^M, G \triangleq R^N, A:F \to G$, $M<N$) in the presence of Gaussian noise. Most of the discussion which follows is, however, extendible to continuous signal reconstruction problems.

2.0 HILBERT SPACE MODELS

A useful view of the discrete deconvolution problem is obtained by regarding the degradation operator $A:F \to G$ as mapping points $f \in F$ into true data points $(Af) \in G$ which are subsequently shifted by the noise vector ε to the available data point g (Figure 1).

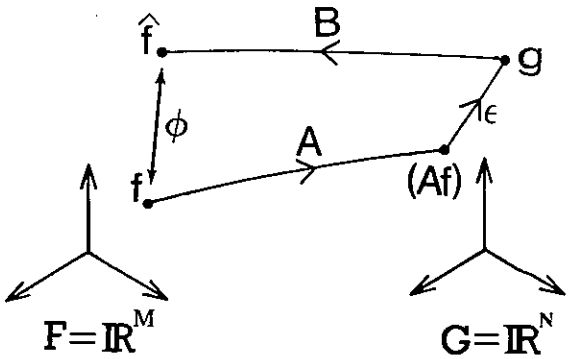


Figure 1

As stated, the objective is to design an operator $B:G \to F$ to map the available data point g to a solution estimate f such that some functional ϕ measuring the 'goodness of

solution' say $\phi = ||f-\hat{f}||$ - is optimised. The presence of noise on g leads to the familiar concept of a convex set D of possible data vectors in G defined by $D = \{x \in G : \rho(g-x) \leq \delta\}$ - where δ is a constant selected so that D contains the true data vector (Af) to a 95% or 99% confidence level. This leads to a set of possible solutions in F defined by the hyper-elliptical convex set $S = \{y \in F : \rho(g-Ay) \leq \delta\}$ (Figure 2).

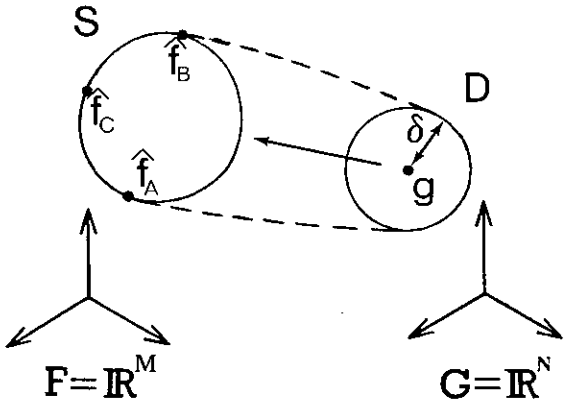


Figure 2

If A is badly conditioned (or even non-singular) S may be large, and in the absence of any other a-priori knowledge, any element of S may be taken as the solution estimate $\hat{f}$. The conventional view of the role of α and the γ functional is thus that of a constrained optimisation problem where α becomes a Lagrange multiplier and γ is optimised so as to enhance some property of the solution estimate whilst remaining sufficiently data consistent (ie. within the set S). Typical choices for γ have included weighted and unweighted square error norms, and negative entropy measures thereby allowing minimum and maximum energy as well as maximum-entropy solutions to be selected from S (solutions $\hat{f}_A$, $\hat{f}_B$, and $\hat{f}_C$ respectively in Figure 2). Alternatively (2) could be viewed as selecting the most data consistent solution (ie. minimise ρ) such that some signal property (measured by ϕ) is maintained at a set a-priori level.

The major problem, of course, is the choice of α. Even with strong a-priori knowledge on the noise process - such as knowing the noise variance σ^2 - it is difficult to choose the confidence level (reflected by δ) such that the solution estimate minimises ϕ. Equally, it would be very unusual to have such strong a-priori knowledge available as, for example, $\gamma(\hat{f}-f_o)$ = (a constant) to assist in choosing α.

These problems of selecting α, and as shall be seen, of selecting ρ, γ and f_o can be readily explained in terms of a model which describes the action of (2) in terms of a 'trajectory' of solutions in the space F.

3.0 SOLUTION TRAJECTORIES

A alternative interpretation of the role of ρ, γ, f_o, and α in (2) is obtained by considering the trajectory of solutions $\hat{f}(\alpha)$ in F obtained by varying the parameter α for preset ρ, γ and f_o. If $\alpha=0$ then $\hat{f}$ is $\hat{f}'$, the minimiser of $\rho(g-A\hat{f})$ which - if ρ is the customary ℓ_2 error measure - gives rise to a simple least squares solution. If $\alpha=\infty$ then $\hat{f}$ equals the a-priori signal estimate f_o. The solutions $\hat{f}(\alpha)$ generated using (2) for α in the range $0 < \alpha < \infty$ describe a continuous trajectory in F between $\hat{f}'$ and f_o. This solution space trajectory has its data space counterpart in G as a trajectory passing between g and (Af_o).

Figure 3 illustrates a general solution trajectory $\hat{f}$ which always starts at $\hat{f}'$ ($\alpha = 0$) and terminates at f_o ($\alpha=\infty$).

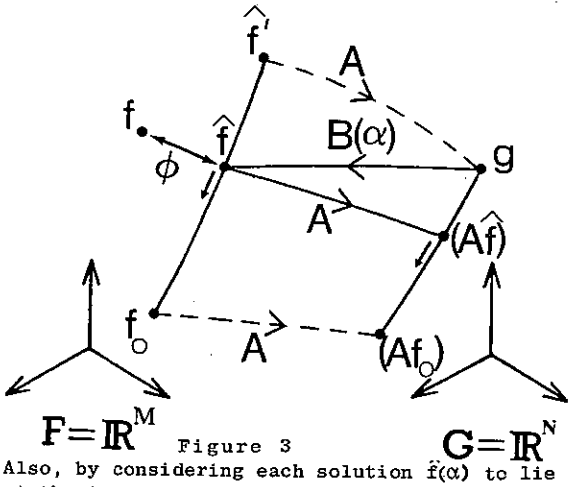


Figure 3

Also, by considering each solution $\hat{f}(\alpha)$ to lie at the tangential intersection of contour lines of constant $\rho(g-A\hat{f})$ and $\gamma(\hat{f}-f_o)$ (where α controls which contour lines determine the intersection) it is observed that the it is the shape of these contours in F (and hence the construction of ρ and γ) which determine the solution trajectory. Having established the solution trajectory by the choice of ρ and γ the process of choosing α becomes that of finding the trajectory location which optimises ϕ and this has led to development of methods for finding that location.

4.0 OPTIMAL SELECTION OF α

Early restoration algorithms based upon (2) chose α by solving (2) for several different values of α and selecting the one which gave the 'best' reconstruction. Unfortunately, this approach assumes that there is some extra a-priori information available on the true solution f which will allow this judgement to be made, or in other words, knowledge of the true solution is necessary to find the best

reconstruction.

A variety of alternative techniques aimed at selecting the best value of α have, however been proposed. One approach was originally known as the Discrepancy method [1] which argued that if $||(Af)-g|| \leq \tau$ - ie. that the data noise level was limited - then α should be such that $||(A\hat{f})-g|| = \tau$, giving rise to solution $\hat{f}_D$ of Figure 4 where the solution trajectory intersects the locus Q of points in G described by $Q=\{x \in G : ||x-g|| = \tau\}$. Similar approaches to this have been used in maximum-entropy image processing for example [4].

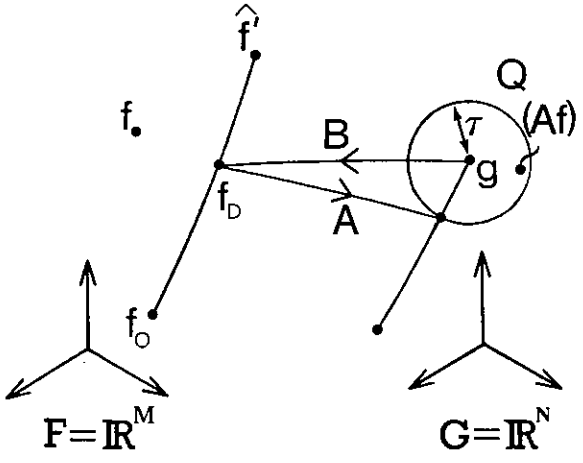


Figure 4

The major disadvantage with this procedure is that a-priori knowledge of the noise variance is required. Also, it is possible to imagine solution trajectories whose intersection with Q need not lie near to (Af) thus giving rise to a solution estimate which may be far from f.

Any optimal method for determining α centres on the 'goodness of solution' measure ϕ. A good choice of ϕ might be, for example, to select α so as to minimise the functional $\phi(\alpha)=||f-\hat{f}(\alpha)||$ or equivalently, if A is a positive-definite operator, to minimise $\phi(\alpha)=||(Af)-(A\hat{f}(\alpha))||$. An alternative strategy is to optimise the expectation of these quantities over the data eg. $\phi(\alpha)=E_g||f-\hat{f}(\alpha)||$. This method is essentially the same as the 'trial and error' approach described above however, and such estimators of α are not computable since f is unknown.

In the absence of such optimal means of selecting the best value of α, several algorithms aimed at providing useful estimations of that value have been developed including strategies based upon maximum-likelihood principles and the Generalised Cross Validation (GCV) approach [5,6]

GCV seeks to select α by using the idea that if α is a good estimate of the optimal value in the sense described by $\phi(\alpha)=||(Af)-(A\hat{f})||$ then $(A\hat{f}_{\lambda}(\alpha))(i)$ - the i-th point of the solution estimate obtained using the data vector g minus the i-th data point - will be close, on average, to g(i).. Thus α is chosen to minimise the function $V(\alpha)$ where:

$$V(\alpha)=\sum_{i=1}^{M}\left[(A\hat{f}_{\lambda}(\alpha))(i)-g(i)\right]^2$$

It can be shown [5] that, under certain conditions, the minimiser of $E_g[V(\alpha)]$ will approximate the minimiser of $E_g[T(\alpha)]=E_g||A\hat{f}(\alpha)-(Af)||^2$. GCV therefore evaluates the best value of by using the reconstructed solution obtained by deleting each data point in turn, re-predicting those data points, and then assigning a measure of that process to each value of α. This process of data-deletion could be extended to multiple data point deletion schemes, but only the simplest case of single point deletions shall be considered here.

Expressions for the cross-validation function $V(\alpha)$ can be obtained in terms of the singular value decomposition (SVD) of the operator A [5] particularly where the form of ρ and γ in (2) are the ℓ_2 norms in F and G and f_o is set to the zero vector. For the case of generalised forms of ρ and γ however, an expression for V does exist [7] as

$$V(B) = ||(I-AB)g||^2/[\text{tr}\,(I-AB)]^2 \qquad (4)$$

This then provides a general means of calculating V for each inverse operator B.

The importance of constraining the set of possible solutions to lie along a trajectory in F becomes apparent now, however, since the minimiser of $V(\alpha)$ or even $T(\alpha)$ along that trajectory may not be close to f, and it is apparent that if ρ and γ are badly selected, even finding the optimum location on the solution trajectory will not necessarily provide a good solution estimate. The real problem therefore is not simply to find an optimal α but also to design an optimal trajectory in F simultaneously.

5.0 MULTI-PARAMETER CROSS-VALIDATION

The objective of the restoration process can therefore be posed as designing B with only the minimisation of ϕ as a goal, rather than say, selecting an arbitrary solution from within the allowable solution set S.

In designing a scheme to do this, it is useful to examine the role of the functional γ in terms of the eigensystem of B. Assume that $\rho(g-Af) = ||g-A\hat{f}||^2$ and let $\gamma(f-f_o) = ||Cf||^2$ where C is a positive-definite matrix. Then (2) provides a solution estimate as:

$$\hat{f}(\alpha)= (A^TA+\alpha C^TC)^{-1}A^Tg \qquad (5)$$

If C is the identity matrix I - as is commonly the case - then it is apparent that the purpose of the inversion operator B is to modify the eigenvalues of A^TA by scaling them additively by α so as to achieve a more stable solution estimate.

This approach leads naturally onto the use of the singular value decomposition (SVD) in the design of B. However, rather than using the SVD in order simply to eliminate the smallest eigenvalues from A^TA (truncation), it is more flexible to filter them. Furthermore, the filtering procedure may be optimised with respect to ϕ which can be taken to the generalised cross-validation function V.

If A is full rank (although it may be near to singular), it is expressible as:

$$A = U\Lambda V^T \qquad (6)$$

where U and V are NxN and MxM unitary matrices respectively and Λ is an NxM diagonal matrix composed of the ordered positive square roots of the eigenvalues of A^TA - $\lambda_i : \lambda_1 > \lambda_2 > \ldots > \lambda_M$:

$$\Lambda^T = \text{diag}\,(\lambda_1, \lambda_2 \ldots, \lambda_M \mid 0)$$

A filtered generalised inverse B of A is thus:

$$B = VF\Lambda^{-1}U^T \qquad (7)$$

where $\Lambda^{-1} = \text{diag}\,(1/\lambda_1, 1/\lambda_2 \ldots, 1/\lambda_M \mid 0)$

with $F = \text{diag}\,(t_1, t_2, \ldots, t_M)$

and where $t_1, t_2, \ldots, t_M$ are the filter-coefficients. If for example, $t_i = 1$, i=1...M then the unstabilised ($\alpha = 0$) solution is obtained. Also if $t_i = \lambda_i^2 / (\lambda_i^2 + \alpha)$ a minimum-energy solution can be selected whilst by using a filter such that $t_i = 1$. i=1,...K and $t_i = 0$, i=K+1,...M the traditional SVD truncation scheme is arrived at.

By optimising the filter weights t_i with respect to V(B), therefore, rather than simply optimising $V(\alpha)$ with respect to α in relation to a previously selected 1-parameter family of solutions it should be possible to obtain significant improvements in the solution estimate.

O' Brien and Holt [7] have proposed a class of estimators aimed at the optimisation of V over the set of filter coefficients in (7) which - by abandoning the conventional approach of designing ρ and γ a-priori and then selecting α by examining the resultant reconstructions - should yield improved solution estimates without requiring knowledge of the noise variance or the nature of the original signal f.

Results will be presented at the conference describing the practical performance of this deconvolution scheme.

6.0 CONCLUSIONS

The commonality of several modern signal reconstruction procedures have been discussed important features of the factors which control their performance reviewed, and an alternative reconstruction algorithm - which employs a modified SVD approach governed by a GCV function - specifically dealing with the discrete deconvolution problem has been described.

7.0 ACKNOWLEDGEMENTS

The authors would like to thank Procurement Executive, Ministry of Defence, UK for their support in this work.

8.0 REFERENCES

[1] Groetsch C.W., The Theory of Tikhonov Regularisation for Fredholm Equations of the First Kind (Pitman, London, 1984)

[2] Abbiss J.B., De Mol C., and Dhadwal H.S., Optica Acta 30 (1982), 1, 107-124

[3] Wernecke S. J., D' Addario L. R., IEEE Trans. Comp. 26 (1977), 4, 351-364

[4] Treitel S.,Lines L. R., Geophysics, 50 (1985), 1, 99-109

[5] Golub G. H., Heath M., Wahba G., Technometrics, 21 (1979), 2, 215-223

[6] Wahba G., SIAM J. Numer. Anal., 14 (1977), 4, 651-666

[7] O' Brien D. M., Holt J. N., J. Austral. Math. Soc. (Series B), 22 (1981), 501-514.

SIGNAL PROCESSING III: Theories and Applications
I.T. Young et al. (editors)
Elsevier Science Publishers B.V. (North-Holland)

FIR-MEDIAN HYBRID FILTERS FOR IMAGE PROCESSING

Eija Heinänen, Ari Nieminen, Pekka Heinonen and Yrjö Neuvo

Tampere University of Technology, Department of Electrical Engineering
P.O. Box 527, SF-33101 Tampere, Finland

ABSTRACT

A new class of median type filters for image processing is analyzed. In the filters linear FIR substructures are used in conjunction with the median operation.The concept of multilevel median operation is introduced to improve the detail preserving property of the conventional median and the FMH filters. In the multilevel filters there exists a tradeoff between noise attenuation and detail preservation. The root signals and noise attenuation properties of the basic FIR-median hybrid (FMH) and multilevel FMH filters are compared with representative edge preserving filtering operations. Tests performed with real images indicate that FMH filters preserve the small details better than the conventional median filters and furthermore they are computationally much more efficient.

I. INTRODUCTION

A fundamental problem in image restoration is to remove the additive noise produced by the imaging system without blurring the fine details of the image. This problem arises e.g. in machine vision, remote sensing, computer tomography, and in other X-ray imaging systems, where the radiation dose should be minimized [1]. Characteristic of these images is that in addition to sharp edges they contain also narrow lines of different orientations. These lines bear often important information.

Two dimensional median filters (MF) have been used with success in many image processing applications. In the two dimensional case the median is taken over all the $(2k+1) \times (2k+1)$ values inside the sliding data window or over the values under a cross shaped mask whose orientation may by vertical-horizontal, or diagonal. The MF are able to smooth noisy images while retaining the edge structures almost intact [2]. However, the MF tend to distort details which are smaller than half of the window. By the details of an image we mean pixels inside particular areas which have high mutual correlation inside the area. So e.g. narrow lines are classified as details.

In this paper we analyze a new class of nonlinear filters called FIR-median hybrid filters (FMH filters) for image processing. These filters reduce the number of data sorting operations to a small constant irrespective of the window size and require, in addition, only simple averaging operations [3, 4, 5].

A new basic concept introduced in this paper is the multilevel median operation that makes it possible to build up both FMH and median filters that retain details of the image irrespective of their orientation. By using the concepts to be developed in this paper details which are significantly smaller than the size of the window can be preserved.

II. FIR-MEDIAN HYBRID FILTERS

In standard MF the median is taken over all samples inside the window. In principle the implementation of a MF requires a simple nonlinear operation. However, when the number of samples is large the ordering procedure becomes cumbersome.

In FMH filters the final output is the median calculated over the outputs of FIR subfilters and the central input sample. Two basic types of FMH filters can be formed. The unidirectional FMH filter includes two subfilters and the bidirectional FMH filter four subfilters. The subfilters are symmetrically located with respect to the central input sample. In the unidirectional FMH filters the subfilters and the central input sample are along the same line and in the bidirectional FMH filter the subfilters form a cross.

The algorithm of **the unidirectional FMH filter** is as follows:

$$y(m,n) = MED[y_1(m,n), y_2(m,n), x(m,n)], \qquad (1)$$

where the signals $y_i(l)$ are the outputs of FIR linear phase subfilters of size l and $x(m,n)$ is the central input sample.

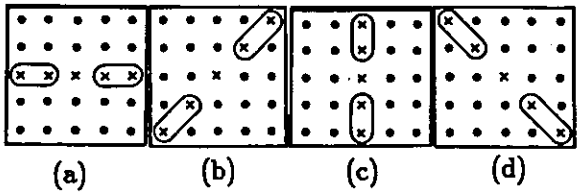


Fig. 1. Mask of the 1LH– (a) and its rotated versions (b), (c), and (d).

We can easily construct rotated one-level unidirectional FMH filters by rotating the basic mask. Rotation changes the root signals.

FMH filters analyzed in this paper have averaging substructures. If e.g. the window size is 5×5 ($k = 2$) it is possible to construct several different unidirectional FMH filters. We will analyze the unidirectional FIR-median hybrid filter 1LH– of Fig. 1 (a). Its rotated versions are shown in Fig. 1 (b), (c), and (d). The number refers to the number of levels in the median tree to be introduced below. The "–"

-sign in the name stands for unidirectional operation. Note that the unidirectional filters do not require true multipliers if the FIR subfilters are of averaging type and the number of elements is an integer power of two.

The bidirectional FMH filter has the following structure:

$$y(m,n) = MED[y_1(m,n), y_2(m,n), y_3(m,n), y_4(m,n), x(m,n)]. \quad (2)$$

Also in the bidirectional case we can construct rotated filter masks.

In this paper we analyze the bidirectional FMH filter 1LH+ of Fig. 2 (a) with averaging subfilters. Its rotated version R1LH+ is shown in Fig. 2 (b). The "+" -sign in the name stands for bidirectional operation.

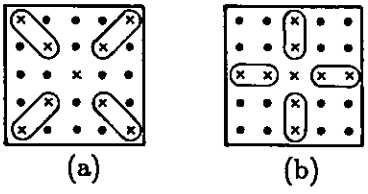


Fig. 2. Mask of the 1LH+ (a) and its rotated version (b).

III. MULTILEVEL FIR-MEDIAN HYBRID AND MEDIAN FILTERS

The FMH filters developed in previous chapter are all direction sensitive. For example the 1LH+ preserves vertical and horizontal lines but not diagonal lines. The R1LH+ preserves diagonal lines but not vertical and horizontal lines. If we increase the number of subfilters by taking e.g. the subfilters of the 1LH+ and the R1LH+ and the median is taken over the eight FIR outputs and the central pixel, the filter is no longer able to preserve subtle details.

To solve this problem we introduce the p-level median operation, the block diagram of which is shown in Fig. 3 (a). The algorithm is composed of tree structured median operators M. The basic filter blocks of the tree structure can be made of unidirectional (Fig. 3 (b)) or bidirectional (Fig. 3 (c)) FMH filters as well as of conventional median filters (Fig. 3 (d)). The notation y_i refers to the outputs of FIR subfilters and x_i to pixels, over which the median is calculated.

In the p-level algorithm there are in total 2^{p-1} basic filter blocks. Each filter block contains two FIR subfilters in the unidirectional case and four in the bidirectional case. The outputs of the filter blocks together with the central input sample are taken to a three-point median block M. The procedure of collecting two outputs of the earlier level and the central input sample to the three-point median operator is repeated until the final output of the algorithm is obtained.

The idea behind the multilevel median operation is to select the output value in such a way that it matches well with the local neighborhood spanned by the basic subfilters. It is easy to see that the structure of Fig. 3 has this desirable property. Consider e.g. a narrow line with different orientations on a noise free background as the input signal to the multilevel structure. By adding subfilters with different orientations to the structure, the filter becomes less sensitive

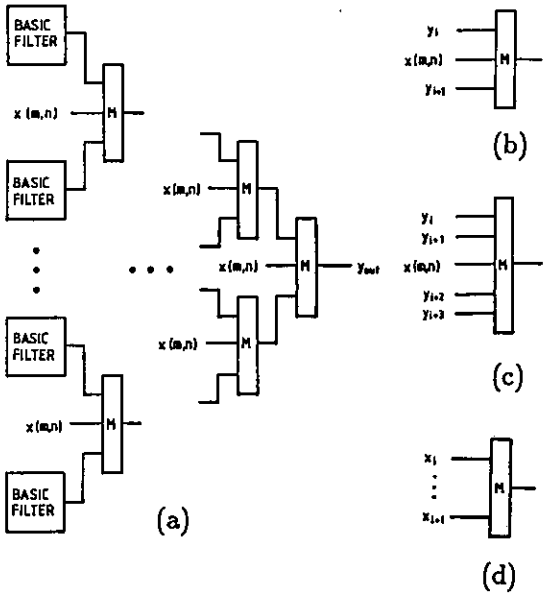


Fig. 3. (a) Block diagram of the multilevel median algorithm. Basic filter block for (b) the unidirectional FMH, (c) the bidirectional FMH filter, and (d) the MF.

to the orientations of fine details. It is easy to see that the basic subfilters can be of FMH or median type.

As an example we consider some multilevel FMH filters. The simplest bidirectional multilevel filter is **the two-level bidirectional FMH filter** 2LH+. It combines the subfilters of the 1LH+ and the R1LH+. **The three-level unidirectional FMH filter** 3LH– is obtained by using filters shown in Fig. 1 as basic filters. The 3LH– filter has eight FIR subfilters like the 2LH+ but it retains finer details than the two-level filters.

Now we show some examples of the multilevel median filters. The shape of the basic MF block can be a line or a cross. The two-level unidirectional MF is obtained by combining blocks shown in Fig. 4 (a) and 4 (b) and the central input sample with the median operation. The two-level 3×3 bidirectional MF (2LM+) we get in the same way using the masks shown in Fig. 4 (c) and 4 (d).

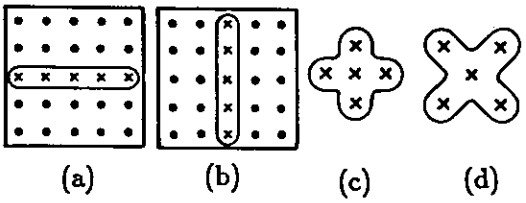


Fig. 4. Examples of MF masks in 5×5 and 3×3 windows.

IV. RESPONSE TO TEST IMAGES

To examine the filters' ability to preserve lines we have used test images, which consist of rings with different widths. They are generated using the formula

$$f(m,n) = \begin{cases} 800, & \text{if } 37 - w \le \sqrt{m^2+n^2} < 37; \\ 200, & \text{otherwise,} \end{cases} \quad (3)$$

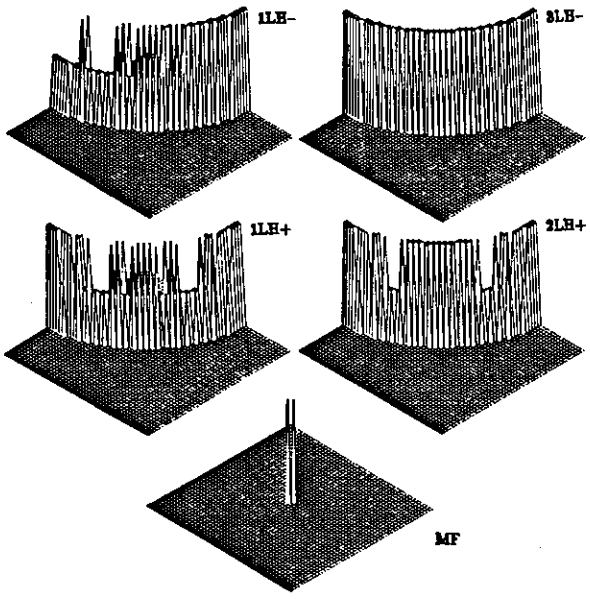


Fig. 5. Ring, w=2, filtered once with 1LH–, 3LH–, 1LH+, 2LH+ and MF.

where $f(m,n)$ is a pixel of the image and w is the width of the ring. We have tested the performance of the following 5×5 filters: 1LH– (Fig. 1 (a)), 3LH–, 1LH+ (Fig. 2 (a)), 2LH+ and MF.

The ring image with $w = 3$ is a root signal to all the filters. The ring with $w = 2$ filtered with the 1LH–, the 3LH–, the 1LH+, the 2LH+, and the MF are shown in Fig. 5. All other filters except the 3LH– modify the image. The ring filtered with the 1LH+ shows clearly that the parts of the edge parallel to the subfilters are preserved. Same effect is also seen with the 1LH– filter. The 2LH+ preserves the root structures of the 1LH+ as well as the R1LH+ . The MF is not able to preserve the ring image, because the ring is smaller than half of the window.

The noise attenuation as function of the relative computer time is shown in Fig.6. The 1LH+ attenuates noise most efficiently. Same attenuation is achieved with the 1LH+ ten times faster than with the MF. Also the 2LH+ attenuates noise somewhat faster than the MF.

Furthermore, the FMH filters have been analyzed with real test images. First we added white saturation distortion to each pixel of the original image with the probability $p = 0.1$ (Fig. 7 (b)). When the noisy image is filtered with the 3×3 2LM+ (Fig. 7 (c)) quite many of the white dots have disappeared and the details are preserved extremely well. To achieve a better noise attenuation we apply once the 5×5 2LH+ to the filtered image of Fig. 7 (c) (Fig. 7 (d)). The details of the original image are still preserved. The 5×5 2LH+ filter can be applied also twice (Fig. 7 (e)) without distorting the image and now almost all the white dots have disappeared. For comparason, the noisy image was filtered with the 5×5 square MF (Fig. 7 (f)). The noise attenuation is about the same as achieved in Fig. 7 (e) but the details of the image are blurred.

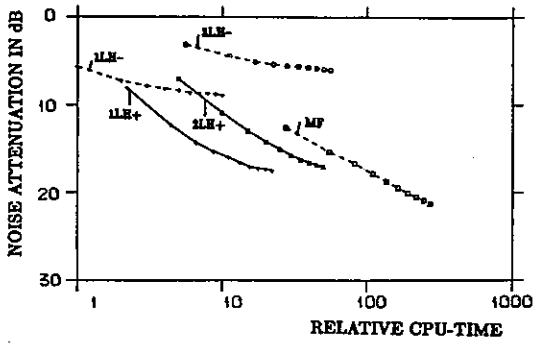


Fig. 6. Noise attenuation as a function of relative cpu-time. 1LH–, 3LH–, 1LH+, 2LH+ and MF.

V. DISCUSSION

In this paper we have tested a new class of median-type filters, FIR-median hybrid filters (FMH), for image processing. In the filters linear substructures are used in conjunction with the median operation. The FMH filters can be divided into two subclasses, the bidirectional and unidirectional FMH filters, which preserve subtle details in a different way. The algorithm can be one-level or tree structured multilevel median operation. The multilevel structure combines the desirable root signal properties of several different one-level FMH structures. The multilevel filtering concept was extended to the median filters.

Some 5×5 FMH filters were compared with the 5×5 MF. FMH filters were shown to be computationally much more efficient and to preserve more subtle details than the MF. Also the noise cleaning ability with respect to the required computer time was shown to be better with the one-level bidirectional FMH filter 1LH+ and two-level bidirectional FMH filter 2LH+ than with the reference filter.

In the area of FMH filters there are many obvious directions for future research. Especially the tradeoff between noise attenuation and direction insensitivity needs to be further studied by using different sized averaging subfilters and various level operations.

REFERENCES

[1] Ritenour, E.R., Nelson, T.R., and Raff, U.: "Applications of the Median Filter to Digital Radiographic Images," *Proc. IEEE ICASSP-84*, San Diego, CA, March 1984, pp. 23.1.1–4.

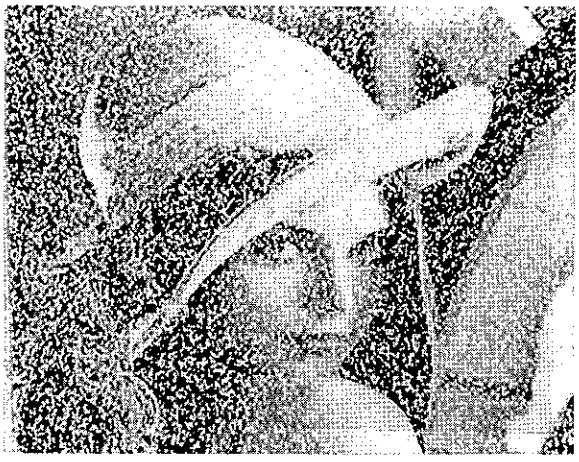
[2] Huang, T.S., Ed., *Topics in Applied Physics, Two Dimensional Digital Signal Processing II*, Berlin: Springer-Verlag, 1981.

[3] Heinonen, P. and Neuvo, Y., "Smoothed Median Filters with FIR Substructures," *Proc. IEEE ICASSP–85*, Tampa, Florida, USA, March 1985, pp. 49–53.

[4] Heinonen, P. and Neuvo, Y., "New Median Type Filters for Image Processing," *Proc. IEEE ISCAS-85*, Kyoto, Japan, June 1985, pp 1329–1331.

[5] Nieminen, A., Heinonen, P. and Neuvo, Y., "A New Class of Detail-Preserving Filters for Image Processing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (in press).

(a)



(b)

(c)

(d)

(e)

(f)

Fig. 7. (a) Original 5×5 image with 8 bits resolution. (b) Each pixel of the original image has been changed to white with the probability $p = 0.1$. (c) The noisy image 7 (b) filtered with the 3×3 2LM+ filter. (d) The image 7 (c) filtered once with the 5×5 2LH+ filter. (e) The image 7 (c) filtered twice with the 5×5 2LH+ filter. (f) The noisy image 7 (b) filtered once with the 5×5 median filter.

AUTHOR INDEX