AN EFFICIENT JACOBI-TYPE ALGORITHM FOR BLIND EQUALIZATION OF PARAUNITARY CHANNELS

Mikael Sørensen, Lieven De Lathauwer, Luc Deneire

Laboratoire I3S, CNRS/UNSA
Les Algorithmes - Euclide-B, 06903 Sophia Antipolis, France
{sorensen, deneire}@i3s.unice.fr
K.U.Leuven - E.E. Dept. (ESAT) - SCD-SISTA
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
Lieven.DeLathauwer@kuleuven-kortrijk.be

ABSTRACT

Blind equalization of convolutive mixtures is often done by resorting to methods based on higher order statistics. Under the assumption that the data have been pre-whitened the problem reduces to the estimation of paraunitary channels. The method PAJOD was developed to equalize paraunitary channels in [8]. Our contribution is an efficient implementation of the PAJOD algorithm which is called PAJOD2. Comparisons between PAJOD and PAJOD2 based on computer simulations will also be reported.

1. INTRODUCTION

Blind equalization of linear time-invariant Multiple Input Multiple Output (MIMO) channels refers to channel equalization techniques where only the observed signal is known. The observed signal is assumed to consist of an unknown convolutive mixture of input signals.

A common strategy in blind separation of an overdetermined instantaneous mixture of statistically independent signals is to decorrelate the data by prewhitening the observed data in an initial stage via for instance an Eigenvalue Value Decomposition (EVD). After the pre-whitening stage the data are uncorrelated and the remaining unitary matrix is resolved by resorting to Higher-Order Statistics (HOS) [3], [6], [12].

This approach have also been adapted to the case of blind equalization of a convolutive mixture of statistically independent signals in [1], [8], [9], [13]. The difference is that after the pre-whitening stage the remaining ambiguity is now a paraunitary matrix and not just a unitary matrix. Hence due to the pre-whitening stage, performed for instance by the algorithm proposed in [14], the problem reduces to a search for a paraunitary equalizer. Again the estimation of the unknown paraunitary matrix can be done by resorting to HOS.

In [8] the cumulant-based algorithm Partial Approximate JOint Diagonalization (PAJOD) was proposed for blind equalization of a paraunitary channel. The PAJOD algorithm applies a Jacobi-type procedure where one of the Jacobi subproblems is solved by a computationally demanding resultant based procedure. It requires the rooting of either a 3rd or 24th degree polynomial in each of its Jacobi subproblems as will be explained later.

Our contribution is a more efficient implementation of the PAJOD algorithm and we will call the computationally improved version PAJOD2.

The rest of the paper is organized as follows. First the notation and system model used throughout the paper will be introduced. Thereafter a review of the PAJOD followed by the more efficient PAJOD2 algorithm will be presented. Finally a comparison of the PAJOD and PAJOD2 methods based on computer simulations will be reported.

1.1 Notations

Let $\mathbb{N}_+$, $\mathbb{R}$ and $\mathbb{C}$ denote the set of positive integer, real and complex numbers respectively and $\mathrm{i} = \sqrt{-1}$. Furthermore let $(\cdot)^*$, $(\cdot)^T$, $(\cdot)^H$, $(\cdot)^\dagger$, $\mathrm{Re}\{\cdot\}$ and $\|\cdot\|$ denote the conjugate, matrix transpose, matrix conjugate-transpose, pseudo-inverse, real part and the Frobenius norm of a matrix respectively. Let $\mathbf{A} \in \mathbb{C}^{m\times n}$, then let $\mathbf{A}_{ij}$ denote the ith row-jth column entry of $\mathbf{A}$. Moreover, let $\mathbf{A}(:, 1 : N)$ designate the submatrix of a $\mathbf{A}$ consisting of the columns from 1 to N of $\mathbf{A}$.

1.2 System Model

Let $\mathbf{s}(n), \mathbf{x}(n) \in \mathbb{C}^N$ be the symbol and observation vector at time instant $n \in \mathbb{N}_+$ respectively. Assume that $\mathbf{s}(n)$ and $\mathbf{x}(n)$ are related via

$$\mathbf{x}(n) = \sum_{k=0}^{K-1} \mathbf{F}(k)\mathbf{s}(n-k),$$

where $\mathbf{F}(k) \in \mathbb{C}^{N\times N}$. Then the problem is to estimate the symbol sequence $\{\mathbf{s}(n)\}_{n\in\mathbb{N}_+}$ based on the observation sequence $\{\mathbf{x}(n)\}_{n\in\mathbb{N}_+}$ via the FIR equalizer

Research supported by: (1) Research Council K.U.Leuven: GOA-Ambiorics, CoE EF/05/006 Optimization in Engineering (OPTEC), CIF1, STRT1/08/23, (2) F.W.O.: (a) project G.0321.06, (b) Research Communities ICCoS, ANMMM and MLDM, (3) the Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, "Dynamical systems, control and optimization", 2007–2011), (4) EU: ERNSI. The work of Mikael Sørensen is supported by the EU by a Marie-Curie Fellowship (EST-SIGNAL program : http://est-signal.i3s.unice.fr) under contract No MEST-CT-2005-021175.

$\{\mathbf{H}(l)\}_{l\in\{0,\ldots,L-1\}} \subset \mathbb{C}^{N\times N}$

$$\begin{aligned}\mathbf{y}(n) &= \sum_{l=0}^{L-1} \mathbf{H}(l)\mathbf{x}(n-l) \\ &= \sum_{l=0}^{L-1}\sum_{k=0}^{K-1} \mathbf{H}(l)\mathbf{F}(k)\mathbf{x}(n-l-k),\end{aligned}$$

where $\mathbf{y}(n)$ is the recovered symbol vector at time instant n, and under the assumptions:

- $\mathbf{s}_i(n)$ are mutually independent i.i.d., zero-mean processes with unit-variance for all $i \in \{0,\ldots,N-1\}$.
- $\mathbf{s}(n)$ is stationary up to order $r = 4$ and hence the marginal cumulants of order $r = 4$ do not depend on n.
- At most one source has zero marginal cumulant of order $r = 4$.
- The global transfer matrix $\mathbf{G}(z) = \mathbf{F}(z)\mathbf{H}(z)$ is paraunitary and hence the equalizer $\mathbf{H}(z)$ is paraunitary since $\mathbf{F}(z)$ is paraunitary by assumption.

2. PAJOD

The notion of contrast optimization was introduced in [10] and applied in the framework of MIMO equalization in [7]. Under the assumption that there exists an equalizer that will fully recover the symbols, an equalizer corresponding to the global maximum of the contrast function is guaranteed to recover the symbol sequence, see [7] for details.

In [8] it was shown that if the cumulants of the observed data are stored in a set of $NL \times NL$ matrices, denoted $\mathcal{M}(\mathbf{b},\gamma)$, in such a way that for a fixed pair $(\mathbf{b},\gamma) = ([b_1,b_2],[\gamma_1,\gamma_2])$ we have the relation

$$\mathcal{M}_{\alpha_1 N+a_1,\alpha_2 N+a_2}(\mathbf{b},\gamma) = \mathrm{Cum}[y_{a_1}(n-\alpha_1), y^*_{a_2}(n-\alpha_2), y_{b_1}(n-\gamma_1), y^*_{b_2}(n-\gamma_2)].$$

Then the function

$$\mathcal{J}_2^2 = \sum_{\mathbf{b},\gamma} \|\mathrm{Diag}\left(\mathbb{H}\mathcal{M}(\mathbf{b},\gamma)\mathbb{H}^{\mathbf{H}}\right)\|^2, \quad (1)$$

where $\|\mathrm{Diag}(\mathbf{A})\|^2 = \sum_i |\mathbf{A}_{ii}|^2$ and $\mathbb{H} = [\mathbf{H}(0),\mathbf{H}(1),\ldots,\mathbf{H}(L-1)] \in \mathbb{C}^{N\times NL}$ is a contrast function. Furthermore it was shown that $\mathbb{H}$ is a semi-unitary matrix, i.e. $\mathbb{H}\mathbb{H}^H = \mathbf{I}$. A matrix for a fixed pair $(\mathbf{b},\gamma)$ will in the following be referred to as a matrix slice of $\mathcal{M}$ and will be denoted by $\mathbf{M}^{(p)}$ where the upper index indicates pth slice of $\mathcal{M}$.

2.1 Jacobi Procedure for Semi-Unitary matrices

To numerically find the semi-unitary matrix $\mathbb{H}$ that will maximize the contrast (1) a Jacobi procedure was proposed in [8]. This procedure can be seen as a double extension of the JADE algorithm [3],[4]. First, the unknown matrix is semi-unitary instead of unitary. Second, only the N first diagonal entries are of interest.

A Jacobi procedure is based on the fact that any $NL \times NL$ unitary matrix with determinant equal to one can be parametrized as a product of Givens rotations [11]:

$$\mathbb{V} = \prod_{i=1}^{NL-1}\prod_{j=i+1}^{NL} \Theta[i,j]^H,$$

where $\Theta[i,j]$ is equal to the identity matrix, except for

$$\begin{aligned}&\Theta_{ii}[i,j] = \Theta_{jj}[i,j] = \cos(\theta[i,j]),\\ &\Theta_{ij}[i,j] = -\Theta_{ji}[i,j]^* = \sin(\theta[i,j])e^{\mathrm{i}\phi[i,j]}, \quad \theta[i,j],\phi[i,j] \in \mathbb{R}.\end{aligned}$$

Let $\mathbb{V}$ denote the product of Givens matrices with the initial value $\mathbb{V} = \mathbf{I}_{NL}$. The updating rules are given by $\mathbb{V} \leftarrow \Theta[i,j]^H\mathbb{V}$ and $\mathcal{M}(\mathbf{b},\gamma) \leftarrow \Theta[i,j]^H\mathcal{M}(\mathbf{b},\gamma)\Theta[i,j]$. In the proposed PAJOD algorithm the semi-unitary matrix $\mathbb{H}$ is determined as the first N rows of the unitary matrix $\mathbb{V}$ that maximizes

$$\begin{aligned}\mathcal{J}_2^2(i,j) &= \sum_{\mathbf{b},\gamma}\sum_{k=1}^{N} \left|\left(\Theta[i,j]^H\mathcal{M}(\mathbf{b},\gamma)\Theta[i,j]\right)_{kk}\right|^2 \\ &= \sum_{\mathbf{b},\gamma}\sum_{k=1}^{N} \left|\sum_{\eta,\mu=1}^{NL} \Theta_{\eta k}[i,j]^*\Theta_{\mu k}[i,j]\mathcal{M}_{\eta\mu}(\mathbf{b},\gamma)\right|^2 \\ &= \sum_{p}\sum_{k=1}^{N} \left|\sum_{\eta,\mu=1}^{NL} \Theta^*_{\eta k}[i,j]\Theta_{\mu k}[i,j]\mathbf{M}^{(p)}_{\eta\mu}\right|^2. \quad (2)\end{aligned}$$

The problem is illustrated in figure 1 for the case when $j \le N$. Since plane rotations where $i > N$ do not have any effect on the first N rows of the matrix slices of $\mathcal{M}$ only Givens rotations where $i \le N$ are considered. Furthermore one has to distinguish between the cases where $j \le N$ and $j > N$.

Let $\overline{\mathbf{M}}^{(p)} = \Theta[i,j]^H\mathbf{M}^{(p)}\Theta[i,j]$ and for notational convenience let $c = cos(\theta[i,j])$ and $s = \sin(\theta[i,j])e^{\mathrm{i}\phi[i,j]}$. Then for the case where $j \le N$ equation (2) is equal to

$$\mathcal{J}_2^2(i,j)\Big|_{j\le N} = \sum_p |\overline{\mathbf{M}}^{(p)}_{ii}|^2 + |\overline{\mathbf{M}}^{(p)}_{jj}|^2 + cst, \quad (3)$$

where $cst \in \mathbb{R}$ is independent of $\Theta[i,j]$ and

$$\overline{\mathbf{M}}^{(p)}_{ii} = \begin{bmatrix} c & -s \end{bmatrix} \begin{bmatrix} \mathbf{M}^{(p)}_{ii} & \mathbf{M}^{(p)}_{ij} \\ \mathbf{M}^{(p)}_{ji} & \mathbf{M}^{(p)}_{jj} \end{bmatrix} \begin{bmatrix} c \\ -s^* \end{bmatrix}$$

and

$$\overline{\mathbf{M}}^{(p)}_{jj} = \begin{bmatrix} s^* & c \end{bmatrix} \begin{bmatrix} \mathbf{M}^{(p)}_{ii} & \mathbf{M}^{(p)}_{ij} \\ \mathbf{M}^{(p)}_{ji} & \mathbf{M}^{(p)}_{jj} \end{bmatrix} \begin{bmatrix} s \\ c \end{bmatrix}.$$

The maximization problem (3) is equivalent to the JADE diagonalization problem and therefore the JADE algorithm [3] can be applied to solve this problem.

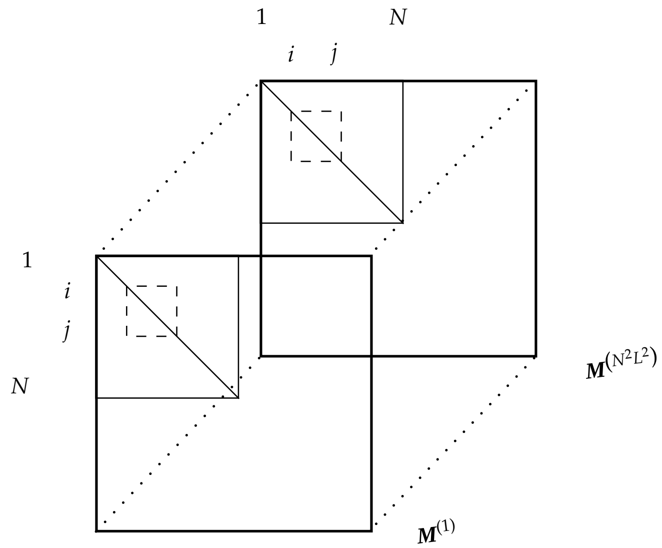

Figure 1: The figure illustrates the PAJOD optimization problem for the case when $j \leq N$ and $\mathbf{b}$ varies in $\{1,\ldots,N\}^2$ and γ in $\{0,,\ldots,L-1\}^2$. The aim of the Givens rotation matrix $\Theta[i,j]$ is to jointly diagonalize the set of matrices $\{\mathbf{M}^{(p)}\}$ by maximizing the entries $\mathbf{M}_{ii}^{(p)}$ and $\mathbf{M}_{jj}^{(p)}$ for all p.

When $j > N$ only the first diagonal term should be maximized, and the equation (2) reduces to

$$\mathcal{J}_2^2(i,j)\Big|_{j>N} = \sum_p |\overline{\mathbf{M}}_{ii}^{(p)}|^2. \tag{4}$$

In [8] a resultant based [5] approach was taken to solve the maximization problem (4). It amounted to the rooting of a 24th order degree polynomial containing at most 8 real roots.

An outline of the PAJOD algorithm can be seen in algorithm 1.

Algorithm 1 Outline of the PAJOD procedure.

Estimate the cumulant tensor $\mathcal{M}(\mathbf{b},\gamma)$
Initialize $\mathbb{V} = \mathbf{I}_{NL}$
Step 1: Repeat until convergence
for $i = 1$ to N **do**
  for $j = i+1$ to NL **do**
    if $j \leq N$ **then**
      calculate optimal $\Theta[i,j]$ from $\mathcal{J}_2^2(i,j)\Big|_{j\leq N}$
    else
      calculate optimal $\Theta[i,j]$ from $\mathcal{J}_2^2(i,j)\Big|_{j>N}$
    end if
    $\mathbf{M}^{(p)} \leftarrow \Theta[i,j]^H \mathbf{M}^{(p)} \Theta[i,j]$
    $\mathbb{V} \leftarrow \Theta[i,j]^H \mathbb{V}$
  end for
end for
Step 2: Check if algorithm has converged. If not, then go to Step 1.
Set $\mathbb{H} = \mathbb{V}(:,1:N)$

3. PAJOD2

This section will introduce the PAJOD2 algorithm which is a computational improved version of the PAJOD algorithm. When $j \leq N$, then the PAJOD2 algorithm will apply the JADE algorithm to solve the Jacobi subproblem, just as in the PAJOD case.

When $j > N$ a more efficient eigenvector based approach will be proposed. In the derivation we will make use of the trigonometric identities

$$\begin{aligned}
2\cos^2(\theta) &= (1+\cos(2\theta)) \\
2\sin^2(\theta) &= (1-\cos(2\theta)) \\
2\cos(\theta)\sin(\theta) &= \sin(2\theta) \\
\cos(2\phi) &= \cos^2(\phi) - \sin^2(\phi) \\
\sin(2\phi) &= 2\cos(\phi)\sin(\phi)
\end{aligned}$$

Let $\theta = \theta[i,j], \phi = \phi[i,j]$, $\hat{c} = \cos(2\theta)$, $\hat{s} = \sin(2\theta)$, $\alpha^{(p)} = \mathbf{M}_{ii}^{(p)} - \mathbf{M}_{jj}^{(p)}$ and $\beta^{(p)} = \mathbf{M}_{ii}^{(p)} + \mathbf{M}_{jj}^{(p)}$, then equation (4) can be written as

$$\begin{aligned}
\mathcal{J}_2^2(i,j)\Big|_{j>N} = \frac{1}{4}\sum_p & \left|\beta^{(p)}\right|^2 + \left|\alpha^{(p)}\right|^2 \hat{c}^2 + 2\mathrm{Re}\left\{\beta^{(p)*}\alpha^{(p)}\right\}\hat{c} \\
& + \left(\left|\mathbf{M}_{ij}^{(p)}\right|^2 + \left|\mathbf{M}_{ji}^{(p)}\right|^2 + 2\mathrm{Re}\left\{\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)} e^{\mathrm{i}2\phi}\right\}\right)\hat{s}^2 \\
& + 2\mathrm{Re}\left\{\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)*}e^{\mathrm{i}\phi} + \mathbf{M}_{ji}^{(p)*}e^{-\mathrm{i}\phi}\right)\right\}\hat{c}\hat{s} \\
& - 2\mathrm{Re}\left\{\beta^{(p)*}\left(\mathbf{M}_{ij}^{(p)}e^{-\mathrm{i}\phi} + \mathbf{M}_{ji}^{(p)}e^{\mathrm{i}\phi}\right)\right\}\hat{s}.
\end{aligned} \tag{5}$$

By inspection of (5) we can identify the constant term as

$$k = \frac{1}{4}\sum_p \left|\beta^{(p)}\right|^2 = \frac{1}{4}\sum_p |\mathbf{M}_{ii}^{(p)} + \mathbf{M}_{jj}^{(p)}|^2. \tag{6}$$

The linear terms in the variables $\hat{c}$ and $\hat{s}$ of (5) can be written as

$$\begin{aligned}
L &= \frac{1}{2}\sum_p \mathrm{Re}\left\{\beta^{(p)*}\alpha^{(p)}\right\}\hat{c} - \mathrm{Re}\left\{\beta^{(p)*}\left(\mathbf{M}_{ij}^{(p)}e^{-\mathrm{i}\phi} + \mathbf{M}_{ji}^{(p)}e^{\mathrm{i}\phi}\right)\right\}\hat{s} \\
&= \frac{1}{2}\sum_p \mathrm{Re}\left\{\beta^{(p)*}\alpha^{(p)}\right\}\hat{c} - \mathrm{Re}\left\{\beta^{(p)*}\left(\mathbf{M}_{ij}^{(p)} + \mathbf{M}_{ji}^{(p)}\right)\right\}\hat{s}\cos(\phi) \\
&+ \mathrm{Re}\left\{\mathrm{i}\beta^{(p)*}\left(\mathbf{M}_{ij}^{(p)} - \mathbf{M}_{ji}^{(p)}\right)\right\}\hat{s}\sin(\phi) \\
&= \sum_p \mathbf{g}^{(p)T}\mathbf{v},
\end{aligned} \tag{7}$$

where

$$\mathbf{v} = \begin{bmatrix} \cos(2\theta[i,j]) \\ \sin(2\theta[i,j])\cos(\phi[i,j]) \\ \sin(2\theta[i,j])\sin(\phi[i,j]) \end{bmatrix}$$

$$\mathbf{z}^{(p)} = \frac{1}{2}\begin{bmatrix} \mathbf{M}_{ii}^{(p)} - \mathbf{M}_{jj}^{(p)} \\ -(\mathbf{M}_{ij}^{(p)} + \mathbf{M}_{ji}^{(p)}) \\ \mathrm{i}(\mathbf{M}_{ij}^{(p)} - \mathbf{M}_{ji}^{(p)}) \end{bmatrix}$$

$$\mathbf{g}^{(p)} = \mathrm{Re}\left\{\left(\mathbf{M}_{ii}^{(p)} + \mathbf{M}_{jj}^{(p)}\right)^* \mathbf{z}^{(p)}\right\}$$

The quadratic term of (5) will now be written as

$$\begin{aligned}
Q &= \frac{1}{4}\sum_p \left|\alpha^{(p)}\right|^2 \hat{c}^2 + 2\mathrm{Re}\left\{\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)*}e^{\mathrm{i}\phi} + \mathbf{M}_{ji}^{(p)*}e^{-\mathrm{i}\phi}\right)\right\}\hat{c}\hat{s} \\
&+ \left(\left|\mathbf{M}_{ij}^{(p)}\right|^2 + \left|\mathbf{M}_{ji}^{(p)}\right|^2 + 2\mathrm{Re}\left\{\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)}e^{\mathrm{i}2\phi}\right\}\right)\hat{s}^2 \\
&= \frac{1}{4}\sum_p \left|\alpha^{(p)}\right|^2 \hat{c}^2 + \left(\left|\mathbf{M}_{ij}^{(p)}\right|^2 + \left|\mathbf{M}_{ji}^{(p)}\right|^2\right)\hat{s}^2\left(\cos^2(\phi) + \sin^2(\phi)\right) \\
&+ 2\mathrm{Re}\left\{\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)}\right\}\hat{s}^2\cos(2\phi) + 2\mathrm{Re}\left\{\mathrm{i}\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)}\right\}\hat{s}^2\sin(2\phi) \\
&+ 2\mathrm{Re}\left\{\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)} + \mathbf{M}_{ji}^{(p)}\right)^*\right\}\hat{c}\hat{s}\cos(\phi) \\
&+ 2\mathrm{Re}\left\{\mathrm{i}\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)} - \mathbf{M}_{ji}^{(p)}\right)^*\right\}\hat{c}\hat{s}\sin(\phi) \\
&= \frac{1}{4}\sum_p \left|\alpha^{(p)}\right|^2 \hat{c}^2 + \left(\left|\mathbf{M}_{ij}^{(p)}\right|^2 + \left|\mathbf{M}_{ji}^{(p)}\right|^2\right)\hat{s}^2\left(\cos^2(\phi) + \sin^2(\phi)\right) \\
&+ 2\mathrm{Re}\left\{\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)}\right\}\hat{s}^2\left(\cos^2(\phi) - \sin^2(\phi)\right) \\
&+ 4\mathrm{Re}\left\{\mathrm{i}\mathbf{M}_{ij}^{(p)*}\mathbf{M}_{ji}^{(p)}\right\}\hat{s}^2\cos(\phi)\sin(\phi) \\
&+ 2\mathrm{Re}\left\{\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)} + \mathbf{M}_{ji}^{(p)}\right)^*\right\}\hat{c}\hat{s}\cos(\phi) \\
&+ 2\mathrm{Re}\left\{\alpha^{(p)}\left(\mathbf{M}_{ij}^{(p)} - \mathbf{M}_{ji}^{(p)}\right)^*\right\}\hat{c}\hat{s}\sin(\phi) \\
&= \mathbf{v}^T\sum_p \mathbf{G}^{(p)}\mathbf{v}, \qquad (8)
\end{aligned}$$

where $\mathbf{G}^{(p)} = \mathrm{Re}\left\{\mathbf{z}^{(p)}\mathbf{z}^{(p)H}\right\}$.

From the equations (6), (7) and (8), equation (4) can be reformulated as

$$\mathcal{J}_2^2(i,j)\Big|_{j>N} = \mathbf{v}^T\mathbf{G}\mathbf{v} + \mathbf{g}^T\mathbf{v} + k, \qquad (9)$$

where $\mathbf{G} = \sum_p \mathbf{G}^{(p)}$ and $\mathbf{g} = \sum_p \mathbf{g}^{(p)}$. We should maximize (9) under the constraint $\|\mathbf{v}\| = 1$. A problem of the same form appeared in [2].

Maximizing (9) subject to the constraint that $\|\mathbf{v}\|^2 = 1$ using the Lagrange multiplier method leads to

$$2(\mathbf{G} + \lambda\mathbf{I})\mathbf{v} + \mathbf{g} = 0,\ \lambda \in \mathbb{R}. \qquad (10)$$

Assume that $(\mathbf{G} + \lambda\mathbf{I})^{-1}$ exists, we have that

$$\mathbf{v} = -\frac{1}{2}(\mathbf{G} + \lambda\mathbf{I})^{-1}\mathbf{g}.$$

Given the EVD $\mathbf{G} = \mathbf{E}\Lambda\mathbf{E}^T$ we have

$$\|\mathbf{v}\|^2 = 1 \Leftrightarrow \frac{1}{4}\sum_{i=1}^{3}\frac{\left(\mathbf{E}_i^T\mathbf{g}\right)^2}{(\Lambda_{ii} + \lambda)^2} = 1, \qquad (11)$$

where $\mathbf{E}_i$ and Λ_{ii} denote the ith eigenvector and eigenvalue of $\mathbf{G}$ respectively. From (11) one can deduce that the problem amounts to rooting a polynomial of degree 6 and thereafter selecting the root of the corresponding $\mathbf{v}$ which maximizes $\mathcal{J}_2^2(i,j)$.

If $(\mathbf{G} + \lambda\mathbf{I})^{-1}$ does not exist[1], which could occur if $\lambda = 0$ and $\mathbf{G}$ is singular or when $\lambda = -\Lambda_{ii}$ for some i, then we have to resort to (10) for the computation of $\mathbf{v}$:

$$\mathbf{v} = -\frac{1}{2}(\mathbf{G} - \Lambda_{ii}\mathbf{I})^{\dagger}\mathbf{g} + c_i\mathbf{E}_i,$$

where c_i is a real constant chosen such that $\|\mathbf{v}\| = 1$ and $\mathcal{J}_2^2(i,j)$ is maximum. If it exists, then it is given by

$$c_i = \mathrm{sign}(\mathbf{E}_i^T\mathbf{g})\sqrt{1 - \|(\mathbf{G} - \Lambda_{ii}\mathbf{I})^{\dagger}\mathbf{g}\|^2/4}.$$

4. COMPUTER RESULTS

Our simulations will based on on 2-Input-2-Output channels ($N = 2$), the channels and equalizers were of the same length ($K = L$) and the data blocks consists each of a QPSK sequence of 512 symbols. The paraunitary channel is generated, just as in [8], as follows:

$$\mathbf{F}(z) = \mathbf{R}(\phi_0)\prod_{m=1}^{L-1}\mathbf{Z}(z)\mathbf{R}(\phi_m)$$

where

$$\mathbf{Z}(z) = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}, \mathbf{R}(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi)\mathrm{e}^{-\mathrm{i}\theta} \\ \sin(\phi)\mathrm{e}^{\mathrm{i}\theta} & \cos(\phi) \end{bmatrix}$$

and the parameters ϕ_i and θ_i are drawn according to a uniform distribution in $[0, 2\pi)$. The filtered QPSK sequences of unit variance are perturbed by an additive white circular complex Gaussian noise with identity covariance matrix.

The algorithms PAJOD and PAJOD2 are first tested on 100 random channels of varying SNR. To measure the elapsed time used to execute the algorithms in MATLAB, the built-in functions *tic*(·) and *toc*(·) are used and the mean and median time results can be seen in figure 2 and 3 respectively. By inspection of the figures it can be seen that the PAJOD2 algorithm is cheaper than the PAJOD algorithm.

A second simulation was conducted in order to investigate the computation time of the algorithms as a function of the filter length. The SNR was fixed to 10 while the filter and channel length varied from 2 to 8 with a hop factor of 1. The mean computation time over 10 simulations result can be seen in figure 4. Here it can be seen that the computational complexity of the PAJOD2 method is consistently lower than the PAJOD method when L is increasing.

[1] This case has not been observed in our simulations.

5. SUMMARY

The problem of blind equalization of paraunitary channels was addressed by the PAJOD approach. After a review of PAJOD method we proposed a computationally more efficient method called PAJOD2. The proposed method simplified the Jacobi-subproblem from the rooting of a 24th degree polynomial to the the rooting of a polynomial of degree 6. Furthermore computer simulations confirmed that the PAJOD2 method is consistently faster than the PAJOD method in the given simulations.

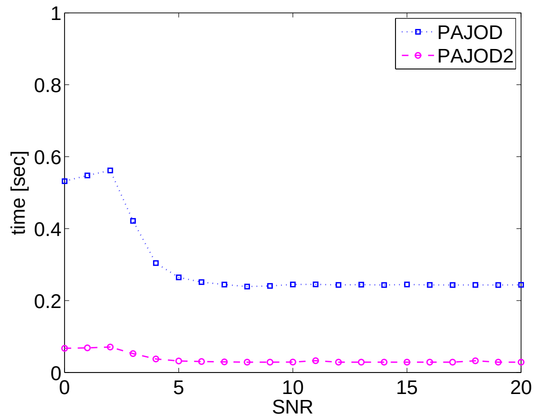

Figure 2: Mean values for the computation time for the simulation as measured by MATLAB.

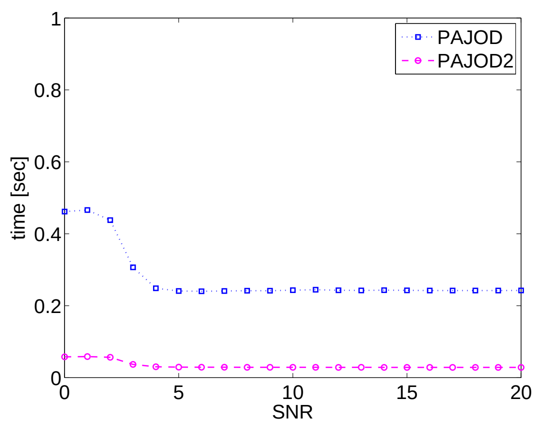

Figure 3: Median values for the computation time for the simulation as measured by MATLAB.

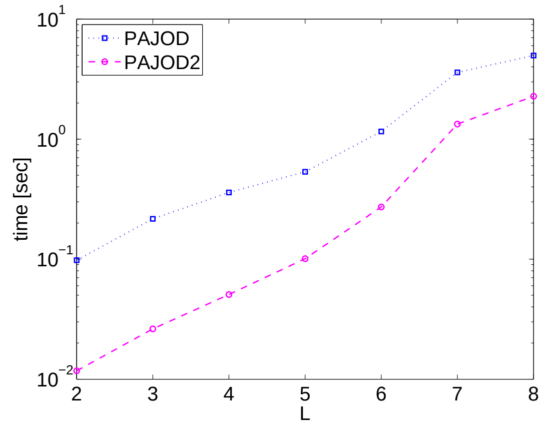

Figure 4: Mean computation time results for filters and channels of varying length L as measured by MATLAB.

REFERENCES

[1] P. D. Baxter and J. G . McWhirter, "Blind Signal Separation of Convolutive Mixtures," in *Proc. Asilomar Conf. Signals, Syst. Comput.* , pp. 124–128, Nov, 2003.

[2] A. Belouchrani and K. Abed Meraim and Y. Hua, "Jacobi-like Algorithms for Joint Block Diagonalization: Applications to Source Localization," in *Proc. IEEE Int. Work. on Intelligent Sig. Proc. and Comm. Sys. (ISPACS)*, 1998.

[3] J.-F. Cardoso and A. Souloumiac , "Blind Beamforming for non-Gaussian Signals," *IEE Proceedings-F*, vol. 140, pp. 362–370 . 1993.

[4] J.-F. Cardoso and A. Souloumiac , "Jacobi Angles for Simultaneous Diagonalization," *SIAM J. Matrix Anal. Appl.*, vol. 17, pp.161–164 . 1996.

[5] D. Cox and J. Little and D. O'Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. New York: Springer Verlag, 1996.

[6] P. Comon, "Independent Component Analysis, a new concept ?," *Signal Processing*, vol. 36, pp. 287–314, April. 1994.

[7] P. Comon, "Contrasts for Multichannel Blind Deconvolution," *IEEE Signal Processing Letters*, vol. 3, pp. 209–211, July. 1996.

[8] P. Comon and L. Rota, "Blind Separation of Independent Sources from Convolutive Mixtures," *IEICE Trans. Fundamentals*, vol. E86-A, pp. 542–549, March. 2003.

[9] L. De Lathauwer and B. De Moor and J. Vandewalle, "An Algebraic Approach to Blind MIMO Identification," in *Proc. ICA2000* , pp. 211–214, June, 2000.

[10] D. Donoho, "On Minimum Entropy Deconvolution," *Applied Time-Series Analysis II*, pp. 565-608. 1981.

[11] G.H. Golub and C.F. Van Loan, *Matrix Computations*. Baltimore: The John Hopkins University Press, 1996.

[12] A. Hyvärinen and E. Oja, "Fast Fixed-Point Algorithm for Independent Component Analysis," *Neural Computation*, vol.9, pp. 1483–1492, 1997.

[13] S. Icart and P. Comon and L. Rota, "Blind Paraunitary Equalization," *Signal Processing*, vol.89, pp. 283–290, March. 2009.

[14] J. G . McWhirter and P. D. Baxter and T. Cooper and J. Foster, "An EVD Algorithm for Para-Hermitian Polynomial Matrices ," *IEEE Trans. Signal Process.*, vol.55, pp. 2158–2169, March. 2007.