15th European Signal Processing Conference (EUSIPCO 2007), Poznan, Poland, September 3-7, 2007, copyright by EURASIP

NEW INSIGHTS IN ADAPTIVE CASCADED FIR STRUCTURE: APPLICATION TO FULLY ADAPTIVE INTERPOLATED FIR STRUCTURES

Eduardo L. O. Batista, Orlando J. Tobias, and Rui Seara

LINSE – Circuits and Signal Processing Laboratory
Department of Electrical Engineering
Federal University of Santa Catarina
88040-900 – Florianópolis – SC – Brazil
E-mails: {dudu, orlando, seara}@linse.ufsc.br

ABSTRACT

This paper presents the development of the least-mean-square (LMS) and normalized LMS (NLMS) algorithms for adapting cascaded FIR filters and the application of such algorithms to the whole adaptation of interpolated FIR filters. The obtained expressions are general and can be extended to any structure composed of the cascade of two FIR filters. The considered approach allows verifying the main characteristics of the adaptive process as well as the limitations of the existing adaptive interpolated FIR structures using adaptive interpolators. Numerical simulation results are presented aiming to confirm the effectiveness of the obtained advances.

1. INTRODUCTION

Interpolated finite impulse response (IFIR) filters are computationally efficient structures, being an interesting alternative to implement FIR filters. The first work on this subject is due to Neuvo *et al*. [1]. Since then, much research effort in IFIR filters has been carried out, aiming to use them in a large number of coefficient-demanding applications, such as line echo canceling [2]-[4], active control [5], and audio processing on a digital hearing aid [6]. The idea behind IFIR filters is to use a sparse FIR filter (with a reduced number of coefficients) cascaded with an interpolator filter that recreates the removed coefficients in an approximate form.

The adaptive version of an IFIR (AIFIR) filter also represents an interesting alternative to implementing coefficient-demanding adaptive FIR (AFIR) filters [2]. An AIFIR filter is carried out by just adapting the sparse filter coefficients whereas the interpolator is maintained fixed [7]. The position of the filters in the cascade can be exchanged, placing the interpolator at the input [8] or at the output [7] of the AIFIR structure. The interpolator position impacts on different update processes because of the time-varying nature of the structure [7], [8]. Despite that, the same steady-state performance is obtained for both realizations.

The computational complexity reduction in an AIFIR structure is obtained at the expense of a higher steady-state mean-square error (MSE) value as compared with that of the standard AFIR filter. Such degradation is due to the use of a fixed interpolator filter, which generally leads to an inadequate recreation of coefficients. A possible solution to this problem is to consider a fully adapted IFIR (FAIFIR) structure, in which both the sparse and interpolator filters are adapted. This class of structure was originally introduced in [9] and applied to an adaptive line enhancer. In [10] and [11], a FAIFIR structure has been introduced and discussed in an ad hoc manner.

Regarding the mathematical treatment, this paper presents the derivation of the least-mean-square (LMS) and normalized LMS (NLMS) algorithms for adapting FIR filters in cascade as well as its application to FAIFIR filters. The presented framework is general, permitting to be applied in any structure formed by a cascade of two adaptive FIR filters. In particular, the provided expressions are applicable to FAIFIR structures regardless of the interpolator and sparse filter positions. Moreover, through the proposed approach more insight on the adaptation process of such filters is gained, permitting to assess the approximations used in [9]-[11].

This paper is organized as follows. In Section 2, the general mathematical description of the cascaded FIR filters is introduced. Section 3 presents the development of the LMS algorithm for such a structure. Section 4 considers the case of the NLMS algorithm. The developed theory for the FAIFIR case is discussed in Section 5. Section 6 shows some simulation results aiming to verify the performance of a FAIFIR structure. Finally, remarks and conclusions of this work are presented in Section 7.

2. MATHEMATICAL TREATMENT OF CASCADED FIR FILTERS

Fig. 1 illustrates the block diagram of a cascade of two FIR filters, where $\mathbf{g} = [g_0 \;\; g_1 \;\; \cdots \;\; g_{M-1}]^{\mathrm{T}}$ and $\mathbf{h} = [h_0 \;\; h_1 \;\; \cdots \;\; h_{N-1}]^{\mathrm{T}}$ are the input and output FIR filters, respectively. Variable $x(n)$ denotes the input signal, $\hat{y}(n)$ represents an intermediary signal, and $y(n)$ is the output signal of the cascade, also obtained by combining the cascade of filters into an equivalent filter, resulting in

$$\mathbf{w}_{\mathrm{eq}} = \mathbf{g} * \mathbf{h} \,. \tag{1}$$

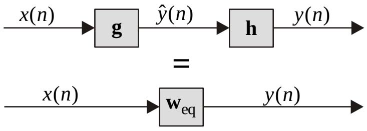

Figure 1 – Block diagram of a cascaded FIR structure and its equivalent filter.

Note from (1) that the equivalent structure is obtained by a convolution operation. To facilitate the mathematical treatment, such an operation is expressed as a product of matrices [9]. For such, let us define the following matrices:

Orlando J. Tobias is also with the Electrical Engineering and Telecom. Dept., Regional University of Blumenau (FURB), Blumenau, SC, Brazil. This work was supported in part by the Brazilian National Research Council for Scientific and Technological Development (CNPq).

$$\mathbf{G}=\begin{bmatrix} g_0 & 0 & 0 & \cdots & 0 \\ g_1 & g_0 & 0 & \cdots & 0 \\ g_2 & g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{M-1} & g_{M-2} & g_{M-3} & \cdots & g_0 \\ 0 & g_{M-1} & g_{M-2} & \cdots & g_1 \\ 0 & 0 & g_{M-1} & \cdots & g_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & g_{M-1} \end{bmatrix} \qquad (2)$$

with dimension $(N+M-1)\times N$, and

$$\mathbf{H}=\begin{bmatrix} h_0 & 0 & 0 & \cdots & 0 \\ h_1 & h_0 & 0 & \cdots & 0 \\ h_2 & h_1 & h_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & h_{N-3} & \cdots & h_0 \\ 0 & h_{N-1} & h_{N-2} & \cdots & h_1 \\ 0 & 0 & h_{N-1} & \cdots & h_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{N-1} \end{bmatrix} \qquad (3)$$

with dimension $(N+M-1)\times M$. Then, (1) is now rewritten as

$$\mathbf{w}_{\mathrm{eq}}=\mathbf{Gh}=\mathbf{Hg}\ . \qquad (4)$$

Due to the convolution operation in both (1) and (4), the equivalent filter has a memory size of $N+M-1$. Now, defining a new input vector with the same dimension of the equivalent filter (4) as

$$\mathbf{x}_{\mathrm{e}}(n)=\left[x(n)\ \ x(n-1)\ \ \cdots\ \ x(n-N-M+2)\right]^{\mathrm{T}} \qquad (5)$$

the input-output relationship of the cascaded structure is written in terms of matrix-vector product as

$$y(n)=\mathbf{w}_{\mathrm{eq}}^{\mathrm{T}}\mathbf{x}_{\mathrm{e}}(n)=\mathbf{h}^{\mathrm{T}}\mathbf{G}^{\mathrm{T}}\mathbf{x}_{\mathrm{e}}(n)=\mathbf{g}^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}\mathbf{x}_{\mathrm{e}}(n)\ . \qquad (6)$$

3. LMS ALGORITHM IN THE CASCADED STRUCTURE

In this section, the adaptive version of the cascade of two FIR filters (see Fig. 1) is derived. For such, we assume that the coefficients of both filters are adjusted according to the LMS algorithm. Fig. 2 shows the block diagram of the structure in question, in which $d(n)$ denotes the signal to be estimated (desired signal) and $e(n)$, the error signal given by

$$e(n)=d(n)-y(n)\ . \qquad (7)$$

The remaining signals in this figure are the same as in Fig. 1.

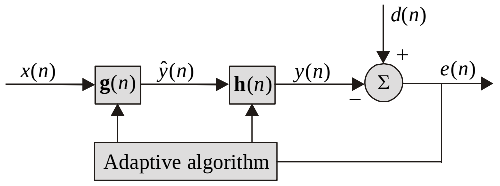

Figure 2 – Block diagram of the adaptive cascaded structure.

By substituting (6) into (7) and considering that now the coefficients are time-varying for both FIR filters, we get

$$\begin{aligned} e(n)&=d(n)-\mathbf{g}^{\mathrm{T}}(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n) \\ &=d(n)-\mathbf{h}^{\mathrm{T}}(n)\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n). \end{aligned} \qquad (8)$$

Now, defining a cost function based on the instantaneous error, one can write

$$\hat{J}_{\mathrm{MSE}}(n)=e^2(n)\ . \qquad (9)$$

Then considering the setup of Fig. 2, the filters' coefficients are updated by using the gradient of the cost function (9) as given in [12]. For such, the required expressions are obtained in the next sections.

3.1 Coefficient update of the input filter

By considering the input filter $\mathbf{g}(n)$ the updating process is described by the gradient rule. Thus,

$$\mathbf{g}(n+1)=\mathbf{g}(n)-\mu_1\nabla_{\mathbf{g}}e^2(n) \qquad (10)$$

with μ_1 representing the step-size parameter for adapting the input filter. By applying the chain rule in (8), the gradient vector is written as

$$\nabla_{\mathbf{g}}e^2(n)=\frac{\partial e^2(n)}{\partial \mathbf{g}(n)}=\frac{\partial e^2(n)}{\partial e(n)}\frac{\partial e(n)}{\partial \mathbf{g}(n)}\ . \qquad (11)$$

Then, from (8), the right hand side terms of (11) are

$$\frac{\partial e^2(n)}{\partial e(n)}=2e(n) \qquad (12)$$

and

$$\frac{\partial e(n)}{\partial \mathbf{g}(n)}=-\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n) \qquad (13)$$

where $\mathbf{H}(n)$ is the time-varying version of (3). Therefore, the LMS-update equation for the input filter of the setup of Fig. 2 is given by

$$\mathbf{g}(n+1)=\mathbf{g}(n)+2\mu_1 e(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\ . \qquad (14)$$

3.2 Coefficient update of the output filter

For the output filter, the update process is again given by

$$\mathbf{h}(n+1)=\mathbf{h}(n)-\mu_2\nabla_{\mathbf{h}}e^2(n) \qquad (15)$$

where μ_2 is the step-size parameter for adjusting the output filter. Then from (8), now considering vector $\mathbf{h}(n)$, according to the chain rule we write

$$\frac{\partial e^2(n)}{\partial \mathbf{h}(n)}=\frac{\partial e^2(n)}{\partial e(n)}\frac{\partial e(n)}{\partial \mathbf{h}(n)} \qquad (16)$$

resulting in (12) and

$$\frac{\partial e(n)}{\partial \mathbf{h}(n)}=-\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n) \qquad (17)$$

where $\mathbf{G}(n)$ is the time-varying version of (2). Finally, the LMS-update equation for the coefficients of the output filter is given by

$$\mathbf{h}(n+1)=\mathbf{h}(n)+2\mu_2 e(n)\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\ . \qquad (18)$$

3.3 Particularities of cascaded adaptive filters

Some important questions arise in the updating process of cascaded adaptive filters:

i) Coefficient initialization. As mentioned in [9], if both coefficient vectors $\mathbf{g}(n)$ and $\mathbf{h}(n)$ are initialized with zeros according to common practice, from (14) and (18) one verifies that such coefficient vectors are unchanged during the updating process. One possible solution is to initialize the coefficient vectors $\mathbf{g}(n)$ and $\mathbf{h}(n)$ (or at least one of them) with nonzero values.

ii) Computational complexity. From (14) and (18) one notices that an evaluation of two matrix products $\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)$ and $\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)$ is required at each iteration. However, under slow adaptation conditions and considering the particular structure of $\mathbf{G}(n)$ and $\mathbf{H}(n)$ matrices, these matrix products can be simplified to an inner vector product, reducing thus the computational burden. Such procedure is discussed in the following sections.

iii) Algorithm stability. Because of the simultaneous adaptation of both filters and the use of two different step-size parameters, to obtain analytical expressions for the stability bounds is somewhat difficult. Then, an alternative solution is to use a conservative step-size value for both filters, obtained experimentally. With such strategy, good practical results are achieved as will be shown in Section 6. A mathematical analysis of the cascaded structure is very complex, thus the determination of stability bounds remains an open problem in the literature of the area.

4. CASCADED NLMS STRUCTURE

In this section, the expressions allowing to adapt the coefficients of the cascaded structure by using the NLMS algorithm are derived. Similarly as in [12], the NLMS updating equation is obtained by minimizing the Euclidean norm of

$$\delta\mathbf{g}(n+1)=\mathbf{g}(n+1)-\mathbf{g}(n) \tag{19}$$

subject to the constraint

$$\mathbf{g}^{\mathrm{T}}(n+1)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)=d(n)\,. \tag{20}$$

Note that the constraint expression (20) is slightly different from that presented in [12] due to the particular characteristics of the cascaded structure. By applying the method of Lagrange multipliers [12] to (19) and (20), the cost function is now given by

$$J_{\mathrm{g}}(n)=\left\|\delta\mathbf{g}(n+1)\right\|^2+\lambda_1\left[d(n)-\mathbf{g}^{\mathrm{T}}(n+1)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right] \tag{21}$$

where λ_1 is a real-valued Lagrange multiplier. By differentiating (21) with respect to $\mathbf{g}(n+1)$, we get

$$\frac{\partial J_{\mathrm{g}}(n)}{\partial\mathbf{g}(n+1)}=2\left[\mathbf{g}(n+1)-\mathbf{g}(n)\right]-\lambda_1\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\,. \tag{22}$$

By setting (22) equal to zero, we obtain

$$\mathbf{g}(n+1)=\mathbf{g}(n)+\frac{1}{2}\lambda_1\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\,. \tag{23}$$

By substituting (23) into (20) results in

$$\begin{aligned}d(n)&=\left(\mathbf{g}(n)+\frac{1}{2}\lambda_1\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right)^{\mathrm{T}}\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\\&=\mathbf{g}^{\mathrm{T}}(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)+\frac{1}{2}\lambda_1\left\|\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right\|^2.\end{aligned} \tag{24}$$

Now, solving (24) for λ_1 we get

$$\lambda_1=\frac{2\left[d(n)-\mathbf{g}^{\mathrm{T}}(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right]}{\left\|\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right\|^2}=\frac{2e(n)}{\left\|\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right\|^2}\,. \tag{25}$$

Finally, substituting (25) into (23), adding a positive constant α_1 for controlling the adaptation process as well as a small positive constant ψ_1, preventing a division by zero [12], the coefficient update equation of the input filter considering an NLMS-like adaptive algorithm is given by

$$\mathbf{g}(n+1)=\mathbf{g}(n)+\frac{\alpha_1}{\left\|\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right\|^2+\psi_1}e(n)\mathbf{H}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\,. \tag{26}$$

In a similar way as (19) and (20), for the output filter we have to minimize

$$\delta\mathbf{h}(n+1)=\mathbf{h}(n+1)-\mathbf{h}(n) \tag{27}$$

subject to the constraint

$$\mathbf{h}^{\mathrm{T}}(n+1)\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}(n)=d(n)\,. \tag{28}$$

Again, by applying the method of Lagrange multipliers [12], one obtains

$$J_{\mathrm{h}}(n)=\left\|\delta\mathbf{h}(n+1)\right\|^2+\lambda_2\left[d(n)-\mathbf{h}^{\mathrm{T}}(n+1)\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right]. \tag{29}$$

Note that (29) has the same form as (21); thus, by achieving a similar development as before, we get the following update equation:

$$\mathbf{h}(n+1)=\mathbf{h}(n)+\frac{\alpha_2}{\left\|\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n)\right\|^2+\psi_2}e(n)\mathbf{G}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n) \tag{30}$$

where α_2 and ψ_2 are small positive constants aiming to control the adaptation process and to prevent a division by zero, respectively.

Regarding practical questions, same considerations as those given in Section 3-3 are also applicable here. However, it is interesting to highlight that, if the simplified form is used, it leads to smaller values for α_1 and α_2 than when using the standard NLMS algorithm to obtain a satisfactory performance.

5. FULLY ADAPTIVE IFIR FILTERS

In this section, by following the procedures previously described, the coefficient update equation is derived for a FAIFIR structure. Fig. 3 shows the block diagram of an IFIR structure, where $\mathbf{w}_{\mathrm{s}}$ characterizes a sparse FIR filter and $\mathbf{i}=[i_0\, i_1\cdots i_{M-1}]^{\mathrm{T}}$, the interpolator filter. The input signal $x(n)$ and its interpolated version $\tilde{x}(n)$ are related by

$$\tilde{x}(n)=x(n)*\mathbf{i}=\sum_{j=0}^{M-1}i_j x(n-j) \tag{31}$$

and the sparse filter output is given by

$$\hat{y}(n)=\tilde{x}(n)*\mathbf{w}_{\mathrm{s}} \tag{32}$$

where $*$ denotes the convolution operator. An IFIR filter may also be implemented by exchanging the order of the blocks. The results presented here are also valid for such a case, provided corresponding modifications are made.

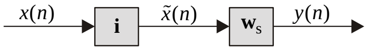

Figure 3 – Block diagram of an IFIR filter.

The factor determining the sparsity in $\mathbf{w}_{\mathrm{s}}$ is termed interpolation factor, which is denoted by L [7]. The sparse filter is obtained by setting to zero $(L-1)$ samples from each L consecutive ones from the N-dimensional model filter $\mathbf{w}=[w_0\ \ w_1\ \ \cdots\ \ w_{N-1}]^{\mathrm{T}}$. Thus, the corresponding N-dimensional sparse vector is

$$\mathbf{w}_{\mathrm{s}}=[w_0\ 0\ \cdots\ 0\ w_L\ 0\ \cdots\ 0\ \ w_{2L}\ \ 0\ \cdots\ 0\ \ w_{(N_{\mathrm{s}}-1)L}\ \ 0\ \cdots\ 0]^{\mathrm{T}} \tag{33}$$

with an input vector given by

$$\tilde{\mathbf{x}}(n)=[\tilde{x}(n)\ \ \tilde{x}(n-1)\ \ \tilde{x}(n-2)\ \ \cdots\ \ \tilde{x}(n-N+1)]^{\mathrm{T}}\,. \tag{34}$$

In (33), N_{s} denotes the number of nonzero coefficients, given by

$$N_{\mathrm{s}}=\left\lfloor\frac{N-1}{L}\right\rfloor+1\,. \tag{35}$$

with $\lfloor\cdot\rfloor$ representing the truncation operation.

Note that an IFIR filter is a particular case of the structures presented in Section 2. Thus, the equivalent coefficient vector for the structure of Fig. 3 is given by

$$\mathbf{w}_{\mathrm{i}}=\mathbf{I}\mathbf{w}_{\mathrm{s}}=\mathbf{W}_{\mathrm{s}}\mathbf{i} \tag{36}$$

where the matrices $\mathbf{I}$ and $\mathbf{W}_{\mathrm{s}}$ are defined in a similar way as in (2) and (3), respectively. The fully adaptive update equations of the IFIR filter are obtained by using the expressions from Sections 3 and 4. Thus, by considering the LMS algorithm, we get

$$\mathbf{i}(n+1)=\mathbf{i}(n)+2\mu_1 e(n)\mathbf{W}_{\mathrm{s}}^{\mathrm{T}}(n)\mathbf{x}_{\mathrm{e}}(n) \tag{37}$$

and

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + 2\mu_2 e(n)\mathbf{I}^T(n)\mathbf{x}_e(n) . \qquad (38)$$

Expressions (37) and (38) are valid for an IFIR structure considering the interpolator in any position, either at the front or end. The implementation of the recursions (37) and (38) requires a considerable computational burden because of the matrix products. However, by considering a slow adaptation condition, such filters can be implemented with less complexity by admitting some simplifications. This procedure is carried out taking into account that the product $\mathbf{W}_s^T(n)\mathbf{x}_e(n)$, given by

$$\mathbf{x}_w(n) = \mathbf{W}_s^T(n)\mathbf{x}_e(n) = [\mathbf{w}_s^T(n)\mathbf{x}_s(n) \;\; \mathbf{w}_s^T(n)\mathbf{x}_s(n-1) \;\; \cdots \;\; \mathbf{w}_s^T(n)\mathbf{x}_s(n-M+1)]^T \qquad (39)$$

can be replaced by its approximated version

$$\mathbf{x}'_w(n) = [\mathbf{w}_s^T(n)\mathbf{x}_s(n) \;\; \mathbf{w}_s^T(n-1)\mathbf{x}_s(n-1) \;\; \cdots \;\; \mathbf{w}_s^T(n-M+1)\mathbf{x}_s(n-M+1)]^T \qquad (40)$$

where $\mathbf{x}_s(n)$ in (39) and (40) is

$$\mathbf{x}_s(n) = \left[x(n) \;\; x(n-1) \;\; \cdots \;\; x(n-N+1)\right]^T . \qquad (41)$$

To obtain the elements from (39), the evaluation of all M inner vector products are required at each iteration n. We can reduce the computational complexity of (39) by using the approximation (40), requiring now the computation of only the first element $\mathbf{w}_s^T(n)\mathbf{x}_s(n)$ at each iteration. With such a reduction the applicability of the cascaded structure is significantly enhanced. The same consideration is also used for computing the product $\mathbf{I}^T(n)\mathbf{x}_e(n)$, resulting in vector $\tilde{\mathbf{x}}(n)$ which is already available from the filtering operation. The simplified update expressions of the FAIFIR-LMS filter are

$$\mathbf{i}(n+1) = \mathbf{i}(n) + 2\mu_1 e(n)\mathbf{x}'_w(n) \qquad (42)$$

and

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + 2\mu_2 e(n)\tilde{\mathbf{x}}(n) . \qquad (43)$$

As before, the same care must be taken to ensure proper operation of the adaptive cascaded structure.

The update expressions using the NLMS algorithm, according to the expressions obtained in Section 4, are

$$\mathbf{i}(n+1) = \mathbf{i}(n) + \frac{\alpha_1}{\left\|\mathbf{W}_s^T(n)\mathbf{x}_e(n)\right\|^2 + \psi_1} e(n)\mathbf{W}_s^T(n)\mathbf{x}_e(n) \qquad (44)$$

and

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + \frac{\alpha_2}{\left\|\mathbf{I}^T(n)\mathbf{x}_e(n)\right\|^2 + \psi_2} e(n)\mathbf{I}^T(n)\mathbf{x}_e(n) . \qquad (45)$$

Then, the corresponding simplified expressions are given by

$$\mathbf{i}(n+1) = \mathbf{i}(n) + \frac{\alpha_1}{\left\|\mathbf{x}_w(n)\right\|^2 + \psi_1} e(n)\mathbf{x}'_w(n) \qquad (46)$$

and

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + \frac{\alpha_2}{\left\|\tilde{\mathbf{x}}(n)\right\|^2 + \psi_2} e(n)\tilde{\mathbf{x}}(n) . \qquad (47)$$

The simplified expressions (42)-(43) and (46)-(47) for the LMS and NLMS algorithms, respectively, are presented in [9]-[11]. In [9], such expressions are obtained by assuming that each filter is time-invariant. This is also the case in the derivations given in [10] and [11], where the NLMS and affine projection (AP) algorithms are also considered. However, the impact of the adopted approximations on the algorithm behavior is not highlighted in these papers.

The use of FAIFIR filters for replacing the standard AFIR filters is only interesting if the simplified structure is adopted. The computational burden required by the standard AFIR filter is much smaller than that required by FAIFIR ones without any simplification. However, it is very important to notice that, if simplified expressions are used, special care with the used step-size value must be taken, avoiding an improper algorithm operation. The simulation results presented in the next section provide more insight on this fact.

6. SIMULATION RESULTS

In this section, considering a system identification problem, some examples are presented aiming to verify the proposed algorithms. The MSE curves of the FAIFIR structures (without and with simplification) are compared with those of the AFIR and AIFIR ones, for both the LMS and NLMS algorithms. Here, the MSE curves of the AFIR structure are shown only for performance comparison purposes. Such a structure does not consider any strategy for reducing the required computational burden. For all simulations the input signal is white with unit variance. We also add to $d(n)$ a measurement noise with a variance $\sigma_v^2 = 10^{-6}$ $(\mathrm{SNR} = 60\,\mathrm{dB})$. Simulations with colored input signals are not shown here since the obtained results are similar to those by using white input signals.

Example 1: In this example, the adaptive structures are used for modeling a plant given by the length-11 vector $\mathbf{w}_1^o =$ [1.00 0.72 0.51 0.37 0.26 0.19 0.14 0.10 0.07 0.05 0.04]^T. The parameters used for the AIFIR and FAIFIR are $L = 2$ and $N_s = 6$. Here, the adaptive algorithm is the LMS one and the step size used is $\mu = \mu_{\max}/5$ (for all filters), where $\mu_{\max}$ is the maximum step size for algorithm convergence (experimentally determined). The obtained values for $\mu_{\max}$ are 0.07 for the AFIR, 0.05 for the AIFIR, 0.06 for the FAIFIR, and 0.03 for the simplified FAIFIR structure. In the case of the FAIFIR structures, the same step size is selected for both algorithms to ensure stability. The MSE curves are obtained from Monte Carlo simulations (average of 100 independents runs), which are shown in Fig. 4. From this figure, we note that the FAIFIR structures present a better steady-state MSE performance than that obtained with the AIFIR one. Observe that the convergence rate of the FAIFIR structure is larger than that of the simplified one when attaining the same steady-state MSE value.

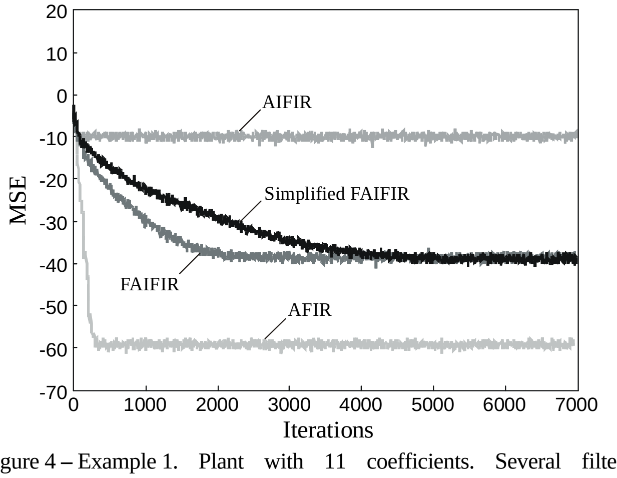

Figure 4 – Example 1. Plant with 11 coefficients. Several filter structures using the LMS algorithm. MSE curves (average of 100 runs).

Example 2: For this example, the plant used is the length-25 vector $\mathbf{w}_2^o =$ [0.01 0.03 0.07 0.12 0.16 0.24 0.28 0.37 0.39 0.49 0.46 0.54 0.49 0.54 0.46 0.48 0.38 0.37 0.28 0.24 0.16 0.12 0.07 0.03 0.01]^T. Again the adaptive algorithm is the LMS one and the choice of the step size is determined as in Example 1. The values obtained for $\mu_{\max}$ are 0.03 for the AFIR filter, 0.02 for the AIFIR, and 0.01 for the simplified FAIFIR structure. The MSE curves are shown in Fig. 5, in which again we verify that the FAIFIR performance is better than the one exhibited by the AIFIR structure. By using the same step size, the FAIFIR performance is similar to that of its simplified version, corroborating the comments made in Section 3.

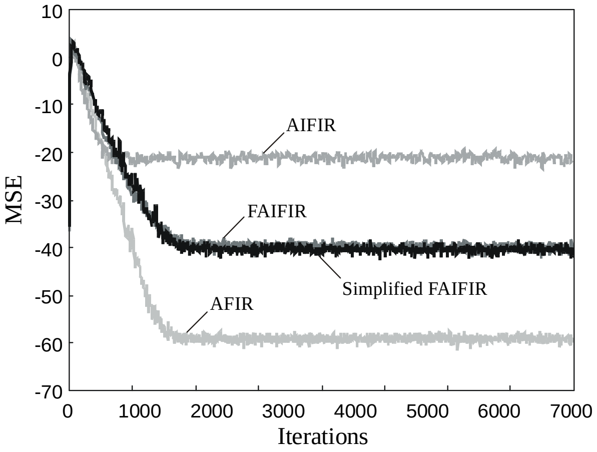

Figure 5 – Example 2. Plant with 25 coefficients. Several filter structures using the LMS algorithm. MSE curves (average of 100 runs).

Example 3: For this example, the plant is the same as in Example 1. Now, the algorithm used is the NLMS one, with $\alpha_1 = \alpha_2 = 0.5$ and $\psi_1 = \psi_2 = 10^{-6}$. The obtained results are shown in Fig. 6. As expected from Example 1, the FAIFIR structure presents a better MSE behavior than the AIFIR one. Now, comparing the MSE performance between the FAIFIR and its simplified version, the difference between them is remarkable. In the latter, we note a higher steady-state MSE as well as the presence of some spikes both due to the values used for the parameters α_1 and α_2.

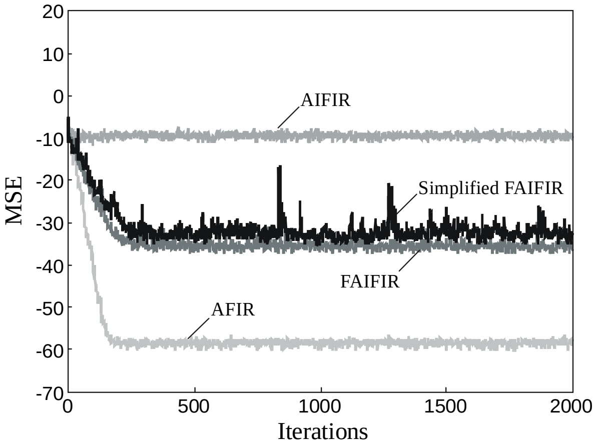

Figure 6 – Example 3. Plant with 11 coefficients. Several filter structures using the NLMS algorithm with $\alpha_1 = \alpha_2 = 0.5$. MSE curves (average of 100 runs).

Example 4: In this example, we have used the same data of Example 3, but now we have reduced the values of the parameters to $\alpha_1 = \alpha_2 = 0.2$. The MSE curves are shown in Fig. 7. Now, we observe almost the same MSE performance for both the FAIFIR structure and its simplified version. This fact demonstrates how critical is the choice of the step-size control parameter values for the NLMS algorithm if a satisfactory MSE performance is required.

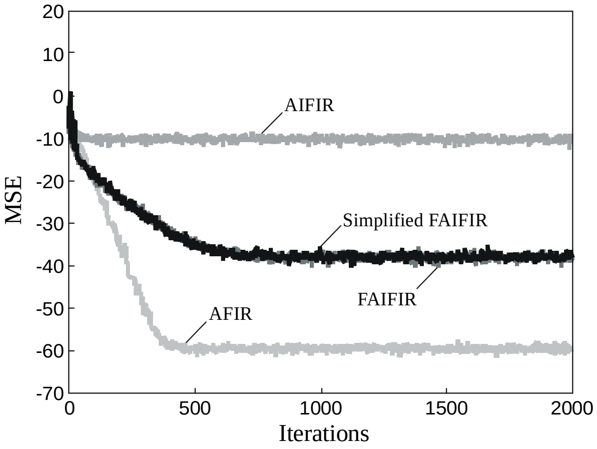

Figure 7 – Example 4. Plant with 11 coefficients. Several filter structures using the NLMS algorithm with $\alpha_1 = \alpha_2 = 0.2$. MSE curves (average of 100 runs).

7. CONCLUDING REMARKS

In this work, general expressions for the LMS and NLMS adaptation of cascaded FIR structures are presented. The application of such expressions to fully adaptive interpolated FIR filter is also discussed, highlighting its advantages, limitations as well as some simplifications aiming to reduce the required computational burden. The presented results bring new insights on adaptive interpolated structures, expanding thus its range of applications.

REFERENCES

[1] Y. Neuvo, C. Y. Dong, and S. K. Mitra, “Interpolated finite impulse response digital filters,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 3, pp. 563-570, Jun. 1984.

[2] A. Abousaada, T. Aboulnasr, and W. Steenaart, “An echo tail canceller based on adaptive interpolated FIR filtering,” *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 39, no. 7, pp. 409-416, Jul. 1992.

[3] O. J. Tobias, R. Seara Jr., and R. Seara, “Echo canceller based on adaptive interpolated FIR filters,” in *Proc. IEEE Int. Telecom. Symp.*, Natal, Brazil, Sept. 2002, pp. 1-5.

[4] S. S. Lin and W. R. Wu, “A low-complexity adaptive echo canceller for xDSL applications,” *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1461-1465, May 2004.

[5] S. Kuo and D. R. Morgan, *Active Noise Control Systems*, John Wiley & Sons, 1996.

[6] L. S. Nielsen and J. Sparso, “Designing asynchronous circuits for low power: An IFIR filter bank for a digital hearing aid,” *Proceedings of the IEEE*, vol. 87, no. 2, pp. 268-281, Feb. 1999.

[7] O. J. Tobias and R. Seara, “Analytical model for the first and second moments of an adaptive interpolated FIR filter using the constrained filtered-X LMS algorithm,” *IEE Proceedings – Vision, Image, Signal Process.*, vol. 148, no. 5, pp. 337-347, Oct. 2001.

[8] R. Seara, J. C. M. Bermudez, and E. Beck, “A new technique for the implementation of adaptive IFIR filters,” in *Proc. Int. Symp. Signals, Systems, Electronics (ISSSE)*, Paris, France, vol. 2, Sept. 1992, pp. 644-647.

[9] M. D. Grosen, “New FIR structures for fixed and adaptive digital filters,” Ph.D. Dissertation, University of California, Santa Barbara, CA, United States, 1987.

[10] R. C. Bilcu, P. Kuosmanen, and K. Egiazarian, “On adaptive interpolated FIR filters,” in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Montreal, Canada, vol. 2, May 2004, pp. 665-668.

[11] R. C. de Lamare and R. Sampaio-Neto, “Adaptive reduced-rank MMSE filtering with interpolated FIR filters and adaptive interpolators,” *IEEE Signal Process. Letters*, vol. 12, no. 3, pp. 177-180, Mar. 2005.

[12] S. Haykin, *Adaptive Filter Theory*, 4 ed., Prentice-Hall, 2002.