

# Filtros FIR Adaptativos em Cascata: Aplicação em Filtros IFIR Adaptativos

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

**Resumo**—Este artigo discute o uso dos algoritmos LMS e NLMS para aplicação em estruturas adaptativas em cascata. Em particular, tais algoritmos são focados para filtros FIR interpolados completamente adaptativos. Contudo, as expressões obtidas são bem gerais, podendo ser estendidas a qualquer tipo de estrutura composta por dois filtros FIR em cascata. Através da abordagem utilizada, são discutidas as principais características do processo adaptativo assim como as limitações das estruturas interpoladas FIR completamente adaptativas. Resultados de simulação são apresentados com o objetivo de ratificar a efetividade das estruturas propostas.

**Palavras-chave**—Algoritmo LMS, Algoritmo NLMS, Filtros adaptativos, Filtros FIR interpolados.

**Abstract**—This paper discusses the use of the LMS and NLMS algorithms for application in cascaded adaptive structures. In particular, such algorithms are aimed at fully adaptive interpolated FIR filters. However, the obtained expressions are general, allowing to be extended for any structure type composed of two FIR filters in cascade. From the present study, the main characteristics of the adaptive process as well as the limitations of fully adaptive interpolated structures are discussed. Simulation results are presented, confirming the effectiveness of the proposed structures.

**Keywords**—LMS algorithm, NLMS algorithm, Adaptive filters, Interpolated FIR filters.

## I. INTRODUÇÃO

Os filtros interpolados de resposta ao impulso finita (IFIR) são estruturas eficientes do ponto de vista computacional, constituindo uma alternativa interessante aos filtros FIR convencionais. O primeiro trabalho nesta área foi apresentado por Neuvo *et al.* [1]. Desde então, muito esforço de pesquisa tem sido realizado, objetivando o uso dos filtros IFIR em aplicações que requerem um grande número de coeficientes. Assim, podem-se mencionar o cancelamento de eco em linha [2]-[4], controle ativo [5] e o processamento de áudio em aparelhos auditivos digitais [6], dentre outras. A idéia por trás dos filtros IFIR é usar um filtro FIR esparso (com número reduzido de coeficientes) em cascata com um filtro interpolador, cuja função é recriar os coeficientes ausentes no filtro esparso.

Eduardo L. O. Batista e Rui Seara, LINSE – Laboratório de Circuitos e Processamento de Sinais, Depto. de Eng. Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, E-mails: {dudu, seara}@linse.ufsc.br.

Orlando J. Tobias, LINSE/UFSC e Departamento de Engenharia Elétrica e Telecomunicações, Universidade Regional de Blumenau, SC, E-mail: tobias.oj@ieee.org.

Este trabalho foi parcialmente financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

A versão adaptativa de um filtro IFIR (AIFIR) representa uma alternativa interessante para a implementação de filtros FIR adaptativos (AFIR) com alta demanda de coeficientes [2]. Um filtro AIFIR é usualmente implementado adaptando apenas os coeficientes do filtro esparso, enquanto que o interpolador é mantido fixo [7]. A posição dos filtros nesta estrutura em cascata pode ser trocada, assim o interpolador pode ser colocado antes ou depois do filtro adaptativo. A posição do interpolador resulta em diferentes processos de adaptação, devido à natureza variante no tempo da estrutura [7], [8]. Apesar disso, o mesmo desempenho de regime permanente é obtido em ambas as implementações.

A redução de complexidade computacional em uma estrutura AIFIR é obtida ao custo de um maior valor de regime permanente do erro quadrático médio (EQM), quando comparado ao obtido por uma estrutura AFIR convencional. Na maioria dos casos, tal degradação de desempenho é consequência da utilização de um interpolador fixo, produzindo uma inadequada recriação dos coeficientes. Uma solução para tal problema é o uso de uma estrutura IFIR inteiramente interpolada (FAIFIR), incorporando um interpolador adaptativo. Tal classe de estrutura foi originalmente proposta em [9]. Em [10], uma estrutura FAIFIR também é introduzida de forma heurística. Recentemente em [11], uma outra forma de tal estrutura também é discutida.

Este trabalho apresenta a derivação dos algoritmos LMS e NLMS para adaptar os coeficientes de filtros FIR em cascata e sua aplicação às estruturas FAIFIR. A abordagem apresentada é generalizada, permitindo sua aplicação para qualquer estrutura composta por uma cascata de dois filtros FIR adaptativos. Particularmente, as expressões obtidas são aplicáveis às estruturas FAIFIR independentemente da posição do filtro esparso e do interpolador. Além disso, tais expressões permitem uma melhor observação das características do processo adaptativo e também das aproximações realizadas em [9]-[11].

Este trabalho está organizado da seguinte forma. Na Seção II, a descrição matemática generalizada da estrutura FIR em cascata é apresentada. A Seção III apresenta o desenvolvimento do algoritmo LMS para tal estrutura. Na Seção IV, o algoritmo NLMS é considerado. A teoria apresentada, aplicada ao filtro FAIFIR, é discutida na Seção V. Na Seção VI, alguns resultados de simulação são apresentados, verificando o desempenho da estrutura FAIFIR. Finalmente, as conclusões do trabalho são apresentadas na Seção VII.

## II. DESCRIÇÃO MATEMÁTICA DE FILTROS FIR EM CASCATA

A Fig. 1 ilustra o diagrama de blocos de uma estrutura de dois filtros FIR em cascata. Nesta figura,  $\mathbf{g}=[g_0 \ g_1 \ \dots \ g_{M-1}]^T$  e  $\mathbf{h}=[h_0 \ h_1 \ \dots \ h_{N-1}]^T$  representam os filtros FIR de entrada e de

saída, respectivamente. A variável  $x(n)$  denota o sinal de entrada,  $\hat{y}(n)$  representa um sinal intermediário e  $y(n)$  é o sinal de saída da estrutura em cascata. O filtro equivalente da estrutura em cascata é representado por  $\mathbf{w}_{\text{eq}}$ , obtido como

$$\mathbf{w}_{\text{eq}} = \mathbf{g} * \mathbf{h}. \quad (1)$$

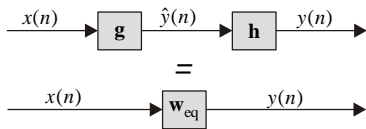


Fig. 1. Diagrama de blocos de uma estrutura FIR em cascata e seu filtro equivalente.

A partir de (1), nota-se que o vetor de coeficientes equivalente é obtido através de uma operação de convolução. Para facilitar o tratamento matemático, tal operação pode ser escrita como um produto de matrizes [9]. Assim, as seguintes matrizes são definidas:

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

com dimensão  $(N + M - 1) \times N$ , e

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

com dimensão  $(N + M - 1) \times M$ . Então, (1) pode ser reescrita como

$$\mathbf{w}_{\text{eq}} = \mathbf{G}\mathbf{h} = \mathbf{H}\mathbf{g}. \quad (4)$$

Devido à operação de convolução presente em (1) e às dimensões das matrizes de (4), observa-se que o filtro equivalente possui um tamanho de memória  $N + M - 1$ . Agora, definindo o vetor de entrada estendido, dado por

$$\mathbf{x}_e(n) = [x(n) \ x(n-1) \ \cdots \ x(n-N-M+2)]^T \quad (5)$$

a relação de entrada e saída da estrutura em cascata pode ser escrita em termos de um produto matricial, resultando em

$$y(n) = \mathbf{w}_{\text{eq}}^T \mathbf{x}_e(n) = \mathbf{h}^T \mathbf{G}^T \mathbf{x}_e(n) = \mathbf{g}^T \mathbf{H}^T \mathbf{x}_e(n). \quad (6)$$

### III. ALGORITMO LMS PARA A ESTRUTURA EM CASCATA

Nesta seção, a versão adaptativa da estrutura de dois filtros em cascata é desenvolvida. Com esse objetivo, os coeficientes de ambos os filtros são adaptados utilizando o algoritmo LMS. A Fig. 2 mostra o diagrama de blocos da estrutura em questão, onde  $\mathbf{g}(n)$  e  $\mathbf{h}(n)$  representam as versões variantes no tempo dos blocos  $\mathbf{g}$  e  $\mathbf{h}$  da Fig. 1,  $d(n)$  denota o sinal a ser estimado (sinal desejado) e  $e(n)$  representa o sinal de erro, dado por

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

Os demais sinais representados nessa figura são os mesmos da Fig. 1.

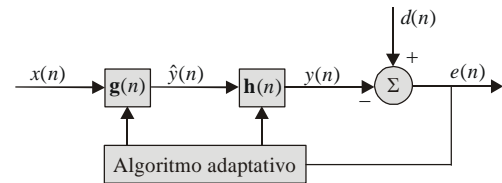


Fig. 2. Diagrama de blocos da estrutura FIR em cascata adaptativa.

Substituindo (6) em (7) e considerando que agora os coeficientes são variantes no tempo em ambos os filtros FIR, obtém-se

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

onde  $\mathbf{G}(n)$  e  $\mathbf{H}(n)$  são as versões variantes no tempo de (2) e (3), respectivamente. Definindo, agora, uma função-custo baseada no erro instantâneo, tem-se

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

Assim, considerando o esquema da Fig. 2, os coeficientes dos filtros são adaptados usando o gradiente da função custo de (9), conforme descrito em [12]. As expressões requeridas para realizar tal adaptação são obtidas nas próximas seções.

#### A. Adaptação dos Coeficientes do Filtro de Entrada

Considerando o filtro de entrada  $\mathbf{g}(n)$ , a adaptação é realizada conforme a regra do gradiente. Assim,

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

com  $\mu_1$  representando o passo de adaptação do filtro de entrada. Aplicando a regra da cadeia, o vetor gradiente em (10) é escrito como

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

Dessa maneira, a partir de (8), os termos de (11) são dados por

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

e

$$\frac{\partial e(n)}{\partial \mathbf{g}(n)} = -\mathbf{H}^T(n) \mathbf{x}_e(n). \quad (13)$$

Assim, a equação de adaptação para os coeficientes do filtro de entrada, usando o algoritmo LMS, é dada por

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

### B. Adaptação dos Coeficientes do Filtro de Saída

Considerando o filtro de saída e usando também a regra do gradiente

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

onde  $\mu_2$  é o passo de adaptação do filtro de saída, de forma análoga a (11), temos

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

resultando em (12) e

$$\frac{\partial e(n)}{\partial \mathbf{h}(n)} = -\mathbf{G}^T(n) \mathbf{x}_e(n). \quad (17)$$

Finalmente, a equação de adaptação para os coeficientes do filtro de saída é dada por

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

### C. Particularidades dos Filtros em Cascata Adaptativos

Algumas questões importantes surgem a partir da análise do processo de adaptação dos filtros em cascata:

- i) Inicialização dos Coeficientes. Conforme mencionado em [9], se ambos os vetores de coeficientes  $\mathbf{g}(n)$  e  $\mathbf{h}(n)$  são inicializados com valores iguais a zero, o que é comumente realizado, o processo adaptativo não ocorre. Tal característica pode ser facilmente verificada a partir de (14) e (18). Tal problema é contornado inicializando os vetores  $\mathbf{g}(n)$  e  $\mathbf{h}(n)$  (ou pelo menos um deles) com valores diferentes de zero.
- ii) Complexidade computacional. Novamente, a partir de (14) e (18) observa-se que o cálculo de dois produtos de matrizes,  $\mathbf{H}^T(n) \mathbf{x}_e(n)$  e  $\mathbf{G}^T(n) \mathbf{x}_e(n)$ , é necessário a cada iteração, requerendo um alto custo computacional. Entretanto, considerando uma condição de adaptação lenta e as estruturas particulares das matrizes  $\mathbf{G}(n)$  e  $\mathbf{H}(n)$ , tais produtos matriciais podem ser implementados com apenas um produto interno vetorial por amostra, reduzindo assim a carga computacional. Tal procedimento é discutido nas próximas seções.
- iii) Estabilidade do algoritmo. Devido à adaptação simultânea dos dois filtros e à presença de dois passos de adaptação, é necessário uma escolha apropriada para seus valores, garantindo a estabilidade do algoritmo. No entanto, a obtenção de limites de estabilidade analíticos para esse caso é bastante difícil. Uma solução alternativa é usar um valor conservativo do passo de adaptação para ambos os filtros, o qual é obtido experimentalmente. Adotando essa estratégia, bons resultados práticos são obtidos, conforme será verificado na Seção VI.

## IV. ALGORITMO NLMS EM CASCATA

Nesta seção, as expressões que permitem a adaptação da estrutura em cascata usando o algoritmo NLMS são desenvolvidas. De maneira similar ao que foi realizado em [12], a equação de adaptação dos coeficientes é obtida minimizando a norma euclidiana de

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

sujeita à seguinte restrição:

$$\mathbf{g}^T(n+1) \mathbf{H}^T(n) \mathbf{x}_e(n) = d(n). \quad (20)$$

Note que a expressão de restrição de (20) é ligeiramente diferente da apresentada em [12], devido às características particulares da estrutura em cascata. Aplicando o método dos multiplicadores de Lagrange [12] a (19) e (20), a função-custo é agora dada por

$$J_g(n) = \|\delta \mathbf{g}(n+1)\|^2 + \lambda_1 \left[ d(n) - \mathbf{g}^T(n+1) \mathbf{H}^T(n) \mathbf{x}_e(n) \right] \quad (21)$$

onde  $\lambda_1$  é um multiplicador de Lagrange com valor real. Diferenciando (21) em função de  $\mathbf{g}(n+1)$ , obtém-se

$$\frac{\partial J_g(n)}{\partial \mathbf{g}(n+1)} = 2[\mathbf{g}(n+1) - \mathbf{g}(n)] - \lambda_1 \mathbf{H}^T(n) \mathbf{x}_e(n). \quad (22)$$

Fazendo (22) igual a zero, temos

$$\mathbf{g}(n+1) = \mathbf{g}(n) + \frac{1}{2} \lambda_1 \mathbf{H}^T(n) \mathbf{x}_e(n). \quad (23)$$

Substituindo (23) em (20) resulta em

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

Agora, resolvendo (24) para  $\lambda_1$ , obtém-se

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

Finalmente, substituindo (25) em (23), adicionando as constantes positivas  $\alpha_1$  e  $\psi_1$  para controlar o processo adaptativo e prevenir a divisão por zero [12], respectivamente, a equação de adaptação dos coeficientes do filtro de entrada é dada por

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

De forma similar a (19) e (20), para o filtro de saída, é necessário minimizar

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

submetida à restrição

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

Novamente, aplicando o método dos multiplicadores de Lagrange [12], obtém-se

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

Note que (29) tem a mesma forma de (21). Assim, através de um procedimento similar ao anterior, a seguinte equação de adaptação é obtida:

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

onde  $\alpha_2$  e  $\psi_2$  são constantes positivas com a mesma finalidade que  $\alpha_1$  e  $\psi_1$ .

Com relação às questões de implementação do algoritmo NLMS em uma estrutura em cascata, as considerações (i) e (ii) da Seção III-C são também aplicáveis para este caso.

### V. FILTROS IFIR INTEIRAMENTE ADAPTATIVOS

Nesta seção, as equações para adaptação dos coeficientes da estrutura FAIFIR são derivadas, partindo das expressões de adaptação da estrutura em cascata previamente apresentadas. A Fig. 3 ilustra o diagrama de blocos de uma estrutura IFIR. Nessa figura,  $\mathbf{w}_s$  denota um filtro esparsos e  $\mathbf{i} = [i_0 \ i_1 \ \dots \ i_{M-1}]^T$  representa o filtro interpolador. O sinal de entrada  $x(n)$  e sua versão interpolada  $\tilde{x}(n)$  são relacionados através de

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

e o sinal de saída do filtro esparsos é dado por

$$\hat{y}(n) = \tilde{x}(n) * \mathbf{w}_s \quad (32)$$

onde “\*” denota a operação de convolução. Um filtro IFIR pode também ser implementado invertendo as posições dos blocos, ou seja, com o filtro interpolador na saída. Os resultados aqui apresentados, com pequenas modificações, são válidos para ambos os casos.

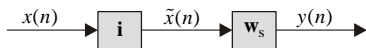


Fig. 3. Diagrama de blocos de um filtro IFIR.

O fator que determina a esparsidade de  $\mathbf{w}_s$  é denominado fator de interpolação, denotado por  $L$  [7]. O filtro esparsos é obtido zerando  $(L-1)$  coeficientes de cada  $L$  consecutivos do vetor de coeficientes do filtro convencional  $\mathbf{w} = [w_0 \ w_1 \ \dots \ w_{N-1}]^T$  (tamanho de memória  $N$ ). Assim, o vetor esparsos correspondente tem dimensão  $N$ , dado por

$$\mathbf{w}_s = [w_0 \ 0 \ \dots \ 0 \ w_L \ 0 \ \dots \ 0 \ w_{2L} \ 0 \ \dots \ 0 \ w_{(N_s-1)L} \ 0 \ \dots \ 0]^T. \quad (33)$$

Em (33),  $N_s$  denota o número de coeficientes diferentes de zero, obtido de

$$N_s = \left\lfloor \frac{N-1}{L} \right\rfloor + 1 \quad (34)$$

onde  $\lfloor \cdot \rfloor$  representa a operação de truncamento. O vetor de entrada do filtro esparsos é composto por amostras do sinal de entrada interpolado, resultando em

$$\tilde{\mathbf{x}}(n) = [\tilde{x}(n) \ \tilde{x}(n-1) \ \tilde{x}(n-2) \ \dots \ \tilde{x}(n-N+1)]^T. \quad (35)$$

Note que o filtro IFIR é um caso particular das estruturas apresentadas na Seção II. Assim, o vetor de coeficientes equivalente para a estrutura da Fig. 3 é dado por

$$\mathbf{w}_i = \mathbf{I} \mathbf{w}_s = \mathbf{W}_s \mathbf{i}. \quad (36)$$

As matrizes  $\mathbf{I}$  e  $\mathbf{W}_s$  são definidas de forma similar às aquelas dadas em (2) e (3), respectivamente. As equações para adaptação completa do filtro IFIR são obtidas usando as expressões das Seções 3 e 4. Assim, considerando o algoritmo LMS, obtém-se

$$\mathbf{i}(n+1) = \mathbf{i}(n) + 2\mu_1 e(n) \mathbf{W}_s^T(n) \mathbf{x}_e(n) \quad (37)$$

e

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

É importante ressaltar, ainda, que as expressões (37) e (38) são válidas para a estrutura IFIR da Fig. 3; também para sua implementação com o interpolador na saída. A implementação de (37) e (38) implica em uma considerável carga computacional devido aos produtos de matrizes envolvidos. Entretanto, considerando uma condição de adaptação lenta, tais filtros podem ser implementados com menor complexidade, admitindo algumas simplificações. Tal procedimento é realizado levando-se em conta que o produto  $\mathbf{W}_s^T(n) \mathbf{x}_e(n)$ , dado por

$$\mathbf{x}'_w(n) = \mathbf{W}_s^T(n) \mathbf{x}_e(n) = [\mathbf{w}_s^T(n) \mathbf{x}_s(n) \ \mathbf{w}_s^T(n) \mathbf{x}_s(n-1) \ \dots \ \mathbf{w}_s^T(n) \mathbf{x}_s(n-M+1)]^T \quad (39)$$

pode ser aproximado por

$$\mathbf{x}'_w(n) \approx [\mathbf{w}_s^T(n) \mathbf{x}_s(n) \ \mathbf{w}_s^T(n-1) \mathbf{x}_s(n-1) \ \dots \ \mathbf{w}_s^T(n-M+1) \mathbf{x}_s(n-M+1)]^T \quad (40)$$

onde  $\mathbf{x}_s(n)$ , de (39) e (40), é dado por

$$\mathbf{x}_s(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T. \quad (41)$$

Para se obter os elementos de (39), o cálculo de todos os  $M$  produtos internos vetoriais são necessários a cada iteração. Entretanto, utilizando a aproximação (40), é necessário agora apenas calcular o primeiro elemento  $\mathbf{w}_s^T(n) \mathbf{x}_s(n)$  a cada iteração. Tal redução de complexidade contribui sobremaneira para a aplicação prática da estrutura em cascata. Uma aproximação similar é realizada para calcular o produto  $\mathbf{I}^T(n) \mathbf{x}_e(n)$  de (38), resultando no vetor  $\tilde{\mathbf{x}}(n)$ , ou seja,  $\tilde{\mathbf{x}}(n) \approx \mathbf{I}^T(n) \mathbf{x}_e(n)$ . Assim, as equações de adaptação simplificadas para o filtro FAIFIR-LMS são

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

e

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

As expressões para adaptação do filtro FAIFIR usando o algoritmo NLMS, de acordo com os resultados da Seção IV, são

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

e

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

Adotando as mesmas aproximações utilizadas para obter (42) e (43), as expressões simplificadas para adaptação usando o algoritmo NLMS são dadas por

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

e

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

A carga computacional requerida para implementação do filtro AFIR convencional é menor do que aquela requerida para o filtro FAIFIR sem simplificação. Tal diferença se deve à necessidade de realizar cálculos adicionais (na versão sem simplificação)

decorrentes dos produtos matriciais  $\mathbf{W}_s^T(n)\mathbf{x}_e(n)$  e  $\mathbf{I}^T(n)\mathbf{x}_e(n)$  para a adaptação do filtro FAIFIR. Assim, o uso de uma estrutura FAIFIR para substituir os filtros AFIR convencionais torna-se interessante somente se as expressões simplificadas são utilizadas. Tais expressões, dadas por (42)-(43) e (46)-(47) para os algoritmos LMS e NLMS, respectivamente, são as mesmas apresentadas em [9]-[11]. No entanto, nessas referências, os efeitos das hipóteses simplificativas consideradas não estão destacados nem avaliados. É importante observar que, quando tais simplificações são adotadas, um cuidado especial deve ser tomado para a escolha dos passos de adaptação e fatores de controle, de forma a evitar a operação imprópria do algoritmo. Os resultados de simulação apresentados na próxima seção permitem verificar melhor tais efeitos sobre o comportamento dos algoritmos aqui avaliados.

## VI. RESULTADOS DE SIMULAÇÃO

Nesta seção, alguns exemplos considerando um problema de identificação de sistemas são apresentados com o objetivo de verificar o desempenho dos algoritmos descritos neste trabalho. As curvas de EQM das estruturas FAIFIR (com e sem simplificação) são comparadas com as curvas obtidas com os filtros AFIR e AIFIR. Para todos os casos, o sinal de entrada é um ruído branco com variância unitária. Também, é adicionado a  $d(n)$  um ruído de medição com variância  $\sigma_v^2 = 10^{-6}$  (SNR = 60dB) para os exemplos considerando o algoritmo LMS e variância  $\sigma_v^2 = 10^{-4}$  (SNR = 40dB) para os exemplos envolvendo o algoritmo NLMS. Simulações considerando um sinal de entrada colorido não são apresentadas neste trabalho, visto que os resultados são similares aos obtidos a partir do ruído branco. O fator de interpolação utilizado para os filtros AIFIR e FAIFIR é  $L=2$  em todos os exemplos. O interpolador fixo do filtro AIFIR é dado por  $\mathbf{i}=[0,5 \ 1,0 \ 0,5]^T$  e o interpolador variável dos filtros FAIFIR é inicializado com esse vetor de coeficientes.

**Exemplo 1:** Neste exemplo, os filtros adaptativos são utilizados para modelar uma planta com 11 coeficientes dada por  $\mathbf{w}_1^0 = [1,00 \ 0,72 \ 0,51 \ 0,37 \ 0,26 \ 0,19 \ 0,14 \ 0,10 \ 0,07 \ 0,05 \ 0,04]^T$ . O algoritmo utilizado é o LMS e o mesmo valor do passo de adaptação  $\mu = \mu_{\max}/5$  é usado para todos os filtros, onde  $\mu_{\max}$  é o valor máximo do passo de adaptação para o qual o algoritmo converge (determinado experimentalmente). Os valores obtidos neste exemplo para  $\mu_{\max}$  são: 0,07 para o filtro AFIR, 0,05 para o AIFIR, 0,06 para o FAIFIR, e 0,03 para a estrutura FAIFIR simplificada. No caso das estruturas FAIFIR, o mesmo passo de adaptação é utilizado para o filtro esparsos e para o interpolador, garantindo assim a estabilidade do algoritmo. As curvas de EQM, obtidas a partir de simulações de Monte Carlo (média de 100 realizações independentes), são apresentadas na Fig. 4. A partir dessa figura, observa-se que as estruturas FAIFIR apresentam melhor desempenho, em termos do EQM de regime permanente, daquele obtido com o filtro AIFIR. Além disso, a convergência do filtro FAIFIR sem simplificações é mais rápida daquela obtida com o FAIFIR simplificado, com o mesmo valor de regime permanente.

**Exemplo 2:** Para este exemplo, a planta tem 25 coeficientes dados por  $\mathbf{w}_2^0 = [0,01 \ 0,03 \ 0,07 \ 0,12 \ 0,16 \ 0,24 \ 0,28 \ 0,37 \ 0,39 \ 0,49 \ 0,46 \ 0,54 \ 0,38 \ 0,37 \ 0,28 \ 0,24 \ 0,16 \ 0,12 \ 0,07 \ 0,03 \ 0,01]^T$ .

Novamente, o algoritmo utilizado é o LMS com valores para o passo de adaptação determinados de maneira similar ao Exemplo 1. Os valores obtidos para  $\mu_{\max}$  são: 0,03 para o filtro AFIR, 0,02 para o AIFIR e 0,01 para os filtros FAIFIR convencional e simplificado. As curvas de EQM são apresentadas na Fig. 5, observando-se novamente o melhor desempenho da estrutura FAIFIR em relação à AIFIR. Também, nota-se que o desempenho da estrutura FAIFIR é similar ao da versão simplificada, decorrente do uso de valores menores para o passo de adaptação.

**Exemplo 3:** Para este exemplo, a planta usada é um filtro FIR cujos coeficientes são dados por  $w_3^0(k) = (1/2)\text{sinc}(k/30)$  com  $k = 0,1, \dots, 100$  (101 coeficientes). Agora, o algoritmo utilizado é o NLMS com  $\alpha_1 = \alpha_2 = 0,2$  e  $\psi_1 = \psi_2 = 10^{-4}$ . Para o caso do filtro FAIFIR simplificado, o uso de fatores de controle  $\alpha_1 = \alpha_2 = 0,2$  tornou o algoritmo instável. Assim, optou-se por reduzir tais fatores para  $\alpha_1 = \alpha_2 = 0,1$  (somente no caso do FAIFIR simplificado). Os resultados obtidos são apresentados na Fig. 6. Conforme esperado, a estrutura FAIFIR e sua versão simplificada obtiveram um desempenho melhor do que o obtido com o AIFIR. No entanto, o desempenho da estrutura FAIFIR convencional é diferente da sua versão simplificada. Observa-se que a simplificação leva, nesse caso, à instabilidade do algoritmo quando usado  $\alpha_1 = \alpha_2 = 0,2$  e, conseqüentemente, à necessidade de considerar valores menores para os fatores de controle. Além disso, o filtro FAIFIR simplificado apresenta um maior valor de regime permanente como também alguns picos na curva de EQM mesmo com o uso de valores menores para os fatores de controle.

**Exemplo 4:** Neste exemplo, os mesmos parâmetros do Exemplo 3 são considerados, com exceção do valor dos fatores de controle utilizados para adaptação do filtro FAIFIR simplificado. Tais fatores são agora reduzidos para  $\alpha_1 = \alpha_2 = 0,06$ . As curvas de EQM obtidas são apresentadas na Fig. 7. Nota-se, para esse caso, que o filtro FAIFIR simplificado atinge praticamente o mesmo valor de EQM de regime permanente obtido com sua implementação completa. Assim, verifica-se que uma escolha adequada para o fator de controle do NLMS é crítica para se obter um desempenho satisfatório em aplicações envolvendo as diferentes implementações do filtro FAIFIR simplificado.

## VII. CONCLUSÕES

Neste trabalho, expressões generalizadas para a adaptação da estrutura FIR em cascata através dos algoritmos LMS e NLMS são apresentadas. A aplicação de tais expressões para o filtro FIR interpolado inteiramente adaptativo também é discutida, ressaltando suas vantagens e limitações. Também, algumas simplificações objetivando reduzir a complexidade computacional são realizadas. Os resultados deste trabalho permitem um melhor entendimento das estruturas interpoladas adaptativas, expandindo assim sua gama de aplicações.

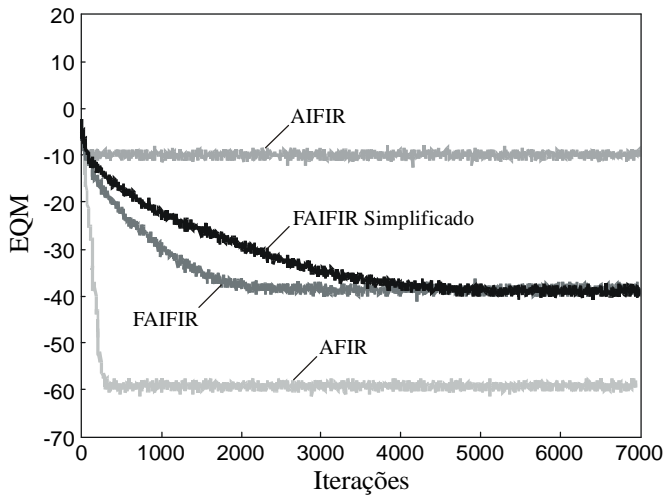


Fig. 4. Exemplo 1. Planta com 11 coeficientes. Diversas estruturas de filtragem usando o algoritmo LMS. Curvas de EQM (média de 100 realizações).

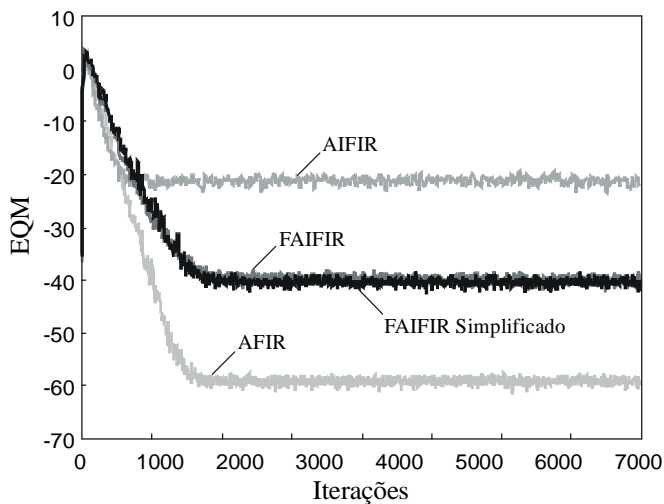


Fig. 5. Exemplo 2. Planta com 25 coeficientes. Diversas estruturas de filtragem usando o algoritmo LMS. Curvas de EQM (média de 100 realizações).

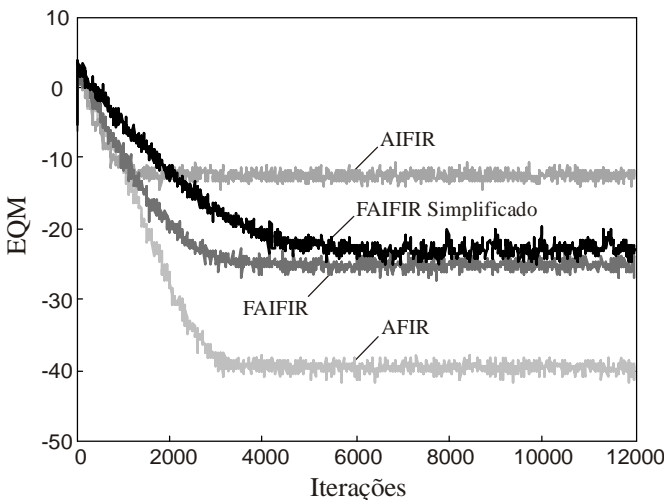


Fig. 6. Exemplo 3. Planta com 101 coeficientes. Diversas estruturas de filtragem usando o algoritmo NLMS ( $\alpha_1 = \alpha_2 = 0,1$  para o filtro FAIFIR simplificado e  $\alpha_1 = \alpha_2 = 0,2$  para os demais filtros). Curvas de EQM (média de 100 realizações).

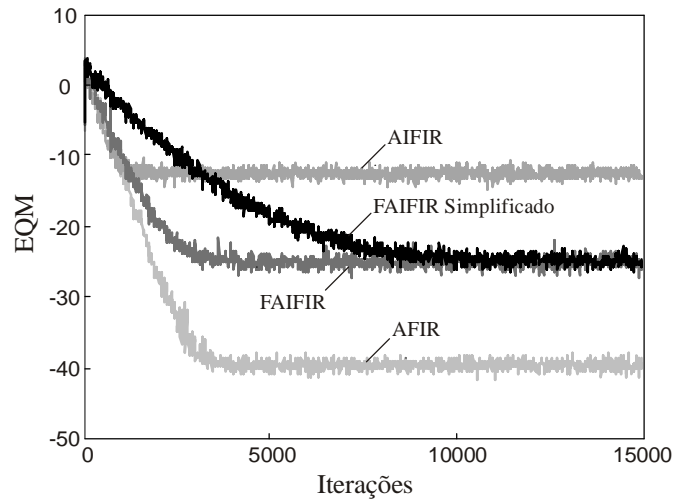


Fig. 7. Exemplo 4. Planta com 101 coeficientes. Diversas estruturas de filtragem usando o algoritmo NLMS ( $\alpha_1 = \alpha_2 = 0,06$  para o filtro FAIFIR simplificado e  $\alpha_1 = \alpha_2 = 0,2$  para os demais filtros). Curvas de EQM (média de 100 realizações).

#### REFERÊNCIAS

- [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. Aboussada, T. Aboulnasr, and W. Steenaert, "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.