

Filtros IFIR Inteiramente Adaptativos sem Efeito de Borda

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

Resumo—Este trabalho apresenta um procedimento para implementação de filtros adaptativos FIR interpolados inteiramente adaptados e com remoção do efeito de borda. A abordagem proposta permite uma redução do erro quadrático médio em regime permanente, eliminando as principais fontes de degradação de desempenho dos filtros adaptativos FIR interpolados. Particularmente, em situações na qual o efeito de borda é significativo e/ou o uso de um filtro interpolador fixo não é tão eficiente, a melhoria de desempenho alcançada com o novo esquema é notável. Além do mais, o custo computacional para implementação de tal procedimento de filtragem é mínimo. Resultados de simulação confirmam a efetividade da abordagem proposta.

Palavras-chave—Filtros adaptativos, filtros FIR interpolados, algoritmo LMS, algoritmo NLMS.

Abstract—This paper presents a procedure for implementing fully adaptive interpolated FIR filters with removed border effect. The proposed approach allows reducing the steady-state mean-square error by eliminating the main sources of performance degradation from the adaptive interpolated FIR filters. Particularly, in situations in which the border effect is significant and/or the fixed interpolator filter is not so effective, the performance enhancement achieved by this new scheme is noticeable. In addition, the computational effort needed for implementing such a procedure is very small. Simulation results confirm the effectiveness of the proposed approach.

Keywords—Adaptive filters, interpolated FIR filters, LMS algorithm, NLMS algorithm.

I. INTRODUÇÃO

Um filtro FIR interpolado (IFIR) pode ser visto como uma implementação eficiente de um filtro FIR convencional [1]. Mais recentemente, as estruturas IFIR vêm sendo também utilizadas com sucesso em filtragem adaptativa [2], [3]. A idéia por trás dos filtros IFIR é implementar um filtro FIR através de uma estrutura em cascata composta por um filtro esparso, apresentando uma quantidade reduzida de coeficientes e um filtro interpolador, cuja função é recriar de forma aproximada os coeficientes ausentes no filtro esparso. Atualmente, diferentes versões de filtros adaptativos IFIR estão sendo consideradas para diversas aplicações, a saber, cancelamento de eco em linhas telefônicas [2]-[4], controle ativo de ruído e vibrações [5], processamento de áudio em aparelhos auditivos digitais [6], dentre outras.

No contexto adaptativo, os filtros IFIR são interessantes devido ao número reduzido de coeficientes para ser adaptado quando comparado aos filtros FIR adaptativos convencionais [7], [8]. No entanto, tal redução é quase sempre acompanhada de um maior valor

do erro quadrático médio (EQM) em regime permanente. Uma das causas de tal degradação advém do uso de um filtro interpolador fixo, resultando em uma recriação inadequada dos coeficientes decorrente do processo de interpolação. A solução para esse problema é o uso de estruturas IFIR inteiramente adaptativas [9]-[11], nas quais o interpolador também é adaptado e, com isso, o processo de recriação dos coeficientes passa a ser realizado de forma mais eficiente. Outra importante causa de degradação de desempenho em filtros adaptativos IFIR é o efeito de borda, originado da obtenção da estrutura IFIR equivalente [12]. Um procedimento proposto em [13] permite a remoção de tal efeito em filtros IFIR adaptativos com interpolador fixo, resultando em um considerável ganho de desempenho. Outras causas de perda de desempenho são decorrentes da característica esparsa e do posto reduzido [9], [11] do filtro IFIR. No entanto, como tal característica é inerente à estrutura interpolada, não há como mitigar seu impacto. Neste trabalho, um procedimento para a implementação de filtros IFIR inteiramente adaptativos e sem o efeito de borda é apresentado. Este artigo está organizado como segue. Na Seção II, os filtros IFIR são descritos sinteticamente, com ênfase à sua estrutura equivalente e às implementações, seja considerando o interpolador na entrada ou na saída do sistema adaptativo. A Seção III discute o procedimento de remoção do efeito de borda em estruturas IFIR. Na Seção IV, a implementação dos filtros IFIR inteiramente adaptativos e sem o efeito de borda é discutida como também as equações de adaptação dos coeficientes através dos algoritmos *least-mean-square* (LMS) e LMS normalizado (*normalized LMS* - NLMS) [14] são derivadas. A Seção V apresenta resultados de simulação e, finalmente, na Seção VI, são considerados os comentários e conclusões do trabalho em questão.

II. FILTROS IFIR

Um filtro interpolado de resposta ao impulso finita (IFIR) é uma realização com número reduzido de coeficientes de um filtro FIR convencional. O diagrama de blocos de tal estrutura é ilustrado na Fig. 1.

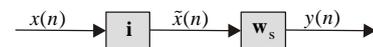


Fig. 1. Diagrama de blocos de um filtro IFIR.

Nesta figura, $x(n)$ e $y(n)$ representam, respectivamente, os sinais de entrada e saída do filtro IFIR. O filtro interpolador, indicado pelo bloco \mathbf{i} , consiste de um filtro FIR com tamanho de memória M e vetor de coeficientes dado por

$$\mathbf{i} = [i_0 \ i_1 \ \dots \ i_{M-1}]^T. \quad (1)$$

A saída $\tilde{x}(n)$ do interpolador está relacionada ao sinal de entrada por

$$\tilde{x}(n) = x(n) * \mathbf{i} = \sum_{j=0}^{M-1} i_j x(n-j) \quad (2)$$

onde “*” denota a operação de convolução. O bloco \mathbf{w}_s da Fig. 1 representa um filtro esparso com tamanho de memória N . O vetor de coeficientes é obtido fixando em zero $L-1$ de cada L coeficientes de um filtro FIR convencional (L denota o fator de esparsidade ou interpolação). Assim, a partir do vetor de coeficientes do filtro FIR convencional

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 Depto. Eng. Elétrica e Telecom., 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).

$$\mathbf{w} = [w_0 \ w_1 \ w_2 \ \dots \ w_{N-1}]^T \quad (3)$$

obtm-se o seguinte vetor esparso de coeficientes:

$$\mathbf{w}_s = [w_0 \ 0 \ \dots \ w_L \ 0 \ \dots \ w_{2L} \ 0 \ \dots \ w_{(N_s-1)L} \ 0 \ \dots \ 0]^T. \quad (4)$$

O número de coeficientes diferentes de zero em (4) é N_s , obtido por

$$N_s = \left\lfloor \frac{N-1}{L} \right\rfloor + 1 \quad (5)$$

com $\lfloor \cdot \rfloor$ representando a operação de truncamento. O sinal de saída do filtro IFIR é então dado por

$$y(n) = \tilde{x}(n) * \mathbf{w}_s \quad (6)$$

ou, ainda, na forma vetorial, através de

$$y(n) = \tilde{\mathbf{x}}^T(n) \mathbf{w}_s \quad (7)$$

com o vetor de entrada interpolado dado por

$$\tilde{\mathbf{x}}(n) = \{\tilde{x}(n) \ \tilde{x}(n-1) \ \tilde{x}(n-2) \ \dots \ \tilde{x}(n-N+1)\}^T. \quad (8)$$

Uma outra maneira de implementar um filtro IFIR está ilustrada na Fig. 2. Nessa figura, temos os mesmos blocos da Fig. 1 apenas com a inversão da posição dos filtros e agora com $\hat{x}(n)$ representando o sinal de entrada filtrado pelo filtro esparso. Para esse caso, a saída do filtro é

$$y(n) = \hat{x}(n) * \mathbf{i} = \hat{\mathbf{x}}^T(n) \mathbf{i} \quad (9)$$

com

$$\hat{\mathbf{x}}(n) = \{\hat{x}(n) \ \hat{x}(n-1) \ \hat{x}(n-2) \ \dots \ \hat{x}(n-M+1)\}^T \quad (10)$$

e

$$\hat{x}(n) = x(n) * \mathbf{w}_s. \quad (11)$$

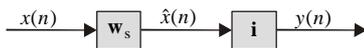


Fig. 2. Diagrama de blocos de um filtro IFIR implementado com o interpolador na saída.

As implementações apresentadas nas Figs. 1 e 2 têm o mesmo vetor de coeficientes equivalente. Tal vetor, com dimensão $N + M - 1$, é obtido por

$$\mathbf{w}_i = \mathbf{i} * \mathbf{w}_s. \quad (12)$$

Conforme descrito em [12], é possível reescrever (12) na forma vetorial, através de uma matriz de interpolação com dimensão $(N + M - 1) \times N$, dada por

$$\mathbf{I} = \begin{bmatrix} i_0 & 0 & 0 & \dots & 0 \\ i_1 & i_0 & 0 & \dots & 0 \\ i_2 & i_1 & i_0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i_{M-1} & i_{M-2} & i_{M-3} & \dots & i_0 \\ 0 & i_{M-1} & i_{M-2} & \dots & i_1 \\ 0 & 0 & i_{M-1} & \dots & i_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & i_{M-1} \end{bmatrix}. \quad (13)$$

Assim, a partir de (13), (12) é reescrita como

$$\mathbf{w}_i = \mathbf{I} \mathbf{w}_s. \quad (14)$$

De maneira análoga à matriz de interpolação \mathbf{I} , também é possível definir uma matriz \mathbf{W}_s , com dimensão $(N + M - 1) \times M$, de tal forma que

$$\mathbf{I} \mathbf{w}_s = \mathbf{W}_s \mathbf{i} = \mathbf{w}_i. \quad (15)$$

Definindo ainda o vetor de entrada estendido $(N + M - 1)$ amostras do sinal de entrada) como

$$\mathbf{x}_e(n) = [x(n) \ x(n-1) \ \dots \ x(n-N-M+2)]^T \quad (16)$$

tem-se a relação de entrada e saída do filtro IFIR dada por

$$y(n) = \mathbf{x}_e^T(n) \mathbf{I} \mathbf{w}_s = \mathbf{x}_e^T(n) \mathbf{W}_s \mathbf{i}. \quad (17)$$

É importante ressaltar que (15) e (17) são válidas tanto para o filtro IFIR com interpolador na entrada (Fig. 1) quanto para aquele com interpolador na saída (Fig. 2).

Conforme mencionado, o papel do filtro interpolador em uma estrutura IFIR é recriar os coeficientes zerados em (4). Geralmente, esses coeficientes são recriados em função dos valores de seus coeficientes vizinhos. Tal processo é aqui ilustrado através de um exemplo com $L = 2$ e usando um interpolador linear, cujo vetor de coeficientes é dado por $\mathbf{i} = [0.5 \ 1 \ 0.5]^T$. Para um filtro esparso com $N = 5$, o vetor de coeficientes é dado por

$$\mathbf{w}_s = [w_0 \ 0 \ w_2 \ 0 \ w_4]^T \quad (18)$$

e o vetor equivalente, obtido a partir de (12) ou (14), é

$$\mathbf{w}_i = [0.5w_0 \ w_0 \ \underline{0.5w_0 + 0.5w_2} \ w_2 \ \underline{0.5w_2 + 0.5w_4} \ w_4 \ \underline{0.5w_4}]^T. \quad (19)$$

Em (19), observa-se que os coeficientes diferentes de zero de (18) são replicados, enquanto os demais coeficientes são recriados em função dos primeiros. Nota-se também novos elementos no vetor equivalente [coeficientes sublinhados em (19)] decorrentes do efeito de borda [12]. Substituindo, agora, o interpolador linear por um interpolador genérico com $\mathbf{i} = [i_0 \ i_1 \ i_2]^T$, tem-se

$$\mathbf{w}_i = [i_0w_0 \ i_1w_0 \ \underline{i_2w_0 + i_0w_2} \ i_1w_2 \ \underline{i_2w_2 + i_0w_4} \ i_1w_4 \ \underline{i_2w_4}]^T. \quad (20)$$

Comparando então (19) com (20), observa-se que os coeficientes do filtro interpolador possuem papéis bastante distintos. Enquanto o coeficiente central i_1 tem a função de replicar os coeficientes diferentes de zero do filtro esparso, os coeficientes dos extremos i_0 e i_2 são os responsáveis pela recriação dos coeficientes que estão zerados a partir de seus vizinhos. Assim, de forma geral, o interpolador deve possuir um coeficiente central e $2(L-1)$ coeficientes nas extremidades, de forma que os coeficientes que estão zerados possam ser recriados em função de seus vizinhos. Portanto, o tamanho de memória do interpolador em função do fator de interpolação L é

$$M(L) = 1 + 2(L-1) = 2L - 1. \quad (21)$$

Por exemplo, um fator de interpolação $L = 3$ resulta em $M = 5$ e $\mathbf{i} = [i_0 \ i_1 \ i_2 \ i_3 \ i_4]^T$. Considerando, ainda, um filtro esparso com $N = 7$ ($N_s = 3$), tem-se

$$\mathbf{w}_s = [w_0 \ 0 \ 0 \ w_3 \ 0 \ 0 \ w_6]^T \quad (22)$$

e um vetor equivalente dado por

$$\mathbf{w}_i = [i_0w_0 \ i_1w_0 \ i_2w_0 \ \underline{i_3w_0 + i_0w_3} \ \underline{i_4w_0 + i_1w_3} \ i_2w_3 \ \underline{i_3w_3 + i_0w_6} \ \underline{i_4w_3 + i_1w_6} \ i_2w_6 \ \underline{i_3w_6} \ \underline{i_4w_6}]^T. \quad (23)$$

De (23), constata-se os coeficientes recriados (destacados pelas caixas) e aqueles sublinhados resultantes do efeito de borda. Nota-se também a replicação dos coeficientes diferentes de zero do vetor esparso de (22) aqui multiplicados pelo coeficiente central do interpolador i_2 .

III. REMOÇÃO DO EFEITO DE BORDA EM FILTROS IFIR

O efeito de borda, descrito na seção anterior, produz uma considerável degradação de desempenho para muitas aplicações [12], [13]. Em [13], é proposto um procedimento para sua remoção, resultando em um filtro IFIR sem efeito de borda (*removed border effect* IFIR - RBEIFIR). Tal procedimento é baseado na aplicação de uma transformação sobre a matriz de interpolação, resultando em

$$\mathbf{I}' = \mathbf{T} \mathbf{I} \quad (24)$$

onde \mathbf{T} é a matriz de transformação considerada com dimensão $N \times (N + M - 1)$, dada por

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \quad (25)$$

Aplicando tal transformação, o vetor de coeficientes equivalente é agora dado por

$$\mathbf{w}'_i = \mathbf{T}\mathbf{I}\mathbf{w}_s = \mathbf{T}\mathbf{W}_s\mathbf{i} = \mathbf{T}\mathbf{w}_i. \quad (26)$$

A implementação do procedimento de remoção do efeito de borda, para o filtro IFIR com interpolador na entrada (Fig. 1), é realizada substituindo o vetor de entrada do bloco esparsado de (8) por sua versão modificada

$$\tilde{\mathbf{x}}'(n) = \mathbf{I}^T \mathbf{x}(n) = \mathbf{I}^T \mathbf{T}^T \mathbf{x}(n) \quad (27)$$

com

$$\mathbf{x}(n) = [x(n) \ x(n-1) \ \cdots \ x(n-N+1)]^T. \quad (28)$$

Para o caso do filtro IFIR com interpolador na saída (Fig. 2), o procedimento de remoção do efeito de borda é análogo. Nesse caso, basta substituir o vetor de entrada do interpolador de (10) pela versão modificada

$$\hat{\mathbf{x}}'(n) = \mathbf{W}_s^T \mathbf{T}^T \mathbf{x}(n). \quad (29)$$

O vetor de coeficientes equivalente para esse caso também é obtido a partir de (26).

Neste trabalho, é generalizado o procedimento de remoção do efeito de borda apresentado em [13]. Assim, levando em conta que os $L-1$ primeiros e os $L-1$ últimos coeficientes do vetor equivalente (filtro IFIR) correspondem ao efeito de borda [ver (20) e (23)], tem-se a seguinte matriz de transformação generalizada com dimensão $N \times (N+M-1)$:

$$\mathbf{T} = \begin{bmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \vdots & \vdots & \ddots & \vdots & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \end{bmatrix}. \quad (30)$$

$L-1$ colunas N colunas $L-1$ colunas

Através de (27) e (30), obtém-se o vetor de entrada do filtro esparsado [utilizado em substituição a (8)] que remove o efeito de borda de um filtro IFIR com interpolador na entrada. Para o caso do filtro IFIR considerando interpolador na saída, obtém-se o vetor de entrada do interpolador a partir de (29) e (30). Por exemplo, a aplicação desse procedimento, para o mesmo caso de (23), resulta no seguinte vetor equivalente modificado:

$$\mathbf{w}'_i = \begin{bmatrix} i_2 w_0 & i_3 w_0 + i_0 w_3 & i_4 w_0 + i_1 w_3 & i_2 w_3 \\ i_3 w_3 + i_0 w_6 & i_4 w_3 + i_1 w_6 & i_2 w_6 \end{bmatrix}^T. \quad (31)$$

O custo computacional para implementação do procedimento de remoção do efeito de borda é muito pequeno [13], o qual é exemplificado a seguir. Assim, tomando o caso de um filtro IFIR com interpolador na entrada e com $L=2$, $M=3$, $N=5$ e $\mathbf{i} = [i_0 \ i_1 \ i_2]^T$, tem-se

$$\tilde{\mathbf{x}}'(n) = \mathbf{I}^T \mathbf{T}^T \mathbf{x}(n) = \begin{bmatrix} i_1 x(n) + i_2 x(n-1) \\ \mathbf{i}^T \mathbf{x}_m(n) \\ \mathbf{i}^T \mathbf{x}_m(n-1) \\ \mathbf{i}^T \mathbf{x}_m(n-2) \\ i_0 x(n-3) + i_1 x(n-4) \end{bmatrix}^T \quad (32)$$

com

$$\mathbf{x}_m(n) = [x(n) \ x(n-1) \ \cdots \ x(n-M+1)]^T. \quad (33)$$

Para se obter (32), a cada iteração, são necessárias as seguintes operações:

i) duas multiplicações e uma soma para obter o primeiro elemento para a iteração atual;

ii) uma multiplicação e uma soma para obter o segundo elemento da iteração atual a partir do primeiro elemento da iteração anterior;

iii) uma multiplicação e uma subtração para obter o último elemento da iteração atual a partir do penúltimo elemento da iteração anterior.

Portanto, no total, são necessárias 4 multiplicações e 3 somas para se computar (32), enquanto no caso do filtro IFIR convencional são necessárias 3 multiplicações e 2 somas. Generalizando, tem-se um total de $M+L-1$ multiplicações e $M+L-2$ somas para a filtragem modificada do filtro RBEIFIR, contra M multiplicações e $M-1$ somas para a filtragem padrão do filtro IFIR. Assim, observa-se um incremento de apenas $2L-2$ operações por amostra. Como o valor de L é geralmente pequeno, tal incremento é praticamente desprezível. Para o caso do filtro IFIR com interpolador na saída, realiza-se a implementação de $\hat{\mathbf{x}}'(n) = \mathbf{W}_s^T \mathbf{T}^T \mathbf{x}(n)$ de forma similar à descrita para (32), resultando também em um acréscimo de apenas $2L-2$ operações por amostra.

IV. FILTROS IFIR INTEIRAMENTE ADAPTATIVOS SEM EFEITO DE BORDA

A abordagem mais simples e mais utilizada para implementação de filtros IFIR adaptativos é através da adaptação apenas dos coeficientes do filtro esparsado [7], [8]. Tal abordagem, conhecida por IFIR adaptativa (*adaptive* IFIR - AIFIR), é usualmente aplicada à estrutura IFIR com interpolador na entrada (Fig. 1) [7]. O procedimento de remoção do efeito de borda, descrito na Seção III, pode ser diretamente aplicado ao filtro AIFIR com interpolador na entrada [13]. A estrutura resultante, denominada RBEIFIR adaptativa (*adaptive* RBEIFIR - ARBEIFIR), apresenta um considerável ganho de desempenho em relação ao filtro AIFIR convencional [13].

Por outro lado, a eficiência dos filtros IFIR adaptativos com interpoladores fixos (AIFIR e ARBEIFIR) depende muito da escolha dos coeficientes do interpolador, o que resulta muitas vezes em perdas de desempenho. A solução para tal problema é utilizar uma implementação inteiramente adaptativa do filtro IFIR (*fully adaptive* IFIR - FAIFIR), em que tanto o filtro esparsado quanto o interpolador são adaptados [9]-[11]. Apesar de seu melhor desempenho em regime permanente, a estrutura FAIFIR apresenta algumas desvantagens em relação à estrutura AIFIR. Dentre elas podemos citar a maior carga computacional requerida para a sua implementação e a necessidade de inicialização dos coeficientes com valores diferentes de zero, tanto para o filtro FIR esparsado quanto para o interpolador [9].

A aplicação do procedimento de remoção do efeito de borda nos filtros FAIFIR é um problema um tanto complexo. Inicialmente, levando em conta que o algoritmo adaptativo realiza a otimização dos coeficientes através da minimização de uma função custo, é comum supor que tal minimização seja capaz de lidar automaticamente com o efeito de borda. De fato, uma redução parcial de tal efeito é observada, porém às custas de uma certa perda de desempenho no processo de recriação dos coeficientes. Além do mais, uma análise cuidadosa dos vetores de coeficientes equivalentes leva a outras conclusões bastante interessantes. É bem conhecido o fato de que quanto maior for a correlação entre os coeficientes, melhor serão os resultados obtidos através do processo de interpolação [1], [8]. No entanto, é possível verificar, a partir de (20) e (23), que o efeito de borda corresponde à existência de dois coeficientes iguais a zero no início e no final do vetor de coeficientes do filtro. Isso fica evidente reescrevendo (20) como segue:

$$\mathbf{w}_i = \begin{bmatrix} i_2 0 + i_0 w_0 & i_1 w_0 & i_2 w_0 + i_0 w_2 & i_1 w_2 \\ i_2 w_2 + i_0 w_4 & i_1 w_4 & i_2 w_4 + i_0 0 \end{bmatrix}^T. \quad (34)$$

Em (34), nota-se que os valores dos coeficientes (sublinhados) relativos ao efeito de borda envolvem zeros que antecedem e sucedem, respectivamente, o primeiro e o último coeficiente do filtro esparso. Assim, temos um efeito correspondente à existência de dois coeficientes iguais a zero que usualmente não são correlacionados aos demais, resultando em uma degradação de desempenho do processo de interpolação. Outro aspecto é o papel do coeficiente central do interpolador, tendo a função de reproduzir os coeficientes do filtro esparso no filtro equivalente. Assim, ele aparece apenas como um fator multiplicativo em relação aos coeficientes do filtro esparso, sem qualquer influência nos demais coeficientes recriados. Como consequência, é possível, por exemplo, fixar o valor desse coeficiente em 1, sem qualquer perda de generalidade da estrutura equivalente.

O processo de adaptação do filtro FAIFIR sem efeito de borda (*fully adaptive* RBEIFIR - FARBEIFIR) é um pouco mais complexo do que o seu congênere [AIFIR sem efeito de borda (ARBEIFIR)]. Isso ocorre porque a transformação usada sobre matriz de interpolação não permite a aplicação direta das expressões dos algoritmos LMS e NLMS. Assim, torna-se necessário uma adequação de tais expressões para a estrutura FARBEIFIR, ilustrada na Fig. 3. Nessa figura temos, além dos mesmos sinais definidos na Fig. 1, o sinal desejado $d(n)$ e o sinal de erro $e(n)$, dado por

$$e(n) = d(n) - y(n). \quad (35)$$

Observa-se da Fig. 3 que tanto o interpolador quanto o filtro esparso são agora funções de n , uma vez que eles são variantes no tempo.

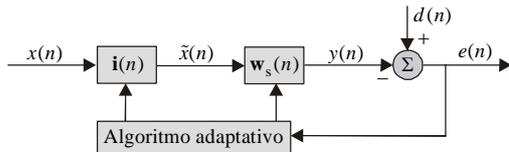


Fig. 3. Filtro IFIR no contexto adaptativo.

Considerando (27) e (29), é possível reescrever (35) para o caso do filtro FARBEIFIR como

$$\begin{aligned} e(n) &= d(n) - \mathbf{w}_s^T(n) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) \\ &= d(n) - \hat{\mathbf{i}}^T(n) \mathbf{W}_s^T(n) \mathbf{T}^T \mathbf{x}(n). \end{aligned} \quad (36)$$

É importante observar que, apesar da estrutura da Fig. 3 apresentar o interpolador na entrada do filtro IFIR, (35) e (36) são válidas também para o caso do filtro IFIR com interpolador na saída.

A. Algoritmo LMS

A função custo utilizada para derivar o algoritmo LMS é [14]

$$\hat{J}(n) = e^2(n). \quad (37)$$

Conforme descrito em [14], os coeficientes são adaptados usando o gradiente da função custo. Dessa maneira, o processo de adaptação do vetor de coeficientes do filtro esparso é obtido por

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) - \mu_1 \nabla_{\mathbf{w}_s} e^2(n) \quad (38)$$

com μ_1 representando o passo de adaptação. Aplicando a regra da cadeia, temos o gradiente de (38) em relação a $\mathbf{w}_s(n)$, dado por

$$\nabla_{\mathbf{w}_s} e^2(n) = \frac{\partial e^2(n)}{\partial \mathbf{w}_s(n)} = \frac{\partial e^2(n)}{\partial e(n)} \frac{\partial e(n)}{\partial \mathbf{w}_s(n)}. \quad (39)$$

Resolvendo os termos de (39), tem-se

$$\frac{\partial e^2(n)}{\partial e(n)} = 2e(n) \quad (40)$$

e

$$\frac{\partial e(n)}{\partial \mathbf{w}_s(n)} = -\mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n). \quad (41)$$

Portanto, a equação de adaptação do vetor de coeficientes do filtro esparso FARBEFIR é dada por

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + 2\mu_1 e(n) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n). \quad (42)$$

A adaptação do interpolador também é realizada em função do gradiente da função custo (37), porém agora em relação aos coeficientes do interpolador. Assim tem-se

$$\hat{\mathbf{i}}(n+1) = \hat{\mathbf{i}}(n) - \mu_2 \nabla_{\hat{\mathbf{i}}} e^2(n) \quad (43)$$

com μ_2 denotando o passo de adaptação. Adotando um procedimento similar ao realizado para o vetor de coeficientes do filtro esparso, obtém-se

$$\hat{\mathbf{i}}(n+1) = \hat{\mathbf{i}}(n) + 2\mu_2 e(n) \mathbf{W}_s^T(n) \mathbf{T}^T \mathbf{x}(n). \quad (44)$$

As equações de adaptação dos coeficientes descritas em (42) e (44) apresentam uma carga computacional elevada devido à existência de produtos das matrizes $\mathbf{I}(n)$, $\mathbf{W}_s(n)$ e \mathbf{T} . No entanto, é possível realizar uma aproximação, assumindo que os coeficientes variam pouco a cada iteração (condição de adaptação lenta). O resultado é a implementação de $\mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n)$ e $\mathbf{W}_s^T(n) \mathbf{T}^T \mathbf{x}(n)$ de forma similar à descrita na Seção III para a implementação de $\tilde{\mathbf{x}}'(n) = \mathbf{I}^T \mathbf{T}^T \mathbf{x}(n)$ e $\hat{\mathbf{x}}'(n) = \mathbf{W}_s^T \mathbf{T}^T \mathbf{x}(n)$. Assim, é possível reescrever (42) como

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + 2\mu_1 e(n) \tilde{\mathbf{x}}'(n) \quad (45)$$

e (44) como

$$\hat{\mathbf{i}}(n+1) = \hat{\mathbf{i}}(n) + 2\mu_2 e(n) \hat{\mathbf{x}}'(n). \quad (46)$$

As implementações dos filtros FAIFIR apresentadas em [9] e [10] também são realizadas usando aproximações e resultam em expressões similares a (45) e (46), porém sem a remoção do efeito de borda.

B. Algoritmo NLMS

De forma similar, a equação de adaptação do vetor de coeficientes esparso do filtro FARBEIFIR é obtida minimizando a norma euclidiana de [14]

$$\delta \mathbf{w}_s(n+1) = \mathbf{w}_s(n+1) - \mathbf{w}_s(n) \quad (47)$$

submetida à seguinte restrição:

$$\mathbf{w}_s^T(n+1) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) = d(n). \quad (48)$$

Nota-se que a restrição (48) é ligeiramente diferente daquela apresentada em [14]. Isso é devido às características particulares do filtro FARBEIFIR. Aplicando o método dos multiplicadores de Lagrange em (47) e (48), obtém-se a seguinte função custo:

$$J_{w_s}(n) = \|\delta \mathbf{w}_s(n+1)\|^2 + \lambda_1 \left[d(n) - \mathbf{w}_s^T(n+1) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) \right] \quad (49)$$

onde λ_1 é um multiplicador de Lagrange com valor real. Diferenciando (49) em relação a $\mathbf{w}_s(n+1)$, tem-se

$$\frac{\partial J_{w_s}(n)}{\partial \mathbf{w}_s(n+1)} = 2[\mathbf{w}_s(n+1) - \mathbf{w}_s(n)] - \lambda_1 \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n). \quad (50)$$

Fazendo (50) igual a zero, obtém-se

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + \frac{1}{2} \lambda_1 \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n). \quad (51)$$

Substituindo (51) em (48), resulta em

$$d(n) = \left(\mathbf{w}_s(n) + \frac{1}{2} \lambda_1 \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) \right)^T \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) \quad (52)$$

$$= \mathbf{w}_s^T(n) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n) + \frac{1}{2} \lambda_1 \|\mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n)\|^2.$$

Resolvendo (52) para λ_1 , obtém-se

$$\lambda_1 = \frac{2[d(n) - \mathbf{w}_s^T(n) \mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n)]}{\|\mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n)\|^2} = \frac{2e(n)}{\|\mathbf{I}^T(n) \mathbf{T}^T \mathbf{x}(n)\|^2}. \quad (53)$$

Agora, substituindo (53) em (51) e adicionando uma constante positiva α_1 para controlar o processo de adaptação [14] e, ainda,

uma outra constante pequena e positiva ψ_1 para prevenir a divisão por zero [14], a equação de adaptação é finalmente dada por

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + \frac{\alpha_1}{\|\mathbf{I}^T(n)\mathbf{T}^T\mathbf{x}(n)\|^2 + \psi_1} e(n)\mathbf{I}^T(n)\mathbf{T}^T\mathbf{x}(n). \quad (54)$$

Então, aplicando uma aproximação similar à utilizada para derivar o algoritmo LMS, obtém-se a equação de adaptação do algoritmo NLMS para o filtro esparsa da estrutura FARBEIFIR. Assim,

$$\mathbf{w}_s(n+1) = \mathbf{w}_s(n) + \frac{\alpha_1}{\|\tilde{\mathbf{x}}'(n)\|^2 + \psi_1} e(n)\tilde{\mathbf{x}}'(n). \quad (55)$$

Um procedimento similar é realizado para o interpolador, resultando na seguinte equação de adaptação:

$$\mathbf{i}(n+1) = \mathbf{i}(n) + \frac{\alpha_2}{\|\hat{\mathbf{x}}'(n)\|^2 + \psi_2} e(n)\hat{\mathbf{x}}'(n). \quad (56)$$

Observa-se, ainda, que (55) e (56) são obtidas assumindo uma variação lenta dos coeficientes e, portanto, um cuidado especial na escolha das constantes de controle α_1 e α_2 deve ser tomado. Em [10] e [11], também são utilizadas aproximações similares.

C. Considerações sobre o Processo Adaptativo

As expressões do algoritmo LMS e NLMS, desenvolvidas para o filtro FARBEIFIR nas Seções IV-A e IV-B, são válidas tanto para estrutura IFIR com interpolador na entrada quanto para a estrutura com interpolador na saída. No entanto, apesar das duas estruturas possuírem as mesmas equações de adaptação e também apresentarem o mesmo vetor de coeficientes equivalente, a complexidade computacional requerida para implementação de cada uma delas é diferente. Em ambos os casos, é necessário calcular tanto $\tilde{\mathbf{x}}'(n)$ quanto $\hat{\mathbf{x}}'(n)$ para realizar a adaptação, o que resulta, respectivamente, em $2M + 2L - 3$ e $2N_s + 2L - 3$ operações de multiplicação e soma por amostra. Além disso, é necessário calcular o sinal de saída do filtro através de

$$y(n) = \mathbf{w}_s^T(n)\tilde{\mathbf{x}}'(n) \quad (57)$$

para a estrutura com interpolador na entrada, e

$$y(n) = \mathbf{i}^T(n)\hat{\mathbf{x}}'(n) \quad (58)$$

para a estrutura com interpolador na saída. O cálculo de (57) implica em $2N_s - 1$ operações de multiplicação e soma, enquanto o cálculo de (58) implica em $2M - 1$ dessas operações. Assim, considerando que na grande maioria dos casos temos $N_s > M$, a implementação de (58) (estrutura com interpolador na saída) é mais vantajosa em termos de complexidade computacional. Tal característica também é observada na implementação do filtro adaptativo FAIFIR.

V. RESULTADOS DE SIMULAÇÃO

Nesta seção, alguns resultados de simulação são apresentados com o objetivo de verificar o desempenho dos filtros IFIR inteiramente adaptativos sem o efeito de borda (FARBEIFIR). Os exemplos considerados focam sobre problemas de identificação de sistemas [14], comparando o desempenho em termos do EQM dos filtros FARBEIFIR com o de outras implementações de filtros FIR interpolados (AIFIR, FAIFIR e ARBEIFIR). Nas simulações com interpolador fixo (AIFIR e ARBEIFIR), são utilizados os interpoladores lineares $\mathbf{i} = [0,5 \ 1 \ 0,5]^T$, para $L = 2$ e $\mathbf{i} = [0,25 \ 0,75 \ 1 \ 0,75 \ 0,25]^T$, para $L = 3$. Em todos os exemplos, o sinal de entrada é Gaussiano branco com variância $\sigma_x^2 = 1$. É também incluído no processo, somado ao sinal $d(n)$, um ruído aditivo Gaussiano com variância $\sigma_z^2 = 10^{-8}$.

Exemplo 1: Neste exemplo, a planta a ser modelada apresenta uma

resposta ao impulso exponencial decrescente, conforme ilustrado na Fig. 4. O tamanho de memória é $N = 11$ e o fator de interpolação é $L = 2$. Os filtros interpolados são adaptados usando o algoritmo LMS com $\mu = 0,1\mu_{\max}$, onde μ_{\max} é o valor máximo de μ para o qual o algoritmo é ainda estável (obtido experimentalmente). Para efeito de comparação, o desempenho de um filtro FIR convencional adaptativo (AFIR) também está incluído neste exemplo. Os resultados obtidos são apresentados na Fig. 5; observa-se que o filtro FARBEIFIR apresenta um desempenho excelente para este caso, alcançando um EQM idêntico ao nível do ruído em regime permanente.

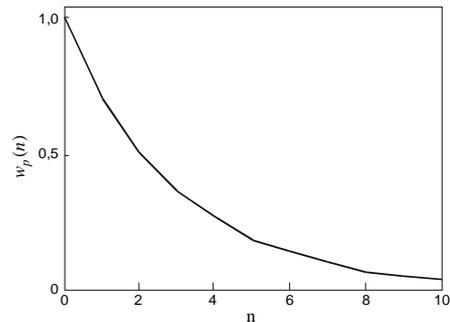


Fig. 4. Planta utilizada no Exemplo 1.

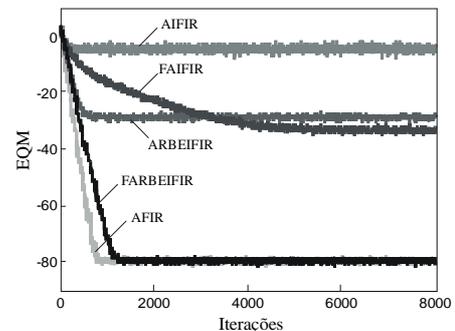


Fig. 5. Curvas de EQM para o Exemplo 1.

Exemplo 2: Para este exemplo, a planta tem $N = 102$ coeficientes cuja resposta ao impulso está ilustrada na Fig. 6. Neste caso, nota-se que o efeito de borda apresenta menor influência devido à pequena magnitude dos coeficientes das bordas. O algoritmo adaptativo utilizado é o LMS com $\mu = 0,2\mu_{\max}$. A Fig. 7 mostra os resultados obtidos com $L = 2$; observa-se que o desempenho das quatro estruturas é bastante próximo. Assim, nota-se que além do efeito de borda apresentar pouca influência, os interpoladores fixos são adequados para este caso, conferindo ainda uma ligeira vantagem do filtro FARBEIFIR sobre os demais. Na Fig. 8, são apresentados os resultados para o mesmo problema, porém agora com $L = 3$. Nota-se que com um fator de interpolação maior, os filtros inteiramente adaptativos têm um melhor desempenho e, dentre eles, o filtro FARBEIFIR apresenta uma pequena vantagem.

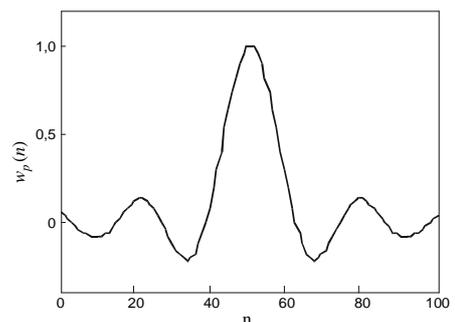
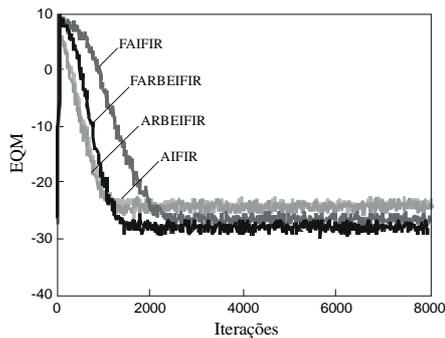
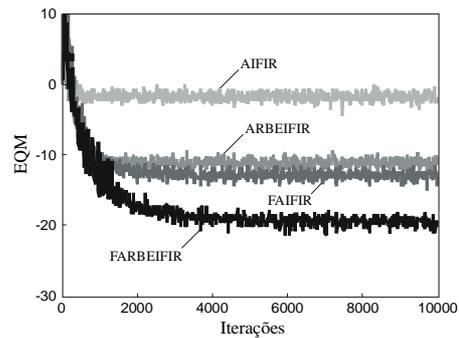
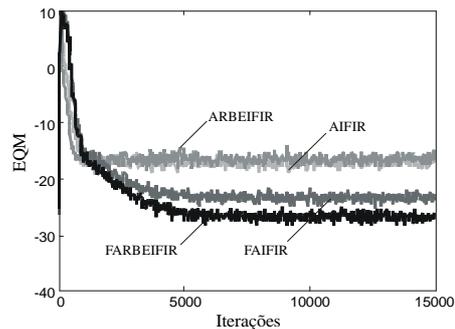


Fig. 6. Planta utilizada no Exemplo 2.

Fig. 7. Curvas de EQM para o Exemplo 2 com $L = 2$.Fig. 11. Curvas de EQM para o Exemplo 3 com $L = 3$.Fig. 8. Curvas de EQM para o Exemplo 2 com $L = 3$.

Exemplo 3: Neste exemplo, a planta usada tem $N = 72$ coeficientes, (Fig. 9). Trata-se de uma planta sofrendo uma considerável influência do efeito de borda. O algoritmo adaptativo usado é o NLMS com $\psi_1 = \psi_2 = 10^{-8}$ e constantes de controle iguais a $\alpha_1 = 0,2$ para o filtro esparsos e $\alpha_2 = 0,02$ para o interpolador. Na Fig. 10, são mostrados os resultados obtidos para $L = 2$, enquanto a Fig. 11 ilustra os resultados para $L = 3$. Em ambos os casos, nota-se um desempenho superior do filtro FARBEIFIR em relação às outras implementações consideradas de filtros IFIR.

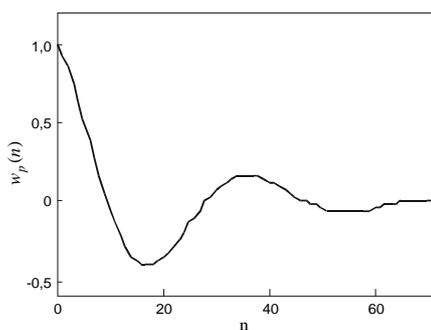
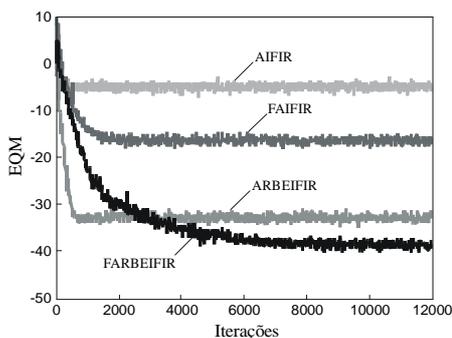


Fig. 9. Planta usada no Exemplo 3.

Fig. 10. Curvas de EQM para o Exemplo 3 com $L = 2$.

VI. CONCLUSÕES

Neste trabalho, um novo procedimento para implementação de filtros IFIR inteiramente adaptativos e sem o efeito de borda foi apresentado. Em comparação com as outras implementações de filtros IFIR disponíveis na literatura, o procedimento proposto produz consideráveis ganhos de desempenho em termos de EQM com um acréscimo de custo computacional praticamente desprezível. Através dos resultados de simulação, foi possível verificar a efetividade da abordagem aqui proposta.

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. 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] 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.
- [8] 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.
- [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] E. L. O. Batista, O. J. Tobias, and R. Seara, "A mathematical framework to describe interpolated adaptive Volterra filters," in *Proc. IEEE Int. Telecomm. Symp.*, Fortaleza, Brazil, Sept. 2006, pp. 144-149.
- [13] E. L. O. Batista, O. J. Tobias, and R. Seara, "Border effect removal for IFIR and interpolated Volterra filters," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Honolulu, USA, vol. 3, Apr. 2007, pp. 1329-1332.
- [14] S. Haykin, *Adaptive Filter Theory*, 4 ed., Prentice-Hall, 2002.