

Algoritmo LMS com Entrada Filtrada/Erro Filtrado Modificado: Algoritmo e Modelo Estocástico

Juan Rodrigo Velásquez López, Orlando José Tobias e Rui Seara

Resumo—Neste artigo, um algoritmo LMS com entrada filtrada/erro filtrado (MFxFeLMS) é proposto. Adicionalmente, um modelo estocástico para os momentos de primeira e segunda ordens do algoritmo MFxFeLMS é derivado. O modelo proposto é obtido sem invocar a clássica Teoria da Independência e considerando a hipótese de adaptação lenta como também sinais de entrada gaussianos. Simulações numéricas mostram muito boa concordância entre os resultados obtidos através do método de Monte Carlo (MC) e do modelo proposto tanto para sinais de entrada gaussianos brancos quanto coloridos.

Palavras-chave—Algoritmo LMS com entrada filtrada/erro filtrado, modelo estocástico, momentos de primeira e segunda ordens.

Abstract—This paper proposes a modified filtered-reference /filtered-error least-mean-square (MFxFeLMS) algorithm. In addition, a stochastic model for the first and second moments of the MFxFeLMS algorithm is derived. The proposed model is obtained without invoking the classic Independence Theory (IT), allowing for slow adaptation and Gaussian input signals. Numerical simulations confirm the very good agreement between the results obtained by the Monte Carlo (MC) method and from the proposed model for both white and colored Gaussian inputs.

Keywords—Filtered-reference/filtered-error LMS algorithm, stochastic model, first and second moments.

I. INTRODUÇÃO

Em aplicações de controle ativo de ruído e vibrações, uma implementação direta da estrutura de filtragem adaptativa nem sempre é possível. Tal problema ocorre porque, em algumas situações, o sinal de erro utilizado para atualizar os coeficientes do filtro adaptativo não se encontra disponível (ou acessível) para ser usado diretamente pelo algoritmo. Nesses casos, apenas tem-se disponível uma versão filtrada da saída do filtro adaptativo ou do sinal de erro, como ocorre em controle de ruído/vibrações e outras aplicações similares.

Juan R. V. López e Rui Seara, LINSE – Laboratório de Circuitos e Processamento de Sinais, Depto. de Eng. Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, E-mails: {juan, 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 pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Dessa forma, a família de algoritmos LMS filtrados (FLMS) deve então ser usada. No entanto, tais algoritmos apresentam um comportamento pobre com vistas às propriedades de velocidade de convergência e estabilidade. Para contornar esses problemas em [1], foi proposta uma modificação no algoritmo adaptativo fazendo com que seu comportamento passasse a ser similar ao do algoritmo LMS convencional tanto em termos de velocidade quanto estabilidade. Tal algoritmo é denominado algoritmo FLMS modificado (MFLMS) [1], [2].

Na literatura, podemos encontrar quatro tipos de algoritmos pertencentes à família dos algoritmos adaptativos de erro filtrado, a saber: LMS filtrado (ou LMS de entrada filtrada) (FxLMS) [3], [4]; LMS de erro filtrado (FeLMS) [5], [6]; FxLMS modificado (MFxLMS) [2] e FeLMS modificado (MFeLMS) [7]. Os dois primeiros algoritmos são amplamente usados em aplicações de controle ativo de ruído. Já os algoritmos MFxLMS e MFeLMS são versões modificadas, respectivamente, dos algoritmos FxLMS e FeLMS. Estas duas últimas versões superam os problemas de convergência apresentados nas suas respectivas versões convencionais. Adicionalmente, um algoritmo generalizado agrupando os algoritmos FxLMS e FeLMS foi proposto em [8], [9], denominado algoritmo LMS com entrada filtrada/erro filtrado (FxFeLMS). Muitas das análises das diferentes versões do algoritmo LMS têm sido realizadas sob a luz da Teoria da Independência (TI) [10],[11]. No entanto, em certas situações, a TI não pode mais ser invocada, como na modelagem dos algoritmos FxLMS e MFeLMS. Dessa forma, os modelos analíticos dos momentos de primeira e segunda ordens dos algoritmos FxLMS e MFeLMS vêm sendo derivados sem ser invocada a TI [12]-[13] e [7]. Em nosso conhecimento, na literatura da área, não tem sido proposto até então uma versão modificada do algoritmo FxFeLMS (MFxFeLMS) e muito menos o seu modelo estocástico. Tal algoritmo e modelo são as contribuições principais deste trabalho. Visto que o uso da TI não é mais adequado para esta classe de algoritmo (como discutido em [7], [12] e [13]), tal hipótese não é aqui considerada. Dessa forma, os modelos propostos para o comportamento médio do vetor de coeficientes e para a curva de aprendizagem são muito mais precisos do que aqueles obtidos sob a hipótese da TI. Através de simulações numéricas, usando o método de Monte Carlo (MC), o desempenho do algoritmo MFxFeLMS é comparado com o do FxFeLMS convencional [8]. Além disso, é verificada uma

concordância muito boa entre simulações MC e os resultados do modelo proposto para o algoritmo MFxFeLMS. Para as avaliações consideradas, são usados tanto sinais de entrada gaussianos brancos quanto coloridos.

II. ALGORITMO MFxFeLMS MODIFICADO

A. Algoritmo FxFeLMS

A Fig. 1 mostra o diagrama em blocos do algoritmo FxFeLMS no qual a seguinte notação é utilizada: $\mathbf{w}_o = [w_{o,0} \ w_{o,1} \ \dots \ w_{o,N-1}]^T$ representa a resposta ao impulso da planta, $\mathbf{w}(n) = [w_0(n) \ w_1(n) \ \dots \ w_{N-1}(n)]^T$ denota o vetor de coeficientes do filtro adaptativo, $\mathbf{s}_1 = [s_{1,0} \ s_{1,1} \ \dots \ s_{1,M_1-1}]^T$, $\mathbf{s}_2 = [s_{2,0} \ s_{2,1} \ \dots \ s_{2,M_2-1}]^T$ e $\hat{\mathbf{s}} = [\hat{s}_0 \ \hat{s}_1 \ \dots \ \hat{s}_{\hat{M}-1}]^T$ caracterizam as respostas ao impulso dos filtros localizados, respectivamente, na saída do filtro adaptativo, no caminho do erro e no caminho do sinal de entrada. Note que $\hat{\mathbf{s}}$ é uma estimativa do produto convolução entre \mathbf{s}_1 e \mathbf{s}_2 . Os sinais $d(n)$ e $z(n)$ são, respectivamente, o sinal desejado e o ruído aditivo de medição (i.i.d de média zero e variância σ_z^2 , decorrelacionado de qualquer outro sinal do sistema).

Neste trabalho, o vetor de entrada é $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T$, com $\{x(n)\}$ caracterizando um processo gaussiano de média zero e variância σ_x^2 . O vetor do sinal de entrada filtrado é denotado por $\mathbf{x}_f(n) = [x_f(n) \ x_f(n-1) \ \dots \ x_f(n-N+1)]^T$, com

$$\mathbf{x}_f(n) = \sum_{i=0}^{\hat{M}-1} \hat{s}_i \mathbf{x}(n-i). \quad (1)$$

Assume-se também que os vetores \mathbf{w}_o e $\mathbf{w}(n)$ têm o mesmo comprimento, enquanto o comprimento dos vetores \mathbf{s}_1 e \mathbf{s}_2 podem ser diferentes.

Da Fig. 1, o sinal de erro é dado por

$$e(n) = \mathbf{w}_o^T \mathbf{x}(n) - \sum_{i=0}^{M_1-1} s_{1,i} \mathbf{w}^T(n-i) \mathbf{x}(n-i) + z(n). \quad (2)$$

Note que o sinal de erro atual depende dos valores passados de $\mathbf{w}(n)$.

O sinal de erro filtrado, usado para a atualização do vetor de coeficientes do algoritmo adaptativo, é expresso como

$$e_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} e(n-i). \quad (3)$$

Então, substituindo (2) em (3), obtém-se

$$e_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_o^T \mathbf{x}(n-i) + z_f(n) - \sum_{i=0}^{M_2-1} \sum_{j=0}^{M_1-1} s_{2,i} s_{1,j} \mathbf{w}^T(n-i-j) \mathbf{x}(n-i-j) \quad (4)$$

com

$$z_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} z(n-i). \quad (5)$$

Alternativamente, (4) pode ser rescrita como

$$e_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_o^T \mathbf{x}(n-i) + z_f(n) - \sum_{i=0}^{M_3-1} s_{3,i} \mathbf{w}^T(n-i) \mathbf{x}(n-i) \quad (6)$$

com

$$\mathbf{s}_3 = \mathbf{s}_1 * \mathbf{s}_2 \quad (7)$$

onde $*$ denota o operador de convolução. Dessa forma, a expressão de atualização do vetor de coeficientes é dada por [8]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e_f(n) \mathbf{x}_f(n). \quad (8)$$

B. Algoritmo FxFeLMS Modificado

O algoritmo MFxFeLMS origina-se da compensação do erro instantâneo [expressão (6)] do algoritmo FxFeLMS. Tal compensação é obtida pela inclusão do termo $-\Lambda_f(n)$ (denominado termo de compensação) em (6) [1]. Dessa forma, as expressões de atualização do vetor de coeficientes e do erro compensado $\hat{e}_f(n)$ são, respectivamente, dadas por

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \hat{e}_f(n) \mathbf{x}_f(n) \quad (9)$$

e

$$\hat{e}_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_o^T \mathbf{x}(n-i) + z_f(n) - \sum_{i=0}^{M_3-1} s_{3,i} \mathbf{w}^T(n-i) \mathbf{x}(n-i) - \Lambda_f(n). \quad (10)$$

O termo de compensação $\Lambda_f(n)$ é determinado forçando (10) ser igual ao sinal de erro do algoritmo LMS convencional [1], segundo o qual

$$e_{\text{LMS}}(n) = \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_o^T \mathbf{x}(n-i) + z_f(n) - \sum_{i=0}^{M_3-1} s_{3,i} \mathbf{w}^T(n) \mathbf{x}(n-i). \quad (11)$$

Note que em (11) o sinal de erro agora depende do valor atual do vetor do filtro adaptativo $\mathbf{w}(n)$ ao invés do vetor $\mathbf{w}(n-i)$. Então, de (10) e (11), o termo de compensação requerido é

$$\Lambda_f(n) = \sum_{i=0}^{M_3-1} s_{3,i} [\mathbf{w}^T(n) - \mathbf{w}^T(n-i)] \mathbf{x}(n-i). \quad (12)$$

Em (12), a diferença $[\mathbf{w}^T(n) - \mathbf{w}^T(n-i)] = 0$ para $i=0$; caso contrário, $[\mathbf{w}^T(n) - \mathbf{w}^T(n-i)]$ é obtida a partir de (9) como segue. Primeiro, note que

$$i = 1 \Rightarrow \mathbf{w}(n) = \mathbf{w}(n-1) + \mu \hat{e}_f(n-1) \mathbf{x}_f(n-1) \quad (13)$$

$$i = 2 \Rightarrow \mathbf{w}(n-1) = \mathbf{w}(n-2) + \mu \hat{e}_f(n-2) \mathbf{x}_f(n-2) \quad (14)$$

$$i = L \Rightarrow \mathbf{w}(n-L+1) = \mathbf{w}(n-L) + \mu \hat{e}_f(n-L) \mathbf{x}_f(n-L) \quad (15)$$

com $L = M_3 - 1$. Agora, determinando as diferenças algébricas de (9) com (13), (14), e (15), respectivamente, obtém-se

$$\begin{bmatrix} \mathbf{w}^T(n+1) - \mathbf{w}^T(n) \\ \mathbf{w}^T(n+1) - \mathbf{w}^T(n-1) \\ \mathbf{w}^T(n+1) - \mathbf{w}^T(n-2) \\ \vdots \\ \mathbf{w}^T(n+1) - \mathbf{w}^T(n-L+1) \end{bmatrix} = \begin{bmatrix} \mathbf{w}^T(n) - \mathbf{w}^T(n-1) \\ \mathbf{w}^T(n) - \mathbf{w}^T(n-2) \\ \mathbf{w}^T(n) - \mathbf{w}^T(n-3) \\ \vdots \\ \mathbf{w}^T(n) - \mathbf{w}^T(n-L) \end{bmatrix} \quad (16)$$

$$+ \mu \hat{e}_f(n) \begin{bmatrix} \mathbf{x}_f^T(n) \\ \mathbf{x}_f^T(n) \\ \mathbf{x}_f^T(n) \\ \vdots \\ \mathbf{x}_f^T(n) \end{bmatrix} - \mu \begin{bmatrix} \hat{e}_f(n-1) \mathbf{x}_f^T(n-1) \\ \hat{e}_f(n-2) \mathbf{x}_f^T(n-2) \\ \hat{e}_f(n-3) \mathbf{x}_f^T(n-3) \\ \vdots \\ \hat{e}_f(n-L) \mathbf{x}_f^T(n-L) \end{bmatrix}.$$

Então, utilizando (13)-(15) para determinar, respectivamente, os termos $\hat{e}_f(n-1) \mathbf{x}_f^T(n-1)$ até $\hat{e}_f(n-L) \mathbf{x}_f^T(n-L)$, e substituindo as expressões resultantes em (16), após uma simples manipulação algébrica, obtém-se

$$\begin{bmatrix} \mathbf{a}_1^T(n+1) \\ \mathbf{a}_2^T(n+1) \\ \mathbf{a}_3^T(n+1) \\ \vdots \\ \mathbf{a}_{M_3-1}^T(n+1) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{a}_1^T(n) \\ \mathbf{a}_2^T(n) \\ \vdots \\ \mathbf{a}_{M_3-2}^T(n) \end{bmatrix} + \mu \hat{e}_f(n) \begin{bmatrix} \mathbf{x}_f^T(n) \\ \mathbf{x}_f^T(n) \\ \mathbf{x}_f^T(n) \\ \vdots \\ \mathbf{x}_f^T(n) \end{bmatrix} \quad (17)$$

para

$$\mathbf{a}_j^T(n) = [\mathbf{w}^T(n) - \mathbf{w}^T(n-j)], \quad j = 1, 2, \dots, M_3 - 1. \quad (18)$$

Alternativamente, (17) pode ser reescrita como

$$\mathbf{a}_j(n+1) = \begin{cases} \mu \hat{e}_f(n) \mathbf{x}_f(n), & j = 1 \\ \mathbf{a}_{j-1}(n) + \mu \hat{e}_f(n) \mathbf{x}_f(n), & 2 \leq j \leq M_3 - 1. \end{cases} \quad (19)$$

Finalmente, substituindo (18) em (12) e o resultado obtido em (10), a expressão do sinal de erro compensado é dada por

$$\hat{e}_f(n) = \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_0^T \mathbf{x}(n-i) - \sum_{i=0}^{M_3-1} s_{3,i} \mathbf{w}^T(n-i) \mathbf{x}(n-i) - \sum_{i=1}^{M_3-1} s_{3,i} \mathbf{a}_i^T(n) \mathbf{x}(n-i) + z_f(n). \quad (20)$$

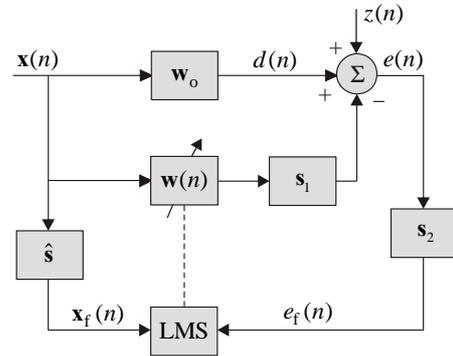


Fig. 1. Diagrama em bloco do algoritmo FxFeLMS.

III. MODELAGEM

Para determinar as expressões do modelo para o algoritmo MFxFeLMS, consideraremos a seguinte assunção A1: as correlações entre os diferentes vetores de entrada são muito mais importantes do que as correlações entre vetores de entrada e vetores de coeficientes. Tal assunção pode ser estendida para as correlações entre vetores de entrada e os vetores \mathbf{a}_j .

A. Comportamento Médio do Vetor de Coeficientes

Nesta seção, a expressão do modelo para o momento de primeira ordem do vetor de coeficientes do filtro adaptativo é derivada. Então, tomando o valor esperado em ambos os lados de (9), tem-se

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu E[\hat{e}_f(n) \mathbf{x}_f(n)]. \quad (21)$$

O termo $\hat{e}_f(n) \mathbf{x}_f(n)$ é obtido pós-multiplicando ambos os lados de (20) por $\mathbf{x}_f(n)$. Então, substituindo (1) em todos os termos do lado direito da expressão resultante, tomando o valor esperado e de acordo com A1, obtém-se

$$\begin{aligned} E[\hat{e}_f(n) \mathbf{x}_f(n)] &= \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ &\quad - \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{w}(n-j)] \\ &\quad - \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{a}_j(n)]. \end{aligned} \quad (22)$$

A matriz de autocorrelação \mathbf{R}_{j-i} em (22) é obtida de acordo

com a forma geral $\mathbf{R}_{\beta-\alpha} = E[\mathbf{x}(n-\alpha) \mathbf{x}^T(n-\beta)]$.

Agora, substituindo (22) em (21), obtém-se

$$\begin{aligned} E[\mathbf{w}(n+1)] &= E[\mathbf{w}(n)] + \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ &\quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{w}(n-j)] \\ &\quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{a}_j(n)]. \end{aligned} \quad (23)$$

Em (23), o termo $E[\mathbf{a}_j(n)]$ é determinado tomando o valor esperado em ambos os lados de (19). Assim,

$$E[\mathbf{a}_j(n+1)] = \begin{cases} \mu E[\hat{e}_f(n)\mathbf{x}_f(n)], & j=1 \\ E[\mathbf{a}_{j-1}(n)] + \mu E[\hat{e}_f(n)\mathbf{x}_f(n)], & 2 \leq j \leq M_3 - 1. \end{cases} \quad (24)$$

B. Valor de Regime Permanente de $\mathbf{w}(n)$

Nesta seção, o valor de regime permanente do vetor de coeficientes é obtido. Assim, tomando o limite quando $n \rightarrow \infty$ em (23), tem-se

$$\begin{aligned} & \lim_{n \rightarrow \infty} E[\mathbf{w}(n+1)] \\ &= \lim_{n \rightarrow \infty} E[\mathbf{w}(n)] + \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \lim_{n \rightarrow \infty} E[\mathbf{w}(n-j)] \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \lim_{n \rightarrow \infty} E[\mathbf{a}_j(n)]. \end{aligned} \quad (25)$$

Agora, assumindo que o algoritmo converge, o valor de regime permanente do vetor de coeficientes é obtido a partir da seguinte condição:

$$\lim_{n \rightarrow \infty} E[\mathbf{w}(n+1)] = \lim_{n \rightarrow \infty} E[\mathbf{w}(n-j)] = \lim_{n \rightarrow \infty} E[\mathbf{w}(n)] = \mathbf{w}_\infty. \quad (26)$$

Então, aplicando o resultado de (26) em (25) e considerando um problema de identificação de sistema como ilustrado na Fig. 1, obtém-se

$$\begin{aligned} \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \mathbf{w}_\infty &= \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ & \quad - \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \lim_{n \rightarrow \infty} E[\mathbf{a}_j(n)]. \end{aligned} \quad (27)$$

A partir de (18), note que $\lim_{n \rightarrow \infty} \{E[\mathbf{w}^T(n)] - E[\mathbf{w}^T(n-j)]\} = \lim_{n \rightarrow \infty} E[\mathbf{a}_j^T(n)] = 0$, cancelando assim o efeito do termo de compensação $\Lambda_f(n)$. Agora, aplicando esse resultado em (27), o valor de regime permanente do vetor de coeficientes é dado por

$$\mathbf{w}_\infty = \left(\sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \right)^{-1} \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0. \quad (28)$$

C. Vetor de Erro nos Coeficientes

Definindo o vetor de erro nos coeficientes como

$$\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_\infty \quad (29)$$

e usando o resultado de (29) em (23), obtém-se

$$\begin{aligned} E[\mathbf{v}(n+1)] &= E[\mathbf{v}(n)] + \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{v}(n-j)] \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \mathbf{w}_\infty \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{a}_j(n)]. \end{aligned} \quad (30)$$

Agora, utilizando a relação $\sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} \mathbf{w}_\infty =$

$$\begin{aligned} & \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_2-1} \hat{s}_i s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \quad [\text{ver (28)}] \text{ em (30), obtém-se} \\ & E[\mathbf{v}(n+1)] = E[\mathbf{v}(n)] \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=0}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{v}(n-j)] \\ & \quad - \mu \sum_{i=0}^{\hat{M}-1} \sum_{j=1}^{M_3-1} \hat{s}_i s_{3,j} \mathbf{R}_{j-i} E[\mathbf{a}_j(n)]. \end{aligned} \quad (31)$$

D. Curva de Aprendizagem

Para determinar a expressão do modelo para a curva de aprendizagem do algoritmo MFxFeLMS, aplica-se (12) em (10), e após uma simples manipulação algébrica, obtém-se

$$\begin{aligned} \hat{e}_f(n) &= \sum_{i=0}^{M_2-1} s_{2,i} \mathbf{w}_0^T \mathbf{x}(n-i) - \sum_{i=0}^{M_3-1} s_{3,i} \mathbf{w}^T(n) \mathbf{x}(n-i) \\ & \quad + z_f(n). \end{aligned} \quad (32)$$

Agora, elevando ao quadrado (32), tomando o valor esperado em ambos os lados da expressão resultante e de acordo com A1, tem-se

$$\begin{aligned} E[\hat{e}_f^2(n)] &= \mathbf{w}_0^T \sum_{i=0}^{M_2-1} \sum_{j=0}^{M_2-1} s_{2,i} s_{2,j} \mathbf{R}_{j-i} \mathbf{w}_0 \\ & \quad - 2 \mathbf{w}_0^T \sum_{i=0}^{M_2-1} \sum_{j=0}^{M_3-1} s_{2,i} s_{3,j} \mathbf{R}_{j-i} E[\mathbf{w}(n)] \\ & \quad + \sum_{i=0}^{M_3-1} \sum_{j=0}^{M_3-1} s_{3,i} s_{3,j} \text{tr}\{\mathbf{R}_{j-i} E[\mathbf{w}(n) \mathbf{w}^T(n)]\} + \sum_{i=0}^{M_2-1} s_{2,i}^2 \sigma_z^2. \end{aligned} \quad (33)$$

Para finalizar a derivação de (33), deve-se determinar o momento de segunda ordem de $\mathbf{w}(n)$. Assim, visto que o modelo é derivado assumindo a hipótese de adaptação lenta, pode-se então considerar a seguinte aproximação [14]:

$$E[\mathbf{w}(n) \mathbf{w}^T(n)] \cong E[\mathbf{w}(n)] E[\mathbf{w}^T(n)]. \quad (34)$$

IV. RESULTADOS DE SIMULAÇÃO

Nesta seção, dois exemplos são apresentados. O primeiro compara o desempenho do algoritmo FxFeLMS com o algoritmo MFxFeLMS proposto e seu modelo, para sinal de

entrada gaussiano branco, sob duas condições de avaliação: mesma velocidade de convergência e mesmo erro em regime permanente. No segundo exemplo, o modelo proposto é também avaliado para sinal de entrada gaussiano colorido. Para os dois exemplos apresentados, é considerado um problema de identificação. Também, assume-se que a estimativa do filtro no caminho do sinal de entrada é dada por $\hat{s} = s_3 = s_1 * s_2$.

Exemplo 1: Neste exemplo, a planta utilizada é $w_o = [-0,457 -0,086 -0,100 -0,071 0,107 0,231 0,347 -0,429 0,458 -0,429 0,347 0,231 0,107 -0,071 -0,100 -0,086 -0,457]^T$. O sinal de entrada é branco com variância $\sigma_x^2 = 1$, e $s_1 = [0,801 0,535 0,266]^T$, $s_2 = [0,122 0,468 0,623 0,519 0,274 0,169 0,055]^T$ e $\hat{s} = [0,977 0,441 0,783 0,874 0,664 0,421 0,208 0,074 0,014]^T$. Os valores máximos do passo de adaptação μ_{\max} para a estabilidade do algoritmo (determinado experimentalmente) são 0,006 e 0,014 para o FxFeLMS e o MFxFeLMS, respectivamente. Simulações Monte Carlo (MC) são obtidas considerando 400 realizações independentes. A Fig. 2 mostra as curvas de aprendizagem para ambos os algoritmos (FxFeLMS e MFxFeLMS), forçando a mesma velocidade de convergência ($\mu = 0,003$), bem como o modelo proposto (33). Dessas curvas, observa-se que o FxFeLMS apresenta um maior erro quadrático médio (EQM) em regime permanente do que o MFxFeLMS. Por outro lado, forçando agora que ambos os algoritmos tenham o mesmo EQM em regime permanente (ver Fig. 3), o MFxFeLMS exibe uma maior velocidade de convergência do que o FxFeLMS. Este último converge somente após 10^5 iterações. Além disso, nas Figs. 2 e 3, observa-se o bom casamento entre as simulações MC e o modelo proposto para o algoritmo MFxFeLMS usando sinal de entrada branco.

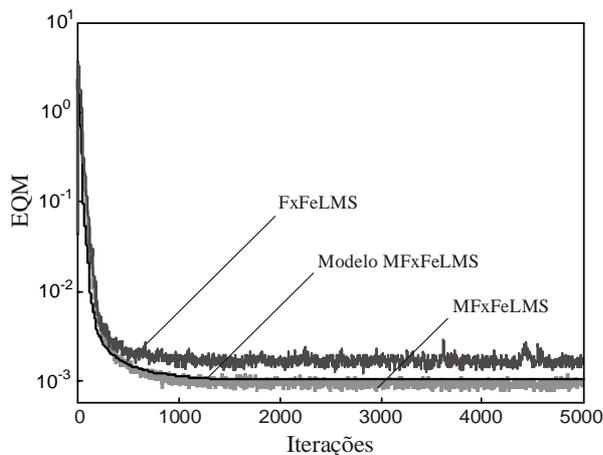


Fig. 2. Exemplo 1. Sinal de entrada branco. Curvas do EQM forçando a mesma velocidade de convergência para os algoritmos FxFeLMS e MFxFeLMS e modelo proposto (33), para $\mu = 0,003$.

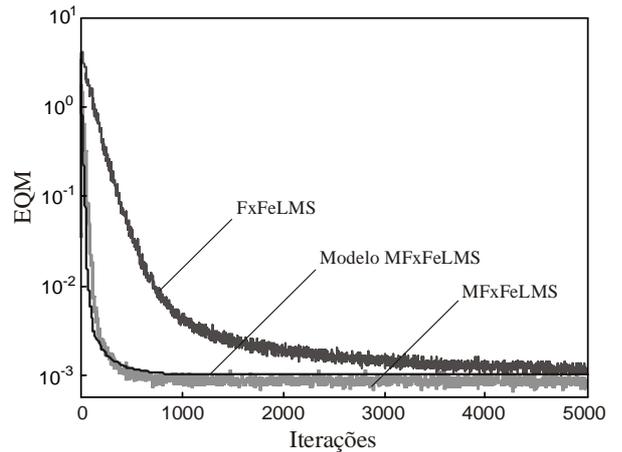


Fig. 3. Exemplo 1. Sinal de entrada branco. Curvas do EQM forçando o mesmo EQM em regime permanente para os algoritmos FxFeLMS e MFxFeLMS, usando, respectivamente, $\mu = 0,0006$ e 0,006, e modelo proposto (33).

Exemplo 2: Neste caso, o sinal de entrada é colorido, obtido através de um processo AR(2), dado por

$$x(n) = a_1 x(n-1) + a_2 x(n-2) + u(n) \quad (35)$$

onde $u(n)$ é um ruído branco com $\sigma_u^2 = 0,39$ e os coeficientes são $a_1 = 0,5833$ e $a_2 = -0,75$, levando a uma dispersão de autovalores da matriz de autocorrelação de entrada igual a 63. A planta e os parâmetros dos filtros são os mesmos utilizados no Exemplo 1. Os valores máximos do passo de adaptação para esse caso são $\mu_{\max} = 0,013$ e 0,023 (determinados experimentalmente) para os algoritmos FxFeLMS e MFxFeLMS, respectivamente. Para esse exemplo, $\mu = 0,0006$ e 0,006 são, respectivamente, usados para o FxFeLMS e o MFxFeLMS. A Fig. 4 mostra, respectivamente, o comportamento médio do vetor de coeficientes e a curvas de aprendizagem (EQM) obtidas por simulação MC e pelo modelo proposto. Dessas figuras, verifica-se uma concordância muito boa entre os resultados de simulação MC e os obtidos pelo modelo proposto para o algoritmo MFxFeLMS. Além do mais, comparando as curvas na Fig. 4(b), nota-se que, para o mesmo EQM em regime permanente, o algoritmo MFxFeLMS apresenta mais rápida convergência.

V. CONCLUSÕES

Neste artigo, o algoritmo LMS com entrada filtrada/erro filtrado modificado (MFxFeLMS) é proposto. Além disso, expressões analíticas para o momento de primeira ordem e curva de aprendizagem foram derivadas considerando adaptação lenta e sem invocar a TI. Através de simulações numéricas, o desempenho do algoritmo proposto foi comparado com o do algoritmo FxFeLMS. Curvas de simulação e do modelo proposto do algoritmo MFxFeLMS também foram mostradas, atestando uma precisão muito boa do modelo obtido.

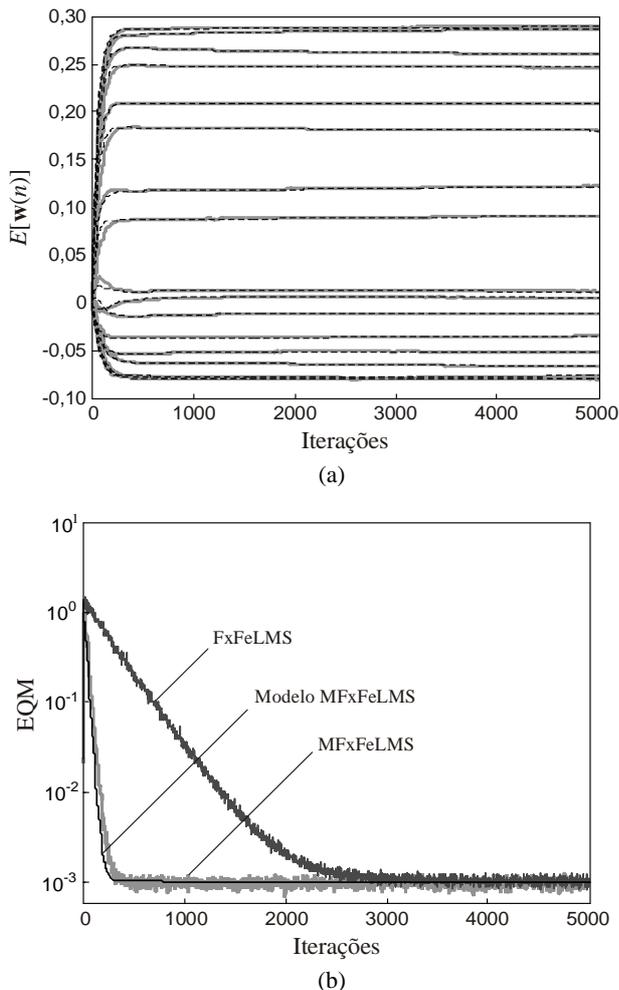


Fig. 4. Exemplo 2. Sinal de entrada colorido. (a) Comportamento médio do vetor de coeficientes do algoritmo MFxFeLMS usando $\mu = 0,006$. (Linhas cinzas) simulação. (Linhas tracejadas escuras) modelo proposto (23). (b) Curvas do EQM: simulação do algoritmo FxFeLMS com $\mu = 0,0006$; simulação do algoritmo MFxFeLMS e modelo proposto (33) com $\mu = 0,006$.

REFERÊNCIAS

- [1] R. D. Poltmann, "Conversion of the delayed LMS algorithm into the LMS algorithm," *IEEE Signal Process. Lett.*, vol. 2, no. 12, pp. 223, Dec. 1995.
- [2] S. C. Douglas, "An efficient implementation of the modified filtered-x LMS algorithm," *IEEE Signal Process. Lett.*, vol. 4, no. 10, pp. 286-288, Oct. 1997.
- [3] S. M. Kuo and D. R. Morgan, *Active Noise control systems: Algorithms and DSP implementations*, New York: John Wiley, 1996.
- [4] S. J. Elliott and P. A. Nelson, "Active noise control," *IEEE Signal Processing Mag.*, vol. 10, no. 4, pp. 12-35, Oct. 1993.
- [5] E. A. Wan, "Adjoint LMS: An efficient alternative to the filtered-x LMS and multiple error LMS algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Atlanta, USA, May 1996, vol. 3, pp. 1842-1845.
- [6] S. Shaffer and C. S. Williams, "The filtered error LMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Boston, USA, Apr. 1983, pp. 41-44.

- [7] J. R. V. Lopez, O. J. Tobias, and R. Seara, "Stochastic model for the modified filtered-error LMS algorithm," in *Proc. 16th European Signal Process. Conf.*, Lausanne, Switzerland, Sep. 2008, pp. 1-5.
- [8] L. Sujbert, "A new filtered LMS algorithm for active noise control," in *Proc. Int. EAA Symp. Active Control of Sound and Vibration*, Fort Lauderdale, Florida, USA, Dec 1999, pp. 1101-1110.
- [9] S. Miyagi and H. Sakai, "Mean-square performance of the filtered-reference/filtered-error LMS algorithm," *IEEE Trans. Circuits Syst.*, vol. 52, no. 5, pp. 2454-2463, Nov. 2005.
- [10] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1998.
- [11] E. Bjarnason, "Analysis of the filtered-X LMS algorithm," *IEEE Trans. Speech and Audio Process.*, vol. 3, no. 6, pp. 504-514, Nov. 1995.
- [12] O. J. Tobias, J. C. M. Bermudez, N. J. Bershad, and R. Seara, "Mean weight behavior of the filtered-X LMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Seattle, USA, May 1998, pp. 3545-3548.
- [13] ———, "Second moment analysis of the filtered-X LMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Phoenix, USA, Mar. 1999, pp. 1873-1876.
- [14] N. J. Bershad, P. Celka, and J. M. Vesin, "Stochastic analysis of gradient adaptive identification of nonlinear systems with memory for Gaussian data and noisy input and output measurements," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 675-689, Mar. 1999.