

Algoritmo LMS com Atraso e Perdas Modificado e Seu Modelo de Predição

Juan R. V. López, Orlando J. Tobias e Rui Seara

Resumo—Este artigo propõe o algoritmo LMS com atraso e perdas modificado (MLDLMS), cujo objetivo é contornar os problemas de instabilidade encontrados no algoritmo LMS com atraso modificado (MDLMS), sob a condição de estimativa imperfeita do atraso. Para o novo algoritmo, é também proposto um modelo para o momento de primeira ordem bem como para a curva de aprendizagem. O modelo é obtido sem invocar a teoria clássica da independência e assumindo a condição de adaptação lenta. Resultados de simulação numérica ratificam a eficácia do novo algoritmo como também de seu modelo de predição para sinais de entrada Gaussianos coloridos.

Palavras-chave—Algoritmo LMS com atraso e perdas modificado, Algoritmo adaptativo, Momento de primeira ordem, Curva de aprendizagem.

Abstract—This paper proposes a modified leaky delayed least-mean-square (MLDLMS) algorithm, aiming to circumvent the instability problems of the modified delayed LMS (MDLMS) algorithm under imperfect system delay estimates. In addition, a model for both the first moment and learning curve of the algorithm is also presented. Such a model is obtained without invoking the classic independence theory and assuming a slow adaptation condition. Numerical simulation results ratify the effectiveness of the new algorithm as well as its prediction model for colored Gaussian inputs.

Keywords—Modified leaky delayed LMS algorithm, Adaptive algorithm, First moment, Learning curve.

I. INTRODUÇÃO

O algoritmo LMS é um dos algoritmos adaptativos mais extensivamente utilizados devido à sua robustez e simplicidade computacional [1], [2]. A vasta variedade de aplicações em que o algoritmo LMS pode ser envolvido origina diversas versões de sua forma convencional. Especificamente, aqui é analisado o caso em que o sinal de erro é disponível para o algoritmo com um certo atraso. Nessa condição, a versão convencional do LMS não opera adequadamente, apresentando uma convergência pobre ou até mesmo instabilidade [3], [4]. Para contornar tal problema, o algoritmo LMS convencional é modificado, resultando no

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).

algoritmo LMS com atraso (DLMS) [5], [6]. Entretanto, esse algoritmo apresenta uma certa degradação nas propriedades de convergência e estabilidade, tornando-se pior à medida que o valor do atraso aumenta. Para superar tal problema, em [7] foi proposta uma modificação do algoritmo DLMS, resultando com que o algoritmo modificado agora tenha as mesmas características de convergência do algoritmo LMS convencional, independente do valor do atraso existente. Tal algoritmo é denominado DLMS modificado (MDLMS) [7], [8]. No entanto, o algoritmo MDLMS tem como inconveniente a divergência, caso a estimativa do atraso não seja perfeitamente determinada. Para mitigar tal problema, neste trabalho, é proposta a inclusão de um fator de perdas no algoritmo MDLMS, dando origem ao algoritmo DLMS com perdas modificado (MLDLMS).

Em nosso conhecimento, na literatura especializada da área, tal algoritmo ainda não foi considerado, muito menos um modelo analítico para ele, que é uma segunda contribuição deste trabalho. Da mesma forma que em [9]-[10], a consideração de análise clássica, invocando a teoria da independência (TI), não é considerada para derivar as expressões do modelo proposto por não ser mais adequada ao problema em questão. Dessa forma, o modelo resultante é bem mais preciso. Além do momento de primeira ordem dos coeficientes do filtro adaptativo, é determinada uma expressão para a curva de aprendizagem, considerando adaptação lenta e sinais de entrada Gaussianos coloridos.

Resultados de simulação numérica ratificam a eficácia do novo algoritmo na condição de estimação imperfeita do atraso. Comparações entre simulações Monte Carlo (MC) e as previsões obtidas pelo modelo proposto são também apresentadas, verificando sua precisão.

II. ALGORITMO DLMS COM PERDAS

A. Descrição do Algoritmo

Nesta seção, o algoritmo DLMS com perdas (LDLMS) é brevemente apresentado. A Fig. 1 ilustra o diagrama de blocos do algoritmo DLMS e a notação utilizada, onde $\mathbf{w}_o = [w_{o,0} \ w_{o,1} \ \dots \ w_{o,N-1}]^T$ e $\mathbf{w}(n) = [w_0(n) \ w_1(n) \ \dots \ w_{N-1}(n)]^T$ representam, respectivamente, o vetor da planta e o do filtro adaptativo. O atraso do sistema é denotado por D e sua estimativa \hat{D} . As variáveis $d(n)$ e $z(n)$ são o sinal desejado (ou primário) e o ruído de medição, respectivamente. O sinal $z(n)$ caracteriza um ruído de medição i.i.d, de média zero com variância σ_z^2 e decorrelacionada com qualquer outro

sinal no sistema. Nesta análise, o vetor de entrada é $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T$, com $\{x(n)\}$ sendo um processo Gaussiano de média zero e variância σ_x^2 . O vetor $\mathbf{x}(n-\hat{D})$ representa o sinal de entrada atrasado pela estimativa de D . Neste trabalho, é assumido que os vetores \mathbf{w}_0 e $\mathbf{w}(n)$ têm igual comprimento. Entretanto, note que D e \hat{D} podem ser diferentes, sendo o caso mais comum em aplicações práticas. A partir da Fig. 1, pode-se escrever o sinal de erro do algoritmo como

$$e(n-D) = d(n-D) - y(n-D) + z(n-D) \quad (1)$$

onde $d(n)$ e $y(n-D)$ são, respectivamente, dados por

$$d(n) = \mathbf{w}_0^T \mathbf{x}(n) = \mathbf{x}^T(n) \mathbf{w}_0 \quad (2)$$

e

$$y(n-D) = \mathbf{w}^T(n-D) \mathbf{x}(n-D) = \mathbf{x}^T(n-D) \mathbf{w}(n-D). \quad (3)$$

Substituindo (3) em (1), obtém-se

$$e(n-D) = d(n-D) - \mathbf{w}^T(n-D) \mathbf{x}(n-D) + z(n). \quad (4)$$

Finalmente, a expressão de atualização dos coeficientes do algoritmo LDLMS é dada por [9]

$$\mathbf{w}(n+1) = \nu \mathbf{w}(n) + \mu e(n-D) \mathbf{x}(n-\hat{D}) \quad (5)$$

onde $\nu = 1 - \mu\gamma$ com γ representando o fator de perdas. Visto que μ é sempre positivo e $0 \leq \mu\gamma \leq 1$, então $0 \leq \nu \leq 1$.

B. Algoritmo LDLMS Modificado

O algoritmo LDLMS modificado (MLDLMS) é obtido pela inclusão de um termo de compensação $\Lambda(n)$ no erro instantâneo [equação (4)]. Dessa forma, o sinal de erro compensado $\hat{e}(n-D)$ é expresso como [7]

$$\hat{e}(n-D) = d(n-D) - \mathbf{w}^T(n-D) \mathbf{x}(n-D) - \Lambda(n) + z(n) \quad (6)$$

e a expressão de atualização dos coeficientes, dada por

$$\mathbf{w}(n+1) = \nu \mathbf{w}(n) + \mu \hat{e}(n-D) \mathbf{x}(n-\hat{D}). \quad (7)$$

O termo $\Lambda(n)$ é determinado forçando que (6) tenha a mesma característica que o sinal de erro do LMS convencional [1], o qual é dado por

$$e(n)|_{\text{LMS}} = d(n-\hat{D}) - \mathbf{w}^T(n) \mathbf{x}(n-\hat{D}) + z(n). \quad (8)$$

Na literatura [8], tal compensação assume uma estimativa perfeita do atraso do sistema. Entretanto, aqui considera-se uma condição mais real, ou seja, uma estimativa imperfeita. Então, igualando (6) e (8), obtém-se o termo de compensação

$$\Lambda(n) = d(n-D) - d(n-\hat{D}) - \mathbf{w}^T(n-D) \mathbf{x}(n-D) + \mathbf{w}^T(n) \mathbf{x}(n-\hat{D}). \quad (9)$$

Visto que uma estimativa do atraso pode ser obtida, a seguinte aproximação é considerada $\mathbf{w}^T(n-D) \mathbf{x}(n-D) \approx \mathbf{w}^T(n-\hat{D}) \mathbf{x}(n-\hat{D})$. Assim,

$$\Lambda(n) = \varepsilon_{\hat{D}-D}(n) + [\mathbf{w}^T(n) - \mathbf{w}^T(n-\hat{D})] \mathbf{x}(n-\hat{D}) \quad (10)$$

e

$$\varepsilon_{\hat{D}-D}(n) = d(n-D) - d(n-\hat{D}). \quad (11)$$

De (7), o termo $[\mathbf{w}^T(n) - \mathbf{w}^T(n-\hat{D})]$ é obtido, permitindo então que (10) seja reescrita como

$$\Lambda(n) = \mu \sum_{j=1}^{\hat{D}} \nu^{j-1} \hat{e}(n-\hat{D}-j) \mathbf{x}^T(n-\hat{D}-j) \mathbf{x}(n-\hat{D}) + \varepsilon_{\hat{D}-D}(n) + (\nu^{\hat{D}} - 1) \mathbf{w}^T(n-\hat{D}) \mathbf{x}(n-\hat{D}). \quad (12)$$

Substituindo (11) em (12) e utilizando (6), o sinal de erro compensado é

$$\begin{aligned} \hat{e}(n-D) &= d(n-\hat{D}) - \mathbf{w}^T(n-D) \mathbf{x}(n-D) \\ &- \mu \sum_{j=1}^{\hat{D}} \nu^{j-1} \hat{e}(n-\hat{D}-j) \mathbf{x}^T(n-\hat{D}-j) \mathbf{x}(n-\hat{D}) \\ &- (\nu^{\hat{D}} - 1) \mathbf{w}^T(n-\hat{D}) \mathbf{x}(n-\hat{D}). \end{aligned} \quad (13)$$

Note que em (13) $\hat{e}(n-D)$ depende de seus valores passados, sendo que tal característica não se encontra presente no DLMS convencional.

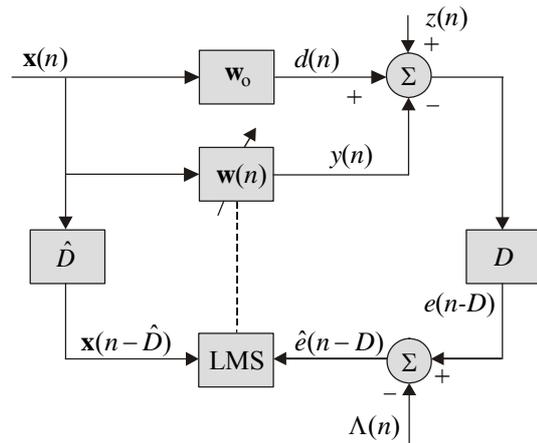


Fig. 1. Diagrama de blocos do algoritmo DLMS.

III. MODELAGEM

A. Considerações de Análise

Para a obtenção das expressões do modelo, são utilizadas as seguintes considerações:

- i) As correlações entre diferentes vetores de entrada são muito mais importantes do que as correlações entre vetores de entrada e vetores de coeficientes [9]-[10].
- ii) O modelo proposto é derivado utilizando-se a hipótese de adaptação lenta, sendo então desconsiderados os termos contendo μ^β com $\beta \geq 3$.
- iii) Visto que o sinal de entrada é Gaussiano, o momento de quarta ordem é determinado utilizando-se o Teorema de Fatoração de Momentos para sinais Gaussianos [11].

B. Comportamento Médio do Vetor de Coeficientes

Nesta seção, uma expressão para o modelo do comportamento médio do vetor de coeficientes é derivada. Então, substituindo (13) em (7) e tomando o valor esperado em ambos os lados da expressão resultante, obtém-se

$$\begin{aligned}
 E[\mathbf{w}(n+1)] = & \\
 & vE[\mathbf{w}(n)] + \mu E[\mathbf{x}(n-\hat{D})d(n-\hat{D})] \\
 & - \mu E[\mathbf{x}(n-\hat{D})\mathbf{x}^T(n-D)\mathbf{w}(n-D)] \\
 & - \mu^2 E[\mathbf{x}(n-\hat{D}) \sum_{j=1}^{\hat{D}} v^{j-1} \hat{e}(n-\hat{D}-j)\mathbf{x}^T(n-\hat{D}-j)\mathbf{x}(n-\hat{D})] \\
 & - \mu(v^{\hat{D}}-1)E[\mathbf{x}(n-\hat{D})\mathbf{x}^T(n-\hat{D})\mathbf{w}(n-\hat{D})] \\
 & + \mu E[\mathbf{x}(n-\hat{D})z(n)].
 \end{aligned} \tag{14}$$

Determinando $d(n)$ e $\hat{e}(n-\hat{D}-j)$ através de (2) e (13), respectivamente, e substituindo-os em (14), usando (i)-(iii) e realizando as devidas manipulações matemáticas, tem-se

$$\begin{aligned}
 E[\mathbf{w}(n+1)] = & vE[\mathbf{w}(n)] + \mu \mathbf{p}_0 - \mu \mathbf{R}_{D-\hat{D}} E[\mathbf{w}(n-D)] \\
 & + \mu(1-v^{\hat{D}})\mathbf{R}_{\hat{D}-\hat{D}} E[\mathbf{w}(n-\hat{D})] \\
 & - \mu^2 \sum_{j=1}^{\hat{D}} v^{j-1} \{ \mathbf{R}^2 + \mathbf{R}_j^2 + \mathbf{R}_j \text{tr}[\mathbf{R}_j] \} \\
 & \times \{ \mathbf{w}_0 - (1-v^{\hat{D}})E[\mathbf{w}(n-\hat{D}-j)] \} \\
 & + \mu^2 \sum_{j=1}^{\hat{D}} v^{j-1} \{ \mathbf{R}_j \mathbf{R}_{D-\hat{D}+j} + \mathbf{R} \mathbf{R}_{D-\hat{D}} + \mathbf{R}_{D-\hat{D}+j} \text{tr}[\mathbf{R}_j] \} \\
 & \times E[\mathbf{w}(n-D-j)]
 \end{aligned} \tag{15}$$

onde $\mathbf{p}_0 = E[\mathbf{x}(n-\hat{D})d(n-\hat{D})]$. As matrizes de autocorrelação envolvendo o sinal de entrada em (15) são obtidas a partir da forma geral $\mathbf{R}_{\beta-\alpha} = E[\mathbf{x}(n-\alpha)\mathbf{x}^T(n-\beta)]$.

C. Vetor de Coeficientes em Regime Permanente

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

$$\begin{aligned}
 \lim_{n \rightarrow \infty} E[\mathbf{w}(n+1)] &= \lim_{n \rightarrow \infty} E[\mathbf{w}(n)] = \\
 \lim_{n \rightarrow \infty} E[\mathbf{w}(n-D)] &= \lim_{n \rightarrow \infty} E[\mathbf{w}(n-\hat{D}-j)] \\
 \lim_{n \rightarrow \infty} E[\mathbf{w}(n-D-j)] &= \mathbf{w}_\infty.
 \end{aligned} \tag{16}$$

Substituindo (16) em (15) e considerando adaptação lenta, obtém-se

$$\mathbf{w}_\infty = [\mathbf{R}_{D-\hat{D}} + \gamma \mathbf{I} + (1-v^{\hat{D}})\mathbf{R}_0]^{-1} \mathbf{p}_0. \tag{17}$$

Para o algoritmo MLDLMS, o efeito do fator de perda é mais evidente quando se considera o caso da estimação imperfeita do atraso no sistema e, em particular, para um sinal de entrada branco. Nesse caso, a matriz $\mathbf{R}_{D-\hat{D}}$ tem somente uma

diagonal diferente de zero (seja a principal ou a secundária, dependendo da diferença $D-\hat{D}$). Por exemplo, para um filtro de dois coeficientes com $\sigma_x^2 = 1$, $D = 1$ e $\hat{D} = 0$, tem-se

$$\mathbf{R}_{D-\hat{D}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}. \tag{18}$$

A matriz $\mathbf{R}_{D-\hat{D}}$ em (18) é singular. Entretanto, adicionando-a o termo $\gamma(\mathbf{I} + \mu \mathbf{R}_0)$, obtém-se

$$\mathbf{R}_{D-\hat{D}} + \gamma(\mathbf{I} + \mu \mathbf{R}_0) = \begin{bmatrix} \gamma(1+\mu) & 1 \\ 0 & \gamma(1+\mu) \end{bmatrix} \tag{19}$$

cuja inversa agora pode ser determinada. Obviamente, o condicionamento dessa matriz depende dos valores de γ e μ . Para o caso de sinal de entrada colorido, a matriz $\mathbf{R}_{D-\hat{D}}$ é não-singular do ponto de vista matemático; porém, quase sempre ela apresenta um pobre condicionamento. Nesse caso, o condicionamento pode ser sempre melhorado pela inclusão do fator de perdas no algoritmo.

D. Vetor de Erro nos Coeficientes

Definindo-se o vetor de erro nos coeficientes como sendo

$$\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_\infty \tag{20}$$

substituindo (20) em (15) e desconsiderando os termos afetados por μ^2 , obtém-se

$$\begin{aligned}
 E[\mathbf{v}(n+1)] = & vE[\mathbf{v}(n)] - \mu \mathbf{R}_{D-\hat{D}} E[\mathbf{v}(n-D)] \\
 & + \mu(1-v^{\hat{D}})\mathbf{R}_{\hat{D}-\hat{D}} E[\mathbf{v}(n-\hat{D})].
 \end{aligned} \tag{21}$$

Neste ponto, pode-se ressaltar melhor a motivação deste trabalho e o objetivo do algoritmo MLDLMS. Considere-se (21) para um caso simples de um filtro adaptativo de dois coeficientes com os seguintes parâmetros: sinal de entrada branco com $\sigma_x^2 = 1$, $D = 0$ e $\hat{D} = 1$. Assim,

$$\mathbf{R}_{D-\hat{D}} = E[\mathbf{x}(n)\mathbf{x}(n-1)] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}. \tag{22}$$

Definindo $E[\mathbf{v}(n)] = [\varepsilon_1(n) \ \varepsilon_2(n)]^T$ e exprimindo (21) em termos dos componentes do vetor; então, as expressões de atualização desses componentes são

$$\begin{cases} \varepsilon_1(n+1) = v\varepsilon_1(n) \\ \varepsilon_2(n+1) = v\varepsilon_2(n) - \mu\varepsilon_1(n-1). \end{cases} \tag{23}$$

Assumindo-se que $\varepsilon_1(0) \neq 0$, $\varepsilon_2(0) \neq 0$, and $|v| < 1$, obtém-se

$$\begin{cases} \varepsilon_1(n) = v^n \varepsilon_1(0) \\ \varepsilon_2(n) = v^n \varepsilon_2(0) - (n-1)\mu v^{n-2} \varepsilon_1(n-1). \end{cases} \tag{24}$$

Note que (24) tende a zero quando $n \rightarrow \infty$ e para $|v| < 1$. Por outro lado, se o fator de perdas é igual a zero então $v = 1$, levando (24) à divergência quando $n \rightarrow \infty$ e $\varepsilon_1(0) \neq 0$.

E. Curva de Aprendizagem e Momento de Segunda Ordem

Para determinar o modelo para a curva de aprendizagem do algoritmo MLDLMS, substitui-se (10) em (6), após simples manipulações algébricas, obtém-se

$$\begin{aligned} \hat{e}(n-D) &= d(n-\hat{D}) - \mathbf{w}^T(n-D)\mathbf{x}(n-D) \\ &\quad - \mathbf{w}^T(n)\mathbf{x}(n-\hat{D}) + \mathbf{w}^T(n-\hat{D})\mathbf{x}(n-\hat{D}) + z(n). \end{aligned} \quad (25)$$

Agora, substituindo (2) em (25), elevando ao quadrado e tomando o valor esperado de ambos os lados da expressão resultante, obtém-se

$$\begin{aligned} E[\hat{e}^2(n)] &= \mathbf{w}_0^T \mathbf{R}_0 \mathbf{w}_0 - 2\text{tr}\{\mathbf{R}_{D-\hat{D}} E[\mathbf{w}(n-D)\mathbf{w}_0^T]\} \\ &\quad + \text{tr}\{\mathbf{R}_{D-\hat{D}} E[\mathbf{w}(n-D)\mathbf{w}^T(n-D)]\} \\ &\quad - 2\text{tr}\{\mathbf{R}_{D-\hat{D}} E[\mathbf{w}(n)\mathbf{w}_0^T]\} + 2\text{tr}\{\mathbf{R}_0 E[\mathbf{w}(n-\hat{D})\mathbf{w}_0^T]\} \\ &\quad + 2\text{tr}\{\mathbf{R}_{\hat{D}-D} E[\mathbf{w}(n)\mathbf{w}^T(n-D)]\} + \text{tr}\{\mathbf{R}_0 E[\mathbf{w}(n)\mathbf{w}^T(n)]\} \\ &\quad - 2\text{tr}\{\mathbf{R}_0 E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n)]\} + \text{tr}\{\mathbf{R}_0 E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n-\hat{D})]\} \\ &\quad - 2\text{tr}\{\mathbf{R}_{\hat{D}-D} E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n-D)]\} + \sigma_z^2. \end{aligned} \quad (26)$$

Usando (i) em (26), e definindo $\xi(n) = E[\hat{e}^2(n)]$ e $\sigma_d^2(n) = \mathbf{w}_0^T \mathbf{R}_0 \mathbf{w}_0$, tem-se

$$\begin{aligned} \xi(n) &= \sigma_d^2(n) - 2\text{tr}\{\mathbf{R}_{D-\hat{D}} \mathbf{k}_{o-D}(n)\} + \text{tr}\{\mathbf{R}_{D-\hat{D}} \mathbf{k}_0(n)\} \\ &\quad - 2\text{tr}\{\mathbf{R}_{D-\hat{D}} \mathbf{k}_o(n)\} + 2\text{tr}\{\mathbf{R}_0 \mathbf{k}_{o-\hat{D}}(n)\} \\ &\quad + 2\text{tr}\{\mathbf{R}_{\hat{D}-D} \mathbf{k}_D(n)\} - 2\text{tr}\{\mathbf{R}_{\hat{D}-D} \mathbf{k}_{D-\hat{D}}(n)\} \\ &\quad + 2\text{tr}\{\mathbf{R}_0 \mathbf{k}_0(n)\} - 2\text{tr}\{\mathbf{R}_0 \mathbf{k}_{\hat{D}}^T(n)\} + \sigma_z^2 \end{aligned} \quad (27)$$

com

$$\begin{aligned} \mathbf{k}_0(n) &= E[\mathbf{w}(n-D)\mathbf{w}^T(n-D)] = E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n-\hat{D})] = E[\mathbf{w}(n)\mathbf{w}^T(n)] \\ \mathbf{k}_{o-D}(n) &= E[\mathbf{w}(n-D)\mathbf{w}_0^T] \\ \mathbf{k}_o(n) &= E[\mathbf{w}(n)\mathbf{w}_0^T] \\ \mathbf{k}_{o-\hat{D}}(n) &= E[\mathbf{w}(n-\hat{D})\mathbf{w}_0^T] \\ \mathbf{k}_D(n) &= E[\mathbf{w}(n)\mathbf{w}^T(n-D)] \\ \mathbf{k}_{\hat{D}}^T(n) &= E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n)] \\ \mathbf{k}_{D-\hat{D}}(n) &= E[\mathbf{w}(n-\hat{D})\mathbf{w}^T(n-D)]. \end{aligned}$$

Para finalizar a derivação de (27), deve-se determinar o momento de segunda ordem do vetor $\mathbf{w}(n)$. Para tal, utiliza-se a hipótese de adaptação lenta; dessa forma, o momento de segunda ordem é determinado a partir da aproximação $E[\mathbf{v}(n)\mathbf{v}^T(n)] \cong E[\mathbf{v}(n)]E[\mathbf{v}^T(n)]$ [12], que é dependente do momento de primeira ordem, já determinado.

IV. RESULTADOS DE SIMULAÇÃO

Nesta seção, os resultados do algoritmo proposto e seu respectivo modelo são avaliados. Para tal, consideramos um problema de identificação de sistemas. Dois exemplos são apresentados.

A. Exemplo 1

Para este exemplo, a planta considerada é dada pelo vetor $\mathbf{w}_0 = [-1,039 \ -1,892 \ -2,162 \ -1,559 \ 2,339 \ 5,046 \ 7,568 \ 9,355 \ 1,000 \ 9,355 \ 7,568 \ 5,046 \ 2,339 \ -1,559 \ -2,162 \ -1,892 \ -1,039]^T$, o valor do atraso no sistema é $D = \hat{D} = 7$ (estimativa perfeita) e o sinal de entrada é um sinal colorido obtido através de um processo AR(2), dado por

$$x(n) = a_1 x(n-1) + a_2 x(n-2) + u(n) \quad (28)$$

onde $u(n)$ é um ruído branco com variância unitária. Os coeficientes do processo AR(2) são $a_1 = 0,58$ e $a_2 = -0,75$, produzindo uma dispersão dos autovalores da matriz de autocorrelação de entrada igual a 63,1. O máximo valor do passo de adaptação, para o qual o algoritmo MLDLMS ainda converge, é $\mu_{\max} = 0,02$. O valor do passo utilizado neste exemplo é $\mu = 0,005$. Simulações Monte Carlo (MC) são obtidas considerando-se 200 realizações independentes. A Fig. 2 mostra as curvas do erro quadrático médio (EQM) obtidas por simulação MC para o algoritmo MLDLMS com $\gamma = 0$ (algoritmo MDLMS convencional), $\gamma = 0,01$ e $\gamma = 0,2$. Dessa figura, nota-se um aumento no valor do erro de regime permanente à medida que o valor de γ aumenta. No entanto, este é o ônus para se obter uma melhora na convergência do algoritmo, quando utilizada alguma perda ($\gamma \neq 0$).

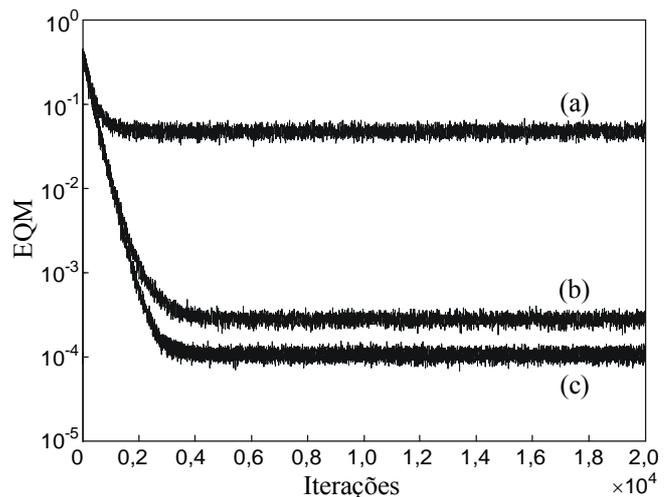


Fig. 2. Exemplo 2. Curvas do EQM do algoritmo MLDLMS para sinal de entrada colorido com $D = 7$, $\hat{D} = 7$ e $\mu = 0,005$. (a) $\gamma = 0,2$. (b) $\gamma = 0,01$. (c) $\gamma = 0$.

B. Exemplo 2

Neste exemplo, ilustram-se comparações dos resultados obtidos por simulação MC (200 realizações independentes) e

pelo modelo proposto considerando $D = 7$, $\hat{D} = 6$ e $\gamma = 0,2$. A planta e o modelo AR(2) são os mesmos do Exemplo 1. O máximo valor do passo de adaptação (determinado experimentalmente), para o qual o algoritmo converge, é $\mu_{\max} = 0,02$. Os resultados numéricos, mostrados nas Figs. 3 e 4, são determinados usando $\mu = 0,005$ e $\mu = 0,008$, respectivamente. As Figs. 3(a) e 4(a) mostram a evolução do comportamento médio do vetor de coeficientes. Dessas figuras, observa-se o bom casamento entre simulação e modelo proposto. As Figs. 3(b) e 4(b) mostram as curvas de EQM, indicando, novamente, um bom casamento entre simulação e modelo. É importante ressaltar que no caso de estimativa imperfeita do atraso, o algoritmo MDLMS convencional (algoritmo MLDLMS com $\gamma = 0$) diverge pelos motivos considerados na Seção III-C. Note que esse não é o caso do algoritmo MLDLMS, quando usado um fator de perdas diferente de zero.

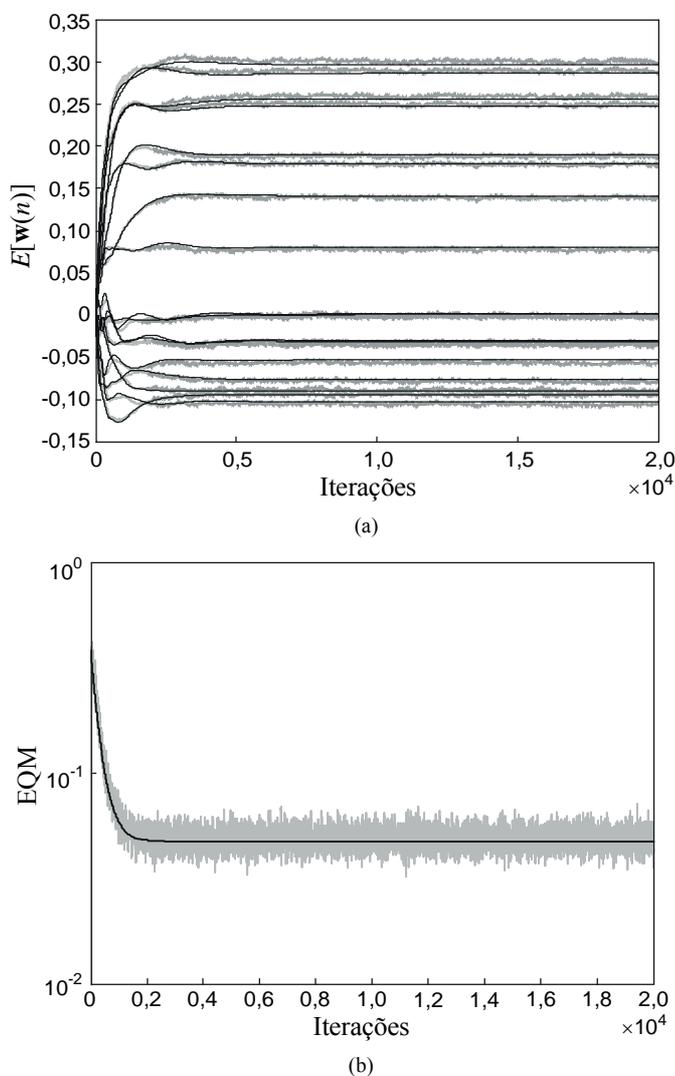


Fig. 3. Exemplo 2. Sinal de entrada colorido para $D = 7$, $\hat{D} = 6$, $\mu = 0,005$ e $\gamma = 0,2$. (a) Evolução de $E[\mathbf{w}(n)]$: simulação MC (linha irregular cinza), modelo proposto (linha contínua preta). (b) Curvas do EQM: simulação MC (linha irregular cinza), modelo proposto (linha contínua preta).

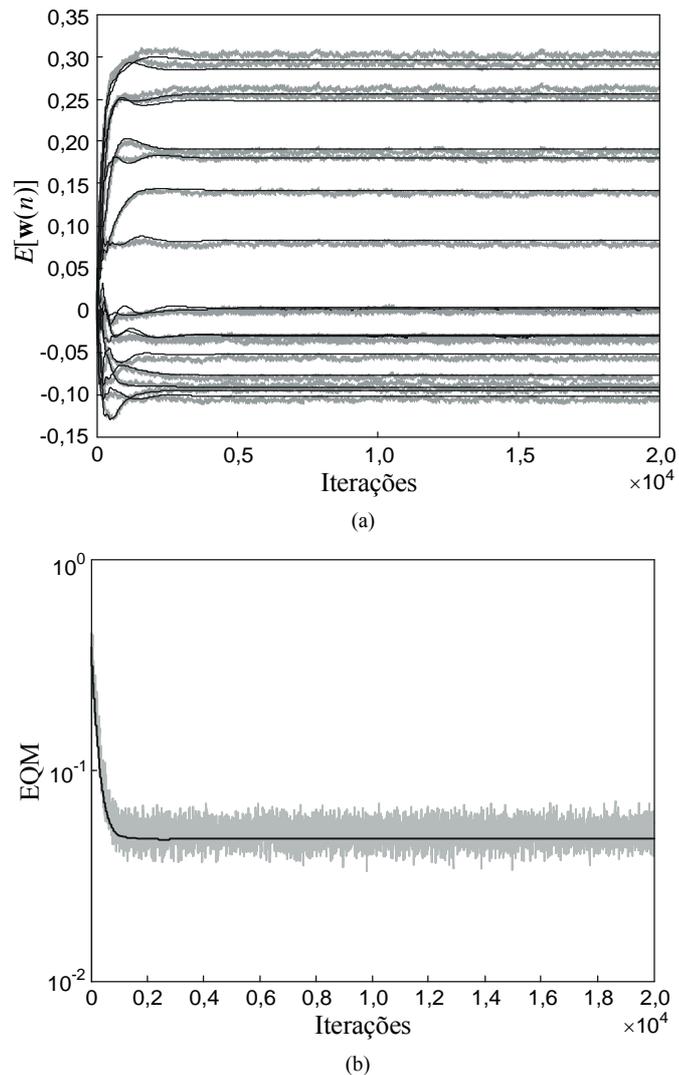


Fig. 4. Exemplo 2. Sinal de entrada colorido para $D = 7$, $\hat{D} = 6$, $\mu = 0,008$ e $\gamma = 0,2$. (a) Evolução de $E[\mathbf{w}(n)]$: simulação MC (linha irregular cinza), modelo proposto (linha contínua preta). (b) Curvas do EQM: simulação MC (linha irregular cinza), modelo proposto (linha contínua preta).

V. CONCLUSÕES

Neste artigo, foi proposto o algoritmo DLMS com perdas modificado para contornar os problemas de estabilidade sob a condição de estimativa imperfeita do atraso do sistema. Expressões analíticas para o momento de primeira ordem e da curva de aprendizagem para o algoritmo MLDLMS foram também derivadas. Tal modelo foi obtido sem invocar a teoria clássica da independência e considerando a hipótese de adaptação lenta. Simulações numéricas verificaram a efetividade do novo algoritmo como também a precisão de seu modelo de predição para sinais de entrada Gaussianos coloridos.

REFERÊNCIAS

- [1] B. Widrow and S. D. Stearns. *Adaptive Signal Processing*, Prentice Hall, 1998.
- [2] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 2001.

- [3] D. D. Falconer, "Adaptive reference echo cancellation," *IEEE Trans. Communications*, vol. 30, no. 9, pp. 2083-2094, Sep. 1982.
- [4] J. G. Proakis, *Digital Communication*, 2nd ed., New York: McGraw-Hill, 1989.
- [5] G. Long, F. Ling, and J. Proakis, "The LMS with delayed coefficient adaptation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 9, pp. 1397-1405, Sep. 1989.
- [6] —, "Corrections to the LMS with delayed coefficient adaptation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 40, no. 1, pp. 230-232, Jan. 1992.
- [7] 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.
- [8] M. Rupp, "Saving complexity of modified filtered-X LMS and delayed update LMS algorithms," *IEEE Trans. Circuits Syst. II: Anal. Dig. Signal Process.*, vol. 44, no. 1, pp. 45-48, Jan. 1997.
- [9] O. J. Tobias, and R. Seara, "Leaky delayed LMS algorithm: analysis for Gaussian data and delay modeling error," *IEEE Trans. On Signal Process.*, Istanbul, Turkey, vol. 52, no. 6, pp. 1596-1606, Jun. 2004.
- [10] 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.
- [11] I. S. Reed, "On a moment theorem for complex Gaussian processes," *IEEE Trans. Inform. Theory*, vol. 8, no. 3, pp. 194-195, Apr. 1962.
- [12] 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 Processing*, vol. 47, no. 3, pp. 675-689, Mar. 1999.