

Análise Estatística de Filtros Volterra Adaptados pelo Algoritmo LMS

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

Resumo—Este trabalho apresenta uma análise estatística do filtro Volterra adaptativo usando o algoritmo LMS no processo de adaptação. Expressões para os momentos de primeira e segunda ordens são obtidas considerando a operação de filtragem Volterra pseudo-linear, representada pelo produto interno de vetores. Resultados de simulação são apresentados atestando a precisão das expressões analíticas obtidas.

Palavras-Chave—Algoritmo LMS, análise estatística, filtragem adaptativa, filtro Volterra.

Abstract—This paper presents a statistical analysis of adaptive Volterra filters using the LMS algorithm for adaptation. Expressions for the first and second order moments are derived, considering the pseudo-linear Volterra filtering operation as a vector inner product. Simulation results confirm the accuracy of the obtained analytical expressions.

Index Terms—LMS algorithm, statistical analysis, adaptive filtering, Volterra filter.

I. INTRODUÇÃO

Nos últimos anos, os filtros adaptativos não-lineares têm expandido seu âmbito de aplicações, particularmente devido ao aumento de capacidade dos DSPs (*digital signal processor*) atualmente disponíveis. Conseqüentemente, o interesse por pesquisas nesta área também tem aumentado consideravelmente. Neste contexto, os filtros Volterra adaptativos se tornaram uma opção interessante para diversas aplicações. Esses filtros têm sido utilizados para controle ativo de processos não-lineares [1], cancelamento de eco acústico [2], identificação e redução de distorções em sistemas de auto-falante [3], equalização de canais de satélites [4], dentre outras aplicações.

A principal vantagem dos filtros Volterra reside no fato de que, analogamente aos filtros lineares, a operação de filtragem pode ser representada pelo produto interno de dois vetores, um deles contendo os coeficientes do filtro e o outro os produtos cruzados das amostras do sinal de entrada. Assim, sua versão adaptativa pode utilizar-se de algoritmos comumente aplicados aos casos de filtragem linear como, por exemplo, o algoritmo LMS. Devido à sua simplicidade e robustez, o algoritmo LMS tem sido aplicado com sucesso para os mais diferentes problemas, tornando-se também uma

boa opção para a implementação de filtros Volterra adaptativos.

Na literatura, são encontrados diversos trabalhos dedicados à análise do filtro Volterra adaptativo. A maioria deles é orientada a problemas de convergência e estabilidade do algoritmo [4]-[6], sendo de grande interesse para avaliar seu desempenho e suas limitações. Outros trabalhos são focados especificamente em filtros que empregam procedimentos de ortogonalização do sinal de entrada [7] ou mesmo filtros com restrição de ordem [5]. Entretanto, nenhum deles apresenta uma análise dos momentos de primeira e segunda ordens para os filtros Volterra adaptados pelo algoritmo LMS.

Neste trabalho, modelos analíticos para os momentos de primeira e segunda ordens do vetor de coeficientes do filtro Volterra adaptado pelo algoritmo LMS são desenvolvidos. A Seção II apresenta uma breve descrição do filtro Volterra, ressaltando a representação pseudo-linear de sua operação de filtragem. Na Seção III, o vetor ótimo de pesos é obtido. Nas Seções IV e V, expressões analíticas para o momento de primeira ordem são derivadas. A Seção VI discute as particularidades da matriz de autocorrelação do sinal de entrada para o filtro Volterra. A Seção VII define o erro quadrático médio, enquanto a Seção VIII discute uma aproximação para o momento de segunda ordem. Na Seção IX, resultados de simulação são apresentados, atestando a validade e a precisão das expressões analíticas obtidas. Finalmente, a Seção X apresenta as conclusões do presente trabalho.

II. FILTRO VOLTERRA

Um filtro Volterra causal e discreto pode ser considerado como um filtro linear acrescido de blocos não-lineares. Essa característica fica bastante evidente a partir de sua relação de entrada e saída [8], dada por

$$\begin{aligned}
 y(n) = & \sum_{m_1=0}^N h_1(m_1)x(n-m_1) \\
 & + \sum_{m_1=0}^N \sum_{m_2=m_1}^N h_2(m_1, m_2)x(n-m_1)x(n-m_2) + \dots \\
 & + \sum_{m_1=0}^N \dots \sum_{m_p=m_{p-1}}^N h_p(m_1, \dots, m_p)x(n-m_1) \dots x(n-m_p),
 \end{aligned} \quad (1)$$

onde $x(n)$ e $y(n)$ denotam o sinal de entrada e o sinal de saída, respectivamente; $h_p(m_1, \dots, m_p)$ representa os coeficientes de ordem p ; N é o tamanho da memória; e P é a ordem do filtro. É importante ressaltar que os elementos

Eduardo L. O. Batista, Orlando J. Tobias e Rui Seara, LINSE - Laboratório de Circuitos e Processamento de Sinais, Departamento de Engenharia Elétrica, Universidade Federal de Santa Catarina, Florianópolis, SC, E-mails: {dudu, orlando, seara}@linse.ufsc.br.

Este trabalho foi parcialmente financiado pela CAPES e pelo CNPq.

redundantes, sem perda de generalidade [1], são removidos de (1). Tem-se, então, um filtro composto por um bloco de primeira ordem (linear), cujos coeficientes são representados por h_1 , seguido de blocos não-lineares (h_2, h_3, \dots, h_p). Denotando-se a saída do bloco de ordem p por $y_p(n)$, podemos reescrever (1) como segue:

$$y(n) = \sum_{p=1}^p y_p(n), \quad (2)$$

com $y_p(n)$ dado por

$$y_p(n) = \sum_{m_1=0}^{N-1} \sum_{m_2=m_1}^{N-1} \dots \sum_{m_p=m_{p-1}}^{N-1} h_p(m_1, m_2, \dots, m_p) \prod_{k=1}^p x(n-m_k). \quad (3)$$

A saída de cada um dos blocos do filtro Volterra pode ser escrita na forma vetorial. Essa abordagem é bem conhecida para o bloco de primeira ordem (teoria de filtros lineares), resultando em

$$y_1(n) = \mathbf{h}_1^T \mathbf{x}_1(n), \quad (4)$$

onde $\mathbf{x}_1(n)$ representa o vetor de entrada de primeira ordem e \mathbf{h}_1 o vetor de coeficientes. Para os demais blocos, um procedimento análogo é possível realizando-se uma operação vetorial pseudo-linear. Por exemplo, para um bloco de segunda ordem, pode-se escrever o vetor de entrada e o de coeficientes como

$$\mathbf{x}_2(n) = [x^2(n) \ x(n)x(n-1) \ \dots \ x(n)x(n-N+1) \ x^2(n-1) \ \dots \ x^2(n-N+1)], \quad (5)$$

$$\mathbf{h}_2 = [h_2(0,0) \ h_2(0,1) \ \dots \ h_2(0,N-1) \ h_2(1,1) \ \dots \ h_2(N-1,N-1)]^T, \quad (6)$$

resultando em uma relação de entrada e saída dada por

$$y_2(n) = \mathbf{h}_2^T \mathbf{x}_2(n). \quad (7)$$

A mesma operação pode ser realizada para os outros blocos não-lineares do filtro Volterra, resultando na seguinte relação de entrada e saída para um bloco de uma ordem p qualquer:

$$y_p(n) = \mathbf{h}_p^T \mathbf{x}_p(n). \quad (8)$$

Definindo-se agora o vetor de entrada do filtro Volterra como

$$\mathbf{x}_v(n) = [\mathbf{x}_1^T(n), \mathbf{x}_2^T(n), \dots, \mathbf{x}_p^T(n)]^T, \quad (9)$$

e o vetor de coeficientes por

$$\mathbf{h}_v = [\mathbf{h}_1^T, \mathbf{h}_2^T, \dots, \mathbf{h}_p^T]^T, \quad (10)$$

tem-se a relação de entrada e saída do filtro Volterra (1), reescrita da seguinte forma:

$$y(n) = \mathbf{h}_v^T \mathbf{x}_v(n). \quad (11)$$

Em (11), tem-se a operação de filtragem do filtro Volterra escrita de maneira pseudolinear como o produto interno entre dois vetores. Essa característica é muito importante, permitindo o desenvolvimento das expressões para os momentos de primeira e segunda ordens para o filtro Volterra, apresentadas a seguir. Além disso, pode-se notar, nesta expressão, a linearidade entre a saída e os coeficientes do filtro. Tal propriedade permite a aplicação direta de

algoritmos adaptativos, usualmente empregados em filtros lineares, também para os filtros Volterra.

III. VETOR DE COEFICIENTE ÓTIMO

A Figura 1 apresenta o diagrama em blocos de um filtro Volterra no contexto da estimação de um sinal $d(n)$ a partir de um sinal de entrada $x(n)$. Com base nesta figura e considerando (11), o erro instantâneo pode ser escrito como

$$e(n) = d(n) - y(n) = d(n) - \mathbf{h}_v^T \mathbf{x}_v(n). \quad (12)$$

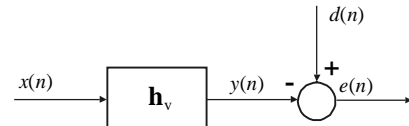


Fig. 1. Diagrama em blocos de um problema de estimação de sinal.

Agora, elevando-se (12) ao quadrado e tomando-se o valor esperado, obtém-se

$$\xi = E[e^2(n)] = E[d^2(n)] - 2E[d(n)\mathbf{x}_v^T(n)]\mathbf{h}_v + \mathbf{h}_v^T E[\mathbf{x}_v(n)\mathbf{x}_v^T(n)]\mathbf{h}_v. \quad (13)$$

Em (13), o vetor \mathbf{h}_v foi retirado do operador esperança por não se tratar de uma variável estatística. Como o objetivo é determinar o vetor de coeficientes que minimize o erro quadrático médio $\xi = E[e^2(n)]$, faz-se seu gradiente em relação ao vetor de coeficientes igual a zero, obtendo-se

$$\nabla_{\mathbf{h}_v} \xi = 0 = -2E[d(n)\mathbf{x}_v(n)] + 2E[\mathbf{x}_v(n)\mathbf{x}_v^T(n)]\mathbf{h}_v. \quad (14)$$

Definindo-se a matriz de autocorrelação do sinal de entrada como $\mathbf{R}_{vv} = E[\mathbf{x}_v(n)\mathbf{x}_v^T(n)]$, e o vetor de correlação entre o sinal desejado $d(n)$ e o sinal de entrada $x(n)$ por $\mathbf{p}_v = E[d(n)\mathbf{x}_v(n)]$, obtém-se, a partir de (14), o vetor de coeficiente ótimo, dado por

$$\mathbf{h}_{v_0} = \mathbf{R}_{vv}^{-1} \mathbf{p}_v. \quad (15)$$

Esta solução é bastante semelhante à solução de Wiener para filtros lineares [9]. No entanto, é importante ressaltar uma principal diferença existente nesta representação. O vetor de entrada $\mathbf{x}_v(n)$, como pode ser observado em (9), é distinto daquele para o caso do filtro linear.

IV. COMPORTAMENTO MÉDIO DOS PESOS

Como mencionado anteriormente, é possível aplicar o bem conhecido algoritmo LMS para efetuar adaptação dos coeficientes do filtro Volterra adaptativo. Assim, a equação de atualização dos pesos pode ser expressa como

$$\mathbf{h}_v(n+1) = \mathbf{h}_v(n) + 2\mu e(n)\mathbf{x}_v(n). \quad (16)$$

A Figura 2 apresenta um diagrama em blocos semelhante ao da Figura 1, porém agora temos um filtro Volterra adaptativo aplicado a um problema de estimação de um sinal $d(n)$ adicionado a um ruído branco $z(n)$ de média zero e decorrelacionado dos demais sinais do sistema.

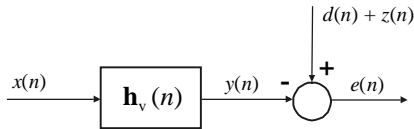


Fig. 2. Diagrama em blocos de um filtro Volterra adaptativo aplicado a um problema de estimação de sinal.

Com base na Figura 2, tem-se o sinal de erro instantâneo dado por

$$e(n) = d(n) + z(n) - y(n) = d(n) + z(n) - \mathbf{h}_v^T(n) \mathbf{x}_v(n). \quad (17)$$

Substituindo-se (17) em (16) e tomando-se o valor esperado, obtém-se

$$E[\mathbf{h}_v(n+1)] = E[\mathbf{h}_v(n)] + 2\mu E[d(n) \mathbf{x}_v(n)] + 2\mu E[z(n) \mathbf{x}_v(n)] - 2\mu E[\mathbf{x}_v(n) \mathbf{h}_v^T(n) \mathbf{x}_v(n)]. \quad (18)$$

Como o ruído $z(n)$ é descorrelacionado dos demais sinais do sistema e possui média nula, temos $E[z(n) \mathbf{x}_v(n)] = 0$. Além disso, a partir da hipótese da independência [9], é possível assumir que a correlação entre os vetores de entrada e o vetor de coeficientes é fraca, resultando em

$$E[\mathbf{x}_v(n) \mathbf{h}_v^T(n) \mathbf{x}_v(n)] \approx E[\mathbf{x}_v(n) \mathbf{x}_v^T(n)] E[\mathbf{h}_v(n)]. \quad (19)$$

Assim, a partir de (18), uma expressão para o comportamento médio dos coeficientes (momento de primeira ordem) do filtro Volterra adaptado pelo algoritmo LMS é obtida. Portanto,

$$E[\mathbf{h}_v(n+1)] = (\mathbf{I} - 2\mu \mathbf{R}_{vv}) E[\mathbf{h}_v(n)] + 2\mu \mathbf{p}_v, \quad (20)$$

onde \mathbf{I} é a matriz identidade.

V. VETOR DE ERRO DOS PESOS

Definindo-se o vetor de erro dos pesos como $\mathbf{v}(n) = \mathbf{h}_v(n) - \mathbf{h}_{v0}$ e substituindo-o em (20), obtém-se

$$E[\mathbf{v}(n+1)] + \mathbf{h}_{v0} = (\mathbf{I} - 2\mu \mathbf{R}_{vv}) E[\mathbf{v}(n)] + \mathbf{h}_{v0} - 2\mu \mathbf{R}_{vv} \mathbf{h}_{v0} + 2\mu \mathbf{p}_v. \quad (21)$$

De (15), tem-se que $\mathbf{R}_{vv} \mathbf{h}_{v0} = \mathbf{p}_v$. Agora, manipulando-se (21), obtém-se o comportamento médio do vetor de erro dos coeficientes, dado por

$$E[\mathbf{v}(n+1)] = (\mathbf{I} - 2\mu \mathbf{R}_{vv}) E[\mathbf{v}(n)]. \quad (22)$$

VI. MATRIZ DE AUTOCORRELAÇÃO

Como mencionado anteriormente, a matriz de autocorrelação do sinal de entrada do filtro Volterra adaptativo é definida por

$$\mathbf{R}_{vv} = E[\mathbf{x}_v(n) \mathbf{x}_v^T(n)]. \quad (23)$$

A partir do vetor de entrada de um filtro Volterra de ordem P , dado por (9), pode-se escrever \mathbf{R}_{vv} da seguinte forma:

$$\mathbf{R}_{vv} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1P} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P1} & \mathbf{R}_{P2} & \cdots & \mathbf{R}_{PP} \end{bmatrix}, \quad (24)$$

onde

$$\mathbf{R}_{p_1 p_2} = \mathbf{R}_{p_2 p_1}^T = E[\mathbf{x}_{p_1}(n) \mathbf{x}_{p_2}^T(n)]. \quad (25)$$

Em (25), $\mathbf{x}_{p_1}(n)$ e $\mathbf{x}_{p_2}(n)$ representam os vetores de entrada dos blocos de ordem p_1 e p_2 , respectivamente.

A matriz \mathbf{R}_{11} é correspondente à matriz de autocorrelação do sinal de entrada de um filtro FIR linear, pois se trata da matriz de autocorrelação de entrada do bloco de primeira ordem do filtro Volterra. Tem-se então a matriz \mathbf{R}_{11} para um filtro Volterra com sinal de entrada branco Gaussiano com variância σ_x^2 , dada por

$$\mathbf{R}_{11} = \begin{bmatrix} \sigma_x^2 & 0 & \cdots & 0 \\ 0 & \sigma_x^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_x^2 \end{bmatrix}. \quad (26)$$

Analisando a expressão do comportamento médio dos pesos, podemos observar que, para filtros lineares com sinal de entrada branco, o comportamento de cada um dos pesos é independente do comportamento dos demais pesos. Isso pode ser constatado escrevendo-se o comportamento médio do i -ésimo peso do filtro a partir de (20) e (26). Assim,

$$E[h_i(n+1)] = (1 - 2\mu \sigma_x^2) E[h_i(n)] + 2\mu E[d(n)x(n-i+1)]. \quad (27)$$

Para um filtro Volterra de segunda ordem, ainda devem ser consideradas as matrizes \mathbf{R}_{12} e \mathbf{R}_{22} na composição total da matriz de autocorrelação do sinal de entrada. Obtendo-se essas matrizes para um sinal de entrada branco Gaussiano com variância σ_x^2 , tem-se

$$\mathbf{R}_{12} = \mathbf{R}_{21}^T = \mathbf{0} \quad (28)$$

e

$$\mathbf{R}_{22} = \begin{bmatrix} 3\sigma_x^2 & 0 & \cdots & 0 & \sigma_x^2 & 0 & \cdots & \sigma_x^2 \\ 0 & \sigma_x^2 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_x^2 & 0 & 0 & \cdots & 0 \\ \sigma_x^2 & 0 & \cdots & 0 & 3\sigma_x^2 & 0 & \cdots & \sigma_x^2 \\ 0 & 0 & \cdots & 0 & 0 & \sigma_x^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma_x^2 & 0 & \cdots & 0 & \sigma_x^2 & 0 & \cdots & 3\sigma_x^2 \end{bmatrix}. \quad (29)$$

Portanto, podemos verificar que existem elementos também fora da diagonal principal da matriz de autocorrelação do sinal de entrada, mesmo para um sinal de entrada branco Gaussiano. Uma consequência disso é uma maior dispersão dos autovalores dessa matriz, degradando a velocidade de convergência do algoritmo LMS [9]. Ainda, podemos observar que, como a matriz \mathbf{R}_{12} é nula, existe um desacoplamento total entre o bloco de primeira e o de segunda ordens. Assim, podemos encontrar limites de estabilidade independentes, levando à escolha de diferentes passos de adaptação para cada bloco, uma vez que tais parâmetros dependem da matriz de autocorrelação do sinal de entrada [9].

Analisando-se filtros Volterra de ordens superiores, podemos observar que a condição de independência entre blocos de diferentes ordens, para sinal de entrada branco, não mais ocorre. Como exemplo, um filtro Volterra adaptativo de terceira ordem com tamanho de memória 3 e sinal de entrada branco, com variância σ_x^2 , tem a matriz \mathbf{R}_{13} dada por

$$\mathbf{R}_{13} = \begin{bmatrix} 3\sigma_x^2 & 0 & 0 & \sigma_x^2 & 0 & \sigma_x^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_x^2 & 0 & 0 & 0 & 0 & 3\sigma_x^2 & 0 & \sigma_x^2 & 0 \\ 0 & 0 & \sigma_x^2 & 0 & 0 & 0 & 0 & \sigma_x^2 & 0 & 3\sigma_x^2 \end{bmatrix}, \quad (30)$$

o que resulta em um acoplamento entre os blocos de primeira e terceira ordens deste filtro.

VII. ERRO QUADRÁTICO MÉDIO

Para determinar uma expressão analítica para a curva de aprendizagem do algoritmo, deve-se inicialmente substituir a definição do vetor de erro dos pesos $\mathbf{v}(n) = \mathbf{h}_v(n) - \mathbf{h}_{v_0}$ em (17), obtendo-se

$$e(n) = d(n) + z(n) - \mathbf{v}^T(n)\mathbf{x}_v(n) - \mathbf{h}_{v_0}^T\mathbf{x}_v(n). \quad (31)$$

Definindo-se o sinal de erro instantâneo ótimo por

$$e_o(n) = d(n) + z(n) - \mathbf{h}_{v_0}^T\mathbf{x}_v(n), \quad (32)$$

pode-se reescrever (31) como

$$e(n) = e_o(n) - \mathbf{v}^T(n)\mathbf{x}_v(n). \quad (33)$$

Elevando-se (33) ao quadrado e tomando-se o valor esperado, obtém-se

$$E[e^2(n)] = E[e_o^2(n)] - E[e_o(n)\mathbf{v}^T(n)\mathbf{x}_v(n)] + E\{[\mathbf{v}^T(n)\mathbf{x}_v(n)]^2\}. \quad (34)$$

Utilizando-se o princípio da ortogonalidade [9] em conjunto com a hipótese da independência, dada por (19), válida também para o vetor de erro dos pesos, tem-se

$$E[e_o(n)\mathbf{v}^T(n)\mathbf{x}_v(n)] = E[\mathbf{v}^T(n)]E[e_o(n)\mathbf{x}_v(n)] = 0. \quad (35)$$

Dado que $E\{[\mathbf{v}^T(n)\mathbf{x}_v(n)]^2\}$ é dimensionalmente um escalar, pode-se então escrever

$$E\{[\mathbf{v}^T(n)\mathbf{x}_v(n)]^2\} = E\{\text{tr}[\mathbf{v}^T(n)\mathbf{R}_{vv}\mathbf{v}(n)]\} = \text{tr}\{E[\mathbf{v}^T(n)\mathbf{v}(n)]\mathbf{R}_{vv}\}. \quad (36)$$

Usando-se a notação $\xi(n) = E[e^2(n)]$, $\xi_{\min} = E[e_o^2(n)]$ e definindo-se a matriz de autocorrelação do vetor de erro dos pesos como $\mathbf{K}(n) = E[\mathbf{v}(n)\mathbf{v}^T(n)]$, a expressão para a curva de aprendizagem é reescrita como

$$\xi(n) = \xi_{\min} + \text{tr}\{\mathbf{K}(n)\mathbf{R}_{vv}\}. \quad (37)$$

VIII. APROXIMAÇÃO DO MOMENTO DE SEGUNDA ORDEM

Para possibilitar a obtenção da curva de aprendizagem, uma expressão recursiva para o momento de segunda ordem dos pesos [matriz $\mathbf{K}(n)$] é requerida. Para tal, através da manipulação de (16), pode-se obter uma expressão recursiva para o vetor de erro dos pesos, dada por

$$\mathbf{v}(n+1) = \mathbf{v}(n) + 2\mu\mathbf{x}_v(n)e_o(n) - 2\mu\mathbf{x}_v(n)\mathbf{x}_v^T(n)\mathbf{v}(n). \quad (38)$$

Multiplicando-se o vetor $\mathbf{v}(n+1)$ por sua versão transposta e tomando-se o valor esperado, obtém-se

$$\begin{aligned} E[\mathbf{v}(n+1)\mathbf{v}^T(n+1)] = & E[\mathbf{v}(n)\mathbf{v}^T(n)] \\ & + 2\mu E[\mathbf{v}(n)e_o(n)\mathbf{x}_v^T(n)] \\ & - 2\mu E[\mathbf{v}(n)\mathbf{v}^T(n)\mathbf{x}_v(n)\mathbf{x}_v^T(n)] \\ & + 2\mu E[e_o(n)\mathbf{x}_v(n)\mathbf{v}^T(n)] \\ & + 4\mu^2 E[\mathbf{x}_v(n)e_o(n)e_o(n)\mathbf{x}_v^T(n)] \\ & - 4\mu^2 E[e_o(n)\mathbf{x}_v(n)\mathbf{v}^T(n)\mathbf{x}_v(n)\mathbf{x}_v^T(n)] \\ & - 2\mu E[\mathbf{x}_v(n)\mathbf{x}_v^T(n)\mathbf{v}(n)\mathbf{v}^T(n)] \\ & - 4\mu^2 E[\mathbf{x}_v(n)\mathbf{x}_v^T(n)\mathbf{v}(n)e_o(n)\mathbf{x}_v^T(n)]. \end{aligned} \quad (39)$$

A resolução de (39) consiste em um trabalho matemático considerável, especialmente devido à existência de elementos dependentes de momentos de ordem elevada do vetor de entrada. No entanto, levando-se em conta que a estabilidade do algoritmo LMS possui forte relação com os autovalores da matriz de autocorrelação do sinal de entrada [9] e, ainda, que a estrutura da matriz de autocorrelação do sinal de entrada do filtro Volterra resulta em autovalores maiores, é possível se inferir que normalmente valores pequenos de μ tornam-se necessários para assegurar a estabilidade do algoritmo. Assim, dessa condição de adaptação lenta, pode-se obter uma aproximação satisfatória para (39), agora desconsiderando-se os termos dependentes de μ^2 . Dessa forma, preserva-se os elementos dominantes como também facilita-se o tratamento matemático. Resolvendo-se os demais termos obtém-se uma aproximação para a expressão recursiva do momento de segunda ordem, dada por

$$\mathbf{K}(n+1) = \mathbf{K}(n) - 2\mu(\mathbf{K}(n)\mathbf{R}_{vv} + \mathbf{R}_{vv}\mathbf{K}(n)). \quad (40)$$

Utilizando-se (40) em conjunto com (37), pode-se obter o comportamento aproximado do erro quadrático médio do filtro Volterra adaptado pelo algoritmo LMS.

IX. RESULTADOS DE SIMULAÇÃO

Para verificar a precisão das expressões obtidas, alguns resultados de simulação são apresentados. Os exemplos desenvolvidos consideram um problema de identificação de sistemas usando sinais de entrada branco e colorido. O ruído de medição $z(n)$ usado é descorrelacionado dos demais sinais do sistema, tendo uma variância $\sigma_z^2 = 0,001$. Os resultados são apresentados para diferentes valores do passo de adaptação, todos relacionados ao valor máximo de μ (μ_{\max}) determinado experimentalmente, para o qual o algoritmo converge. Isso permite verificar as aproximações efetuadas, levando-se em conta diferentes velocidades de convergência.

Exemplo 1: Neste exemplo, um filtro Volterra de segunda ordem e tamanho de memória 7 é usado para identificar uma planta com a seguinte relação de entrada e saída:

$$\begin{aligned} d(n) = & x(n) + 0,5x(n-1) + 0,1x(n-2) - 0,2x(n-3) \\ & - 0,3x(n-4) - 0,1x(n-5) + 0,08x(n-6) + 0,7x^2(n) \\ & + 0,3x(n)x(n-1) - 0,1x(n)x(n-3) + 0,05x(n)x(n-6) \\ & + 0,5x^2(n-1) + 0,2x(n-1)x(n-2) - 0,2x(n-1)x(n-5) \\ & - 0,35x^2(n-2) - 0,05x(n-2)x(n-4) + 0,02x(n-2)x(n-5) \\ & - 0,2x^2(n-3) + 0,05x(n-3)x(n-5) + 0,1x^2(n-4) \\ & + 0,05x(n-4)x(n-5) + 0,02x(n-4)x(n-6) - 0,1x^2(n-5) \\ & + 0,05x^2(n-6). \end{aligned}$$

O sinal de entrada é branco e com variância unitária. O passo de adaptação é igual a $0,5\mu_{\max}$, com $\mu_{\max} = 0,005$. A Figura 3 apresenta o comportamento médio de alguns dos coeficientes obtido por simulação Monte Carlo (média de 100 realizações) comparado com o resultado obtido por (20). Através desta figura, podemos constatar uma precisão muito boa do modelo, exceto por uma pequena diferença existente na fase transitória. Tal diferença é devido à aproximação considerada em (19) (hipótese da independência). A Figura 4 apresenta a curva de aprendizagem obtida com (37) e (40) comparada com a obtida por simulação (média de 100 realizações). Neste caso, uma maior discrepância é verificada devido à aproximação usada na obtenção do momento de segunda ordem. Com a redução do valor do passo de adaptação (condição de adaptação lenta) ambas as diferenças diminuem, como será observado nos próximos exemplos.

Exemplo 2: O mesmo filtro do Exemplo 1 é considerado, modificando-se apenas o valor do passo de adaptação para $0,2\mu_{\max}$. A Figura 5 mostra os resultados de maneira similar aos da Figura 3, enquanto a Fig. 6 ilustra o comportamento do erro quadrático médio seguindo o mesmo padrão da Figura 4. Neste exemplo, podemos constatar a excelente precisão da modelagem desenvolvida, o que pode ser verificado através da similaridade entre as curvas obtidas para uma condição de adaptação lenta.

Exemplo 3: Novamente, o filtro do Exemplo 1 é considerado. Agora, o sinal de entrada é um sinal colorido obtido a partir de um processo auto-regressivo dado por $x(n) = \alpha x(n-1) + \sqrt{1-\alpha^2} u(n)$, onde $u(n)$ é um ruído branco com variância unitária, $\alpha = 0,5$ e dispersão dos autovalores da matriz de autocorrelação do sinal de entrada igual a 113,25. O passo de adaptação é $0,5\mu_{\max}$ com $\mu_{\max} = 0,0015$. Os resultados do comportamento médio dos coeficientes são mostrados na Figura 7, enquanto a curva de aprendizagem é ilustrada na Figura 8. Novamente, ótimos resultados são obtidos, comprovando tanto a validade das expressões analíticas desenvolvidas quanto as aproximações consideradas.

Exemplo 4: Neste exemplo, uma planta Volterra de terceira ordem e tamanho de memória 4 é considerada, possuindo os seguintes vetores de coeficientes:

$$\begin{aligned} \mathbf{h}_1 &= [0,75 \quad 0,3 \quad -0,1 \quad -0,3]^T; \\ \mathbf{h}_2 &= [-0,37 \quad -0,33 \quad -0,28 \quad -0,23 \quad 0,04 \\ & \quad 0,07 \quad 0,09 \quad 0,13 \quad 0,07 \quad 0,06]^T; \\ \mathbf{h}_3 &= [0,27 \quad -0,16 \quad 0,06 \quad -0,01 \quad 0,22 \quad 0,14 \quad -0,02 \\ & \quad -0,25 \quad 0,17 \quad -0,03 \quad 0,2 \quad 0,11 \quad -0,15 \quad -0,04 \\ & \quad 0,2 \quad 0,19 \quad -0,04 \quad 0,17 \quad -0,19 \quad -0,06]^T. \end{aligned}$$

Esta planta é identificada usando um filtro Volterra adaptativo também de terceira ordem e com tamanho de memória 4. O valor do passo de adaptação é $0,25\mu_{\max}$ com $\mu_{\max} = 0,001$, e o sinal de entrada é branco com variância unitária. As Figuras 9 e 10 mostram os resultados obtidos via simulação Monte Carlo (média de 100 realizações) e através das expressões analíticas desenvolvidas. Novamente, observa-se uma concordância muito boa entre as simulações e os correspondentes modelos.

X. CONCLUSÕES

Neste artigo, uma análise estatística do filtro Volterra adaptado pelo algoritmo LMS é apresentada. Essa análise é justificada pelo grande potencial de aplicação que este tipo de filtro não-linear tem demonstrado. O modelo pseudolinear da operação de filtragem Volterra é explorado, permitindo a derivação de expressões analíticas para o seu comportamento. Através dos resultados de simulação, é possível comprovar a validade da modelagem analítica proposta.

REFERÊNCIAS

- [1] L. Tan and J. Jiang, "Adaptive Volterra Filters for Active Control of Nonlinear Noise Processes," *IEEE Trans. on Signal Processing*, vol. 49, no. 8, pp. 1667-1676, Aug. 2001.
- [2] A. Stenger, L. Trautmann, and R. Rabenstein, "Nonlinear Acoustic Echo Cancellation with Second Order Adaptive Volterra Filters," *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. 2, pp. 15-19, Mar. 1999.
- [3] M. Tsujikawa, T. Shiozaki, Y. Kajikawa, and Y. Nomura, "Identification and Elimination of Second-order Nonlinear Distortion of Loudspeaker Systems Using Volterra Filters," *IEEE Int. Symp. on Circ. and Systems*, vol. 5, pp. 28-31, May 2000.
- [4] A. Gutierrez and W. E. Ryan, "Performance of Adaptive Volterra Equalizers on Nonlinear Satellite Channels," *IEEE Int. Conf. on Communications*, vol. 1, pp. 19-22, Jun. 1995.
- [5] P. M. Clarkson and M. V. Dokić, "Stability and Convergence Behavior of Second Order LMS Volterra Filter," *Electronic Letters*, vol. 27, no. 5, pp. 441-443, Feb. 1991.
- [6] J. Chao and A. Inomata, "A Convergence Analysis of Volterra Adaptive Filters," *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 9-12, June 1997.
- [7] T. Ogunfunmi and S. L. Chang, "Second Order Adaptive Volterra System Identification Based on Discrete Nonlinear Wiener Model," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 148, no. 1, pp. 21-29, Feb. 2001.
- [8] V. J. Mathews, "Adaptive Polynomial Filters," *IEEE Signal Processing Magazine*, vol. 8, pp. 10-26, July 1991.
- [9] B. Farhang-Boroujeny, *Adaptive Filters Theory and Applications*, John Wiley & Sons Ltd., 1999.

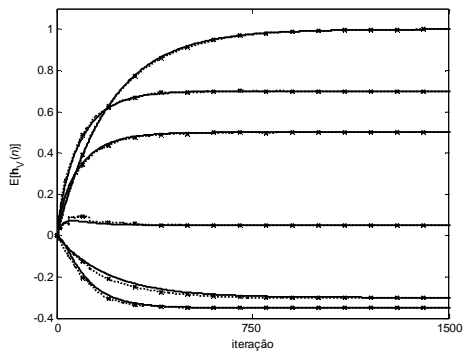


Fig. 3. Exemplo 1. Evolução de $E[h_v(n)]$ com $0,5\mu_{\max}$. (Linha pontilhada com marcadores) simulação Monte Carlo. (Linha sólida) modelo eq. (20).

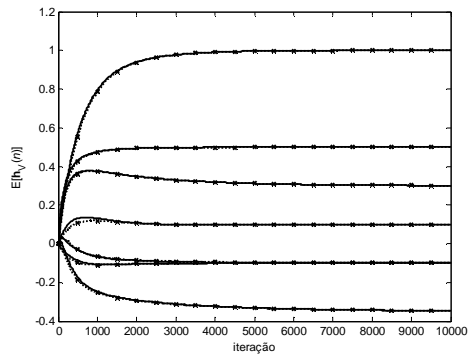


Fig. 7. Exemplo 3. Evolução de $E[h_v(n)]$ com $0,5\mu_{\max}$. (Linha pontilhada com marcadores) simulação. (Linha sólida) modelo eq. (20).

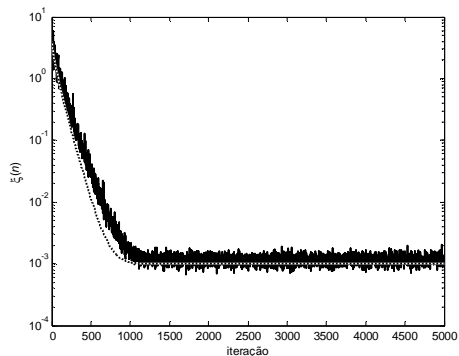


Fig. 4. Exemplo 1. Evolução de $\xi(n)$ com $0,5\mu_{\max}$. (Linha sólida) simulação Monte Carlo. (Linha pontilhada) modelo eqs. (37) e (40).

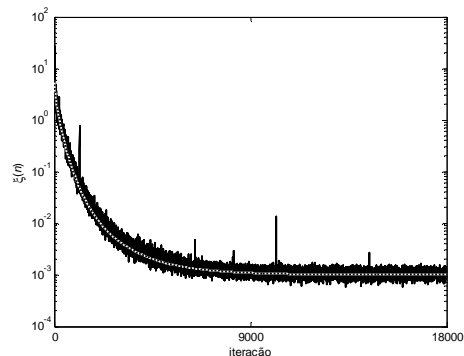


Fig. 8. Exemplo 3. Evolução de $\xi(n)$ com $0,5\mu_{\max}$. (Linha sólida) simulação. (Linha pontilhada) modelo eqs. (37) e (40).

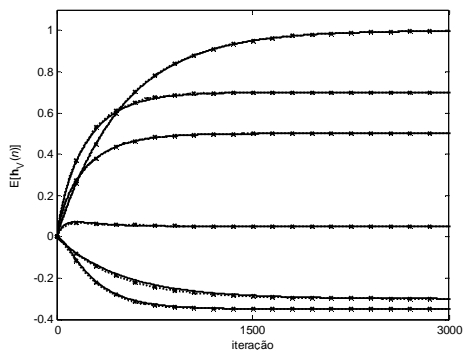


Fig. 5. Exemplo 2. Evolução de $E[h_v(n)]$ com $0,2\mu_{\max}$. (Linha pontilhada com marcadores) simulação. (Linha sólida) modelo eq. (20).

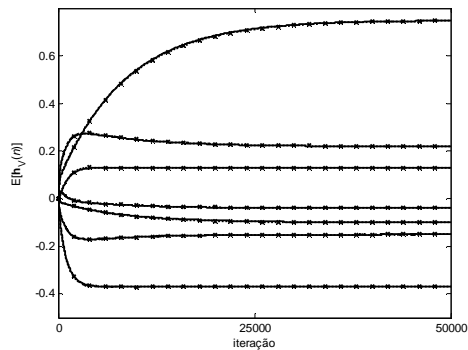


Fig. 9. Exemplo 4. Evolução de $E[h_v(n)]$ com $0,25\mu_{\max}$. (Linha pontilhada com marcadores) simulação. (Linha sólida) modelo eq. (20).

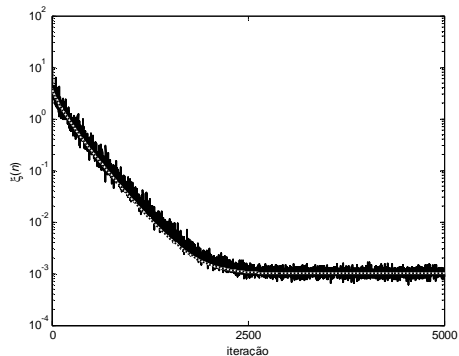


Fig. 6. Exemplo 2. Evolução de $\xi(n)$ com $0,2\mu_{\max}$. (Linha sólida) simulação. (Linha pontilhada) modelo eqs. (37) e (40).

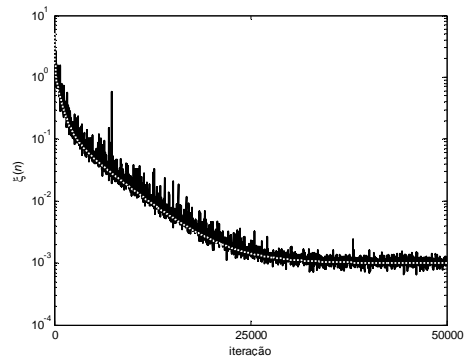


Fig. 10. Exemplo 4. Evolução de $\xi(n)$ com $0,25\mu_{\max}$. (Linha sólida) simulação. (Linha pontilhada) modelo eqs. (37) e (40).