

# AVALIAÇÃO E PROJETO DE ALGORITMOS ADAPTATIVOS DE PASSO VARIÁVEL

CÁSSIO G. LOPES<sup>1</sup>, JOSÉ CARLOS M. BERMUDEZ<sup>2</sup>

Universidade Federal de Santa Catarina  
Laboratório de Pesquisas em Processamento Digital de Sinais - LPDS  
Campus Universitário, Trindade – Fpolis, SC

<sup>1</sup>cassio@eel.ufsc.br

<sup>2</sup>bermudez@fastlane.com.br

## RESUMO

Reunindo indicadores absolutos de desempenho do algoritmo LMS e uma representação apropriada da evolução em passo variável, propõe-se uma metodologia de avaliação e projeto de algoritmos de passo variável. A potencialidade de alguns algoritmos é explorada: quando corretamente projetados, apresentam um desempenho superior aos seus concorrentes mais complexos, mesmo em ambientes não-estacionários.

## 1. INTRODUÇÃO

Desenvolvido na Universidade de Stanford, Califórnia, por B. Widrow e M. E. Hoff Jr. em 1959, o algoritmo LMS representa um dos mais importantes algoritmos adaptativos existentes. Baseado na estimação instantânea do gradiente, assemelha-se ao método do gradiente-descendente, evoluindo estocasticamente em direção ao ponto de erro mínimo da superfície de desempenho. Embora robusto e simples, classicamente o algoritmo LMS é governado por um passo de cálculo *fixo*  $\mu$ , envolvendo uma escolha conflitante entre a velocidade de convergência e o desajuste em regime permanente, o que limita o desempenho do algoritmo [1], [5], [6], [7]. É corrente na literatura que, sob qualquer ponto de vista de otimização, a adaptação é melhor assistida pela utilização de passo variável [4]. Em ambientes estacionários, a escolha de uma seqüência de passos apropriada acelera significativamente a convergência do algoritmo para um dado desajuste em regime permanente. Em ambientes não-estacionários, os algoritmos de passo variável são geralmente mais eficientes no rastreamento da solução ótima [5]. Em contrapartida, a flexibilidade oferecida pela variação do passo exige o emprego de *algoritmos específicos* para a ajuste do passo, o que inspirou a proposição de várias estruturas de adaptação do passo na literatura [6], [7], [8], [9], [10], [11], [12], [13], [14], [15]. As estruturas possuem *parâmetros intrínsecos* de controle, que devem ser projetados para gerarem uma seqüência de passos adequada. As questões que emergem são então: (i) Qual a *seqüência de passos* que deve ser implementada; (ii) Que *conjunto de parâmetros* dos algoritmos gera tal seqüência. O tratamento dessas questões certamente envolve a determinação de indicadores absolutos do limite de desempenho do algoritmo LMS, o que pode ser obtido com o modelo analítico da evolução do erro médio quadrático (EMQ) [1], [2], [16]. A não-utilização de critérios dessa natureza na avaliação dos algoritmos pode levar a conclusões equivocadas acerca de seus respectivos desempenhos [16]. A idéia aqui é utilizar a *evolução ótima teórica* do algoritmo LMS em conjunto com uma representação apropriada da evolução dos algoritmos de passo variável, constituindo *uma metodologia de avaliação dos algoritmos de*

*controle do passo*. Para os casos de identificação de sistemas (Fig. 1), com o sinal de referência  $x(n)$  ruído branco, é também possível projetar os algoritmos de controle e seus parâmetros. Demonstra-se que algoritmos simples [6], considerados de desempenho inferior na literatura [7], [9], [12], na verdade têm sido mal projetados.

## 2. EVOLUÇÃO ÓTIMA DO LMS

Empregando passo variável, a equação de adaptação do algoritmo LMS é dada por [1], [16]:

$$W(n+1) = W(n) + \mu(n)e(n)X(n) \quad (1)$$

O passo de cálculo  $\mu(n)$  deve ser mantido no intervalo  $[\mu_{\min}, \mu_{\max}]$ , visando capacidade de rastreamento em ambientes não-estacionários e estabilidade no caso estacionário [1], [6], [7]. O erro de estimação  $e(n)$  é definido como  $e(n) = d(n) - y(n)$ , onde  $y(n) = W^T(n)X(n)$  é a resposta do filtro adaptativo com coeficientes  $W(n) = [w_0(n), w_1(n), \dots, w_{N-1}(n)]^T$  a um vetor de entradas  $X(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T$ , e  $d(n) = W_o^T X(n) + z(n)$  é o sinal desejado, composto pela resposta do sistema desconhecido  $W_o$  à entrada  $x(n)$  e pelo ruído de medição  $z(n)$  (Fig. 1). A base analítica utilizada para derivação da *evolução ótima do algoritmo LMS* é o modelo analítico do EMQ,  $\xi(n) = E\{e^2(n)\}$ , com passo fixo, válido para sinais gaussianos [1], [2], [16]:

$$\xi(n) = \xi_{\min} + \text{Tr}\{RK(n)\} \quad (2)$$

$$K(n+1) = K(n) - \mu \cdot [RK(n) + K(n)R] + \mu^2 \cdot [R \cdot \text{Tr}\{RK(n)\} + 2RK(n)R + R \cdot \xi_{\min}] \quad (3)$$

onde  $R = E\{X(n)X^T(n)\}$  é a matriz de auto-correlação do sinal  $x(n)$ ,  $K(n) = E\{V(n)V^T(n)\}$  é a matriz de auto-correlação do vetor de erro nos coeficientes  $V(n) = W(n) - W_o$  e  $\xi_{\min}$  é o nível de ruído não-correlacionado. De (2) e (3) é possível determinar uma *seqüência de passos* que leve o algoritmo LMS à *evolução ótima* em ambientes estacionários, operando com passo variável [3], [12], [16]:

$$\mu_{\text{opt}}(n) = \frac{\text{Tr}\{R^2 K(n)\}}{\text{Tr}\{R^2\} \text{Tr}\{RK(n)\} + 2 \text{Tr}\{R^3 K(n)\} + \xi_{\min} \cdot \text{Tr}\{R^2\}} \quad (4)$$

Esse resultado é de extrema importância, formalizando a adaptação do algoritmo LMS com passo variável. No entanto, é normalmente negligenciado na literatura [6], [7], [11], [13], [14], [15], onde muitas vezes os algoritmos são justificados ou avaliados comparativamente entre si, com base em resultados de simulação. Essa política pode resultar em conclusões precipitadas acerca do desempenho dos diferentes algoritmos de controle do passo, conforme demonstram as seções seguintes.

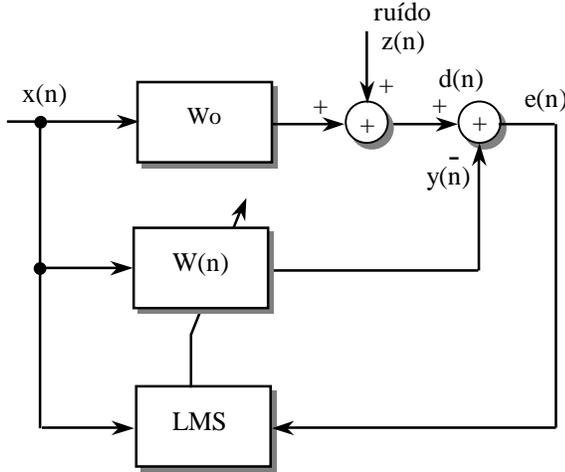


Fig. 1 - O problema da identificação de sistemas.

### 3. O PLANO DE APRENDIZADO

Os algoritmos de passo variável são normalmente avaliados usando curvas separadas para o EMQ e para o passo como funções de  $n$  [6], [7], [11], [15]. No entanto, avaliar o impacto da sequência de passos no EMQ não é uma tarefa trivial, quando realizada em curvas separadas. Conforme ilustra a Fig. 2, as curvas  $\xi(n)$  e  $\mu(n)$  podem ser vistas como projeções de uma trajetória espacial  $\xi(\mu, n)$ . Uma terceira projeção, até então inexplorada, situa-se no plano de aprendizado  $[\xi, \mu]$  [16]. Capturando informações simultâneas do passo de cálculo e da figura de mérito correspondente, o EMQ, caracteriza-se de maneira mais compacta a adaptação com passo variável, gerando-se trajetórias características dos algoritmos, nas quais o número de iterações  $n$  é um parâmetro implícito. Dado que os algoritmos de passo variável geralmente determinam o passo em função de alguma variável do processo adaptativo [6], [7], [8], [11], [13], [14], achou-se conveniente refletir os eixos do plano de aprendizado (plano  $[\mu, \xi]$ ), explicitando o passo de cálculo como variável dependente do erro médio quadrático (Fig. 3). Embora implícito em  $n$ , o plano de aprendizado ainda preserva alguma informação temporal da adaptação: é possível avaliar os algoritmos verificando qual trajetória alcança o menor erro para um dado número de iterações pré-estabelecido (projeção sobre o eixo  $\xi$ ) [16]. Para sinais correlacionados, os modelos analíticos necessitam de informações não conhecidas *a priori*, como a solução de Wiener e a matriz  $R$ . Nesses casos, a metodologia proposta pode ser empregada na avaliação dos algoritmos de passo variável. Quando o sinal de referência é ruído branco, como na formulação de identificação de sistemas (Fig. 1), é possível utilizar a metodologia também para projetar os algoritmos de passo variável e seus parâmetros.

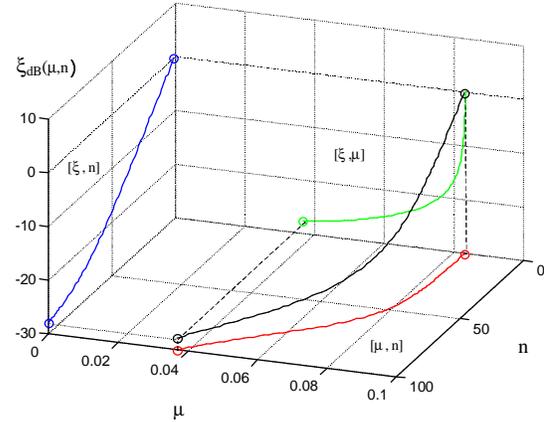


Fig. 2 – Projeções possíveis da trajetória espacial.

#### 3.1 O Regime Transitório e a Trajetória Ótima

A trajetória ótima é o lugar geométrico dos pontos no plano  $[\mu, \xi]$  que caracterizam a evolução ótima do algoritmo LMS, segundo as expressões (2), (3) e (4). A dinâmica dos algoritmos de passo variável pode ser representada por trajetórias no plano de aprendizado. Os algoritmos serão eficientes na medida em que suas trajetórias se aproximarem da trajetória ótima.

#### 3.2 Geração de Trajetórias Características

Basicamente, gerar trajetórias características consiste em traçar o EMQ em função da sequência de passos empregada. Além da teoria da independência [1], [11], algumas das hipóteses normalmente empregadas no estudo de algoritmos de passo variável são [6], [7], [11], [16]:

- O passo de cálculo  $\mu(n)$  e seu valor quadrático  $\mu^2(n)$  são estatisticamente independentes do vetor de entradas  $X(n)$  e do vetor de erro nos coeficientes  $V(n)$ ;
- O valor esperado do passo de cálculo quadrático é aproximadamente igual ao quadrado do valor esperado do passo,  $E[\mu^2(n)] \approx E^2[\mu(n)]$ ;

Usando essas hipóteses, pode-se generalizar a equação (3), chegando-se à equação (5), válida para sinais gaussianos [16]:

$$K(n+1) = K(n) - E[\mu(n)] \cdot [RK(n) + K(n)R] + E^2[\mu(n)] \cdot [R \cdot \text{Tr}\{RK(n)\} + 2RK(n)R + R \cdot \xi_{\min}] \quad (5)$$

De posse das equações (2) e (5) e de um modelo para o passo de cálculo médio  $E[\mu(n)]$ , descreve-se o comportamento do EMQ de algoritmos de passo variável, obtendo-se então, as trajetórias no plano  $[\mu, \xi]$ .

#### 3.3 O Regime Permanente e a Curva de Estagnação

A curva de estagnação é o lugar geométrico dos pontos onde pode ocorrer a estagnação do algoritmo de controle do passo para as diferentes trajetórias características possíveis. A curva é definida como:

$$\xi_{\infty}(\mu) = \lim_{n \rightarrow \infty} \xi(n, \mu) \quad (6)$$

Com as hipóteses utilizadas na seção anterior, a curva de estagnação pode ser obtida das expressões existentes de desajuste para passo fixo, encontradas em [1] e [2], por exemplo. Para  $x(n)$  ruído branco, a curva de estagnação é dada por:

$$\xi_{\infty}(\mu) = \frac{1}{1 - \frac{N}{2} \left( \frac{\mu \sigma_x^2}{1 - \mu \sigma_x^2} \right)} \cdot \xi_{\min} \quad (7)$$

onde  $N$  é a ordem do filtro e  $\sigma_x^2$  é a variância do sinal  $x(n)$ .

### 3.4 As Regiões de Estagnação e Transitória

As proximidades da curva de estagnação compreendem uma região de convergência, em geral, lenta. É útil definir graficamente limites para tais “proximidades”, dado que nessa região a escolha do passo é mais crítica [16]. A definição da região implica na escolha de um limite superior para o passo, com o EMQ associado. Uma boa definição para a *região de estagnação* é utilizar o EMQ em regime permanente correspondente ao *passo ótimo inicial*  $\mu_{\text{opt}}(0)$ , dado que a seqüência ótima de passos e a curva de aprendizado associada são sempre decrescentes a partir da inicialização:

$$\xi_{\infty}(\mu)_{\mu=\mu_{\text{opt}}(0)} \quad (8)$$

Traçando-se uma reta suporte, *paralela* ao eixo dos passos (reta tracejada na Fig. 3), que passe pelo ponto correspondente à estagnação com passo fixo ótimo inicial  $\mu_{\text{opt}}(0)$  e ao erro associado, divide-se o plano de aprendizado em *região de estagnação* e *região transitória*, como indicado na Fig. 3.

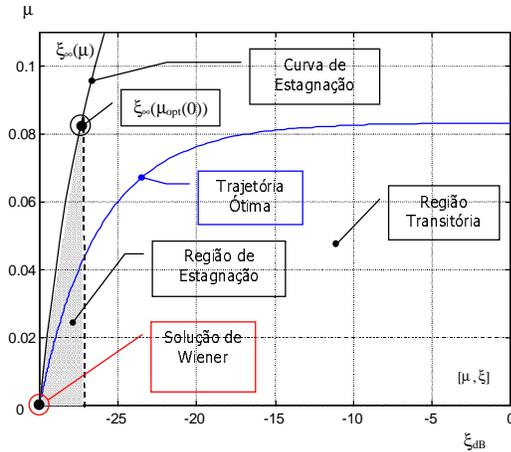


Fig.3 - O plano de aprendizado.

### 3.5 A Solução de Wiener

Representa o *ponto de estagnação ótimo*, onde o algoritmo atinge o erro médio quadrático mínimo em regime permanente, dado pela solução de Wiener.

### 3.6 Utilizando o Plano de Aprendizado

Os algoritmos são avaliados e projetados graficamente, comparando-se trajetórias no plano de aprendizado. É interessante observar algumas heurísticas na utilização do plano:

- A proximidade com a trajetória ótima é mais importante na região de estagnação. Trajetórias relativamente distantes da trajetória ótima na região transitória, mas que se aproximem da curva ótima na região de estagnação, podem representar uma boa convergência;
- Deve-se evitar trajetórias próximas à curva de estagnação; ela representa o lugar geométrico onde a taxa de convergência é nula (regime permanente); além de representar uma diminuição da taxa de convergência, aumenta-se a possibilidade de estagnação do passo de cálculo, provocando desajustes grandes;
- A região inferior do plano de aprendizado, próxima ao eixo  $\xi$ , deve ser evitada, pois representa uma região de dinâmica lenta ( $\mu$  pequeno);

## 4. APLICAÇÕES

Em se tratando de um processo dinâmico, a inicialização e o intervalo de truncamento possuem um papel importante na adaptação. No caso de ruído branco,  $\mu_{\text{opt}}(0)$  em (4) pode ser determinado a partir da igualdade (9), pressupondo  $W(0)=[0]_{N \times 1}$ :

$$\text{Tr}\{K(0)\} = |W_0|^2 = \frac{\sigma_d^2 - \sigma_z^2}{\sigma_x^2} \quad (9)$$

com  $\sigma_d^2$  representando a variância do sinal desejado  $d(n)$  e  $\sigma_z^2$  o nível de ruído não-correlacionado. Esse será o valor ideal  $\mu_{\text{fopt}}$  para truncamento (limite superior), podendo também ser utilizado na etapa de inicialização. Assim, fazendo  $\mu_{\text{fopt}}=\mu_{\text{opt}}(0)$ :

$$\mu_{\text{fopt}} = \frac{\sigma_d^2 - \sigma_z^2}{(N+2)\sigma_x^2(\sigma_d^2 - \sigma_z^2) + N\sigma_x^2\sigma_z^2} \quad (10)$$

Esse resultado será utilizado nos exemplos a seguir, que correspondem à formulação identificação de sistemas. Em [6] foi proposto um algoritmo simples de passo variável da família LMS. Denominado “Variable Step-Size LMS Algorithm” (“VSS”), o algoritmo adapta o passo pelo erro de estimação quadrático, necessitando o ajuste de duas constantes de controle:

$$\mu(n+1) = \alpha \cdot \mu(n) + \gamma \cdot e^2(n) \quad (11)$$

Tomando-se o valor esperado em (11) e manipulando-se a equação (43) em [6], chega-se, respectivamente, a:

$$E[\mu(n+1)] = \alpha \cdot E[\mu(n)] + \gamma \cdot \xi(n) \quad (12)$$

$$\gamma = \left[ 1 - \left( \frac{1-M}{1+M} \right)^2 \right] \cdot \frac{(1-\alpha^2)}{2(3-\alpha)\text{Tr}\{R\}\sigma_z^2} \quad (13)$$

Dado um desajuste de projeto  $M$  e uma estimativa inicial  $\alpha_0$ , determina-se a constante  $\gamma_0$  correspondente através de (13). O conjunto  $\{\alpha_0, \gamma_0\}$  gera uma trajetória característica no plano  $[\mu, \xi]$ , através de (2), (5) e (12). Sob orientação da trajetória ótima, pode-se corrigir a trajetória inicial refinando-se  $\alpha_0$ . O conjunto corrigido  $\{\alpha_{\text{cor}}, \gamma_{\text{cor}}\}$  é aquele que fornece a maior

proximidade com a trajetória ótima, segundo as heurísticas descritas.

#### 4.1 Um Exemplo com Especificações Rigorosas

O exemplo seguinte utiliza como sinal de referência  $x(n)$  um ruído branco gaussiano com variância unitária. Estipulou-se um desajuste desejado em regime de  $M=1\%$ , com o filtro de Wiener  $W_o^T = [0,1; 0,1; \dots; 0,1]$  e o filtro adaptativo possuindo ordem  $N=100$ . Prevenindo a modelagem exata, foi adicionado à saída da planta um ruído branco gaussiano  $z(n)$  com variância  $\sigma_z^2 = 10^{-3}$ .

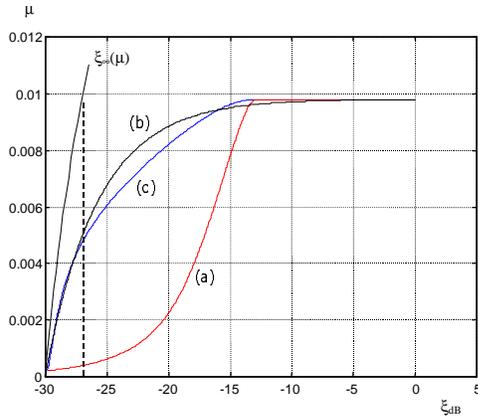


Fig. 4 - Algoritmo VSS: (a) trajetória original; (b) trajetória ótima; (c) trajetória corrigida.

Os limites para o passo são  $\mu_{\min}=0$  e  $\mu_{\max}=0,0098$ , e  $\mu(0)=\mu_{\text{opt}}$ . Como sugerido em [6], utilizou-se  $\alpha_{\text{vss}}=0,97$  e  $\gamma_{\text{vss}}=0,0057$  para atingir o desajuste desejado ( $M=1\%$ ). Nota-se na Fig. 4 que a trajetória correspondente (curva (a)) está distante do caminho ótimo (curva (b)). Corrigindo a trajetória, chega-se ao conjunto de parâmetros  $\alpha_{\text{cor}}=0,998$  e  $\gamma_{\text{cor}}=3,9133 \cdot 10^{-4}$ , representada pela curva (c) na Fig. 4. Claramente, a trajetória corrigida leva a um desempenho superior do algoritmo, como mostra a Fig. 5. Simulações realizadas com desajustes ainda menores (0,2% ou menos), utilizados em [7], por exemplo, revelaram discrepâncias consideravelmente maiores, sempre favoráveis à metodologia de projeto proposta.

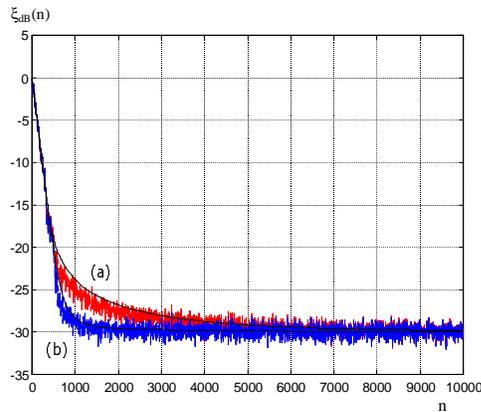


Fig. 5 - Algoritmo VSS: (a) curva original; (b) curva corrigida. Simulação com 100 realizações.

#### 4.2 Comparando Algoritmos: MVSS versus VSS

Aqui será demonstrada a importância de uma avaliação mais sistemática dos algoritmos. Em [7] foi proposto o algoritmo “MVSS”, uma modificação do algoritmo VSS. Infelizmente, não são fornecidos os filtros de Wiener empregados nos exemplos. Porém, nos casos em que  $x(n)$  é ruído branco, pode-se inferir das curvas e das informações disponíveis um filtro de Wiener equivalente que permita reproduzir os exemplos [16].

**Baixa RSN (Relação Sinal-Ruído):** será avaliado o Exemplo 1 de [7]. O sinal de referência é ruído branco gaussiano com variância unitária, excitando um sistema desconhecido FIR de ordem  $N=4$  e um filtro adaptativo FIR de mesma ordem. O ruído de medição na saída da planta é gaussiano e branco, com variância  $\sigma_z^2 = 1$ .

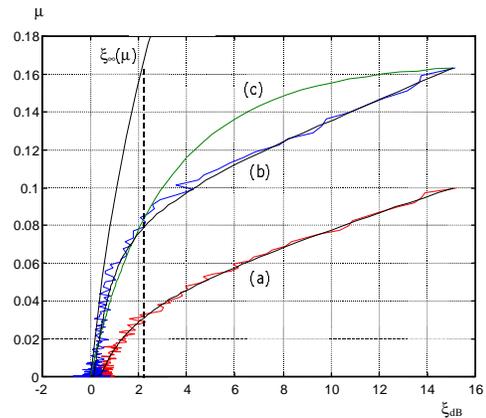


Fig. 6 - Algoritmo VSS: (a) original; (b) corrigido; (c) trajetória ótima. Simulação com 200 realizações

O algoritmo MVSS foi utilizado com os parâmetros  $\alpha_{\text{mvss}}=0,97$ ,  $\gamma_{\text{mvss}}=10^{-3}$  e  $\beta_{\text{mvss}}=0,99$ , escolhidos de maneira a produzir um EMQ em excesso de aproximadamente  $-34$  dB em regime. Um filtro de Wiener equivalente é  $W_o=[2,8; 2,8; 2,8; 2,8]^T$ . O algoritmo VSS foi implementado com  $\alpha_{\text{vss}}=0,97$  e  $\gamma_{\text{vss}}=10^{-5}$ , o que corresponde a um desajuste de  $M=6,9 \cdot 10^{-4}$ . O intervalo de truncamento utilizado (original) foi  $\mu_{\max}=0,1$  e  $\mu_{\min}=5 \cdot 10^{-4}$ .

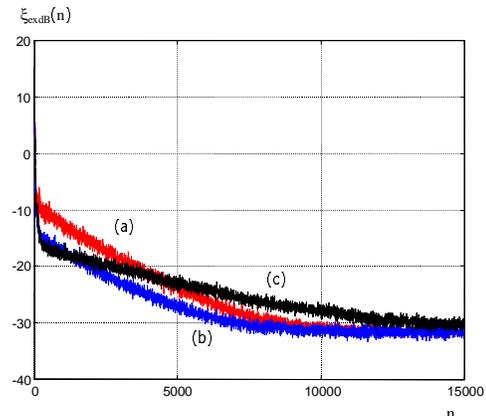


Fig. 7 - EMQ em excesso. Algoritmos: (a) VSS original; (b) VSS corrigido; (c) MVSS. Simulação com 200 realizações.

A correção da trajetória do algoritmo VSS é ilustrada na Fig. 6, envolvendo a inicialização e truncamento com  $\mu_{\min}=5 \cdot 10^{-4}$ ,  $\mu(0)=\mu_{\max}=\mu_{\text{opt}}=0,163$ , e as constantes corrigidas  $\alpha_{\text{cor}}=0,973$  e  $\gamma_{\text{cor}}=9,05 \cdot 10^{-6}$ . A Fig. 7 revela uma melhoria substancial no desempenho do algoritmo VSS, superando o algoritmo MVSS.

**Alta RSN:** O Exemplo 3 em [7] também será avaliado no plano de aprendizado. Este exemplo é semelhante ao anterior, sendo reproduzido com o filtro de Wiener equivalente  $W_o=[1,58; 1,58; 1,58; 1,58]^T$  e com intervalo de truncamento original  $\mu_{\max}=0.1$  e  $\mu_{\min}=5 \cdot 10^{-4}$ . À saída da planta foi adicionado um ruído de medição branco e gaussiano com variância  $\sigma_z^2=10^{-3}$ .

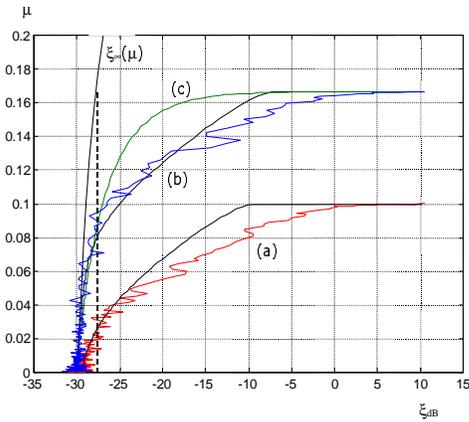


Fig. 8 – Algoritmo VSS: (a) trajetória original; (b) trajetória corrigida; (c) trajetória ótima. Simulação com 200 realizações.

Os parâmetros  $\alpha_{\text{mvss}}=0,97$ ,  $\beta_{\text{mvss}}=0,99$ ,  $\gamma_{\text{mvss}}=1$ ,  $\alpha_{\text{vss}}=0,97$ ,  $\gamma_{\text{vss}}=0,02$  foram selecionados para produzir um erro médio quadrático em excesso de aproximadamente  $-60\text{dB}$  em regime permanente [7]. O desajuste correspondente às constantes utilizadas no algoritmo VSS original é  $M \approx 1.38 \cdot 10^{-3}$ .

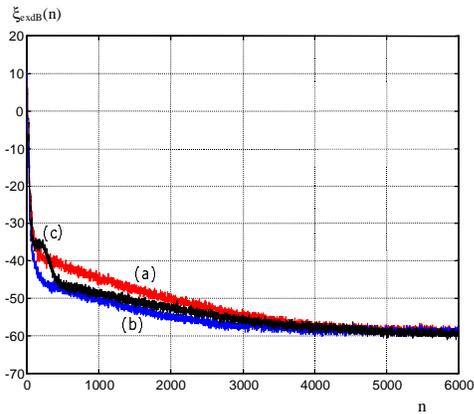


Fig. 9 - EMQ em excesso. Algoritmos: (a) VSS original; (b) VSS corrigido; (c) MVSS. Simulação com 200 realizações

Na Fig. 8, a trajetória original do VSS foi corrigida através dos valores  $\alpha_{\text{cor}}=0,977$  e  $\gamma_{\text{cor}}=0,0154$ , e com inicialização e truncamento de  $\mu(0)=\mu_{\max}=\mu_{\text{opt}} \approx 0,167$ . Novamente, nota-se na

Fig. 9 uma melhoria expressiva do desempenho do algoritmo VSS, chegando a superar o algoritmo MVSS.

### 4.3 Correção da Dinâmica ou Busca Automática?

Dois abordagens, de certa forma concorrentes, serão avaliadas nesta seção: a otimização automática do passo, pretendida por alguns algoritmos [1], [11], [12], e a correção da dinâmica, proposta neste trabalho.

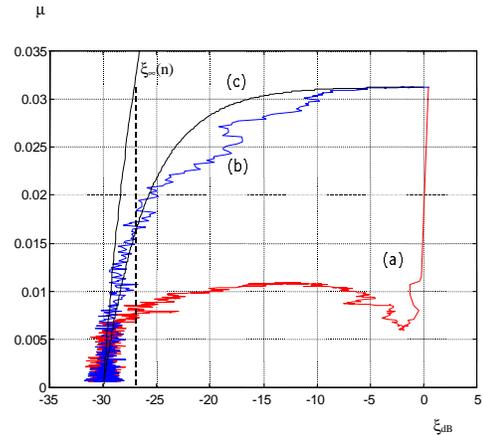


Fig. 10 – Algoritmos: (a) NASS; (b) VSS corrigido. (c) trajetória ótima. Simulação com 100 realizações.

Serão avaliados os algoritmos “NASS” [12], mais recente na literatura, e o algoritmo VSS. Ambos foram projetados para atingir um desajuste de  $M \approx 1\%$ , verificando-se a dinâmica correspondente no plano de aprendizado. O filtro de Wiener utilizado foi  $W_o^T = \frac{1}{\sqrt{30}} [1, 1, 1, \dots, 1]$ , o que determina  $|W_o|=1$  para  $N=30$ . A variância do ruído de medição  $z(n)$ , branco e gaussiano, adicionado à saída da planta, é  $\sigma_z^2=10^{-3}$ .

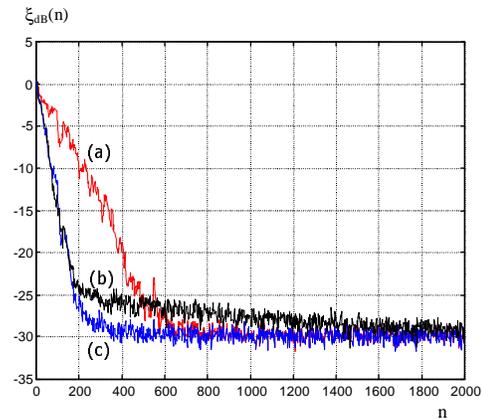


Fig. 11 – Algoritmos: (a) NASS; (b) VSS original; (c) VSS corrigido. Simulação Monte Carlo com 100 realizações

O sinal de referência utilizado é ruído branco gaussiano com variância unitária. O conjunto de parâmetros para o algoritmo VSS, projetado pela metodologia proposta, é  $\alpha_{\text{cor}}=0,994$  e  $\gamma_{\text{cor}}=0,0039$ . Seguindo a sugestão de [12], o algoritmo NASS foi implementado com fatores de perdas  $\rho=\rho_c=0,002$ , visando o

desajuste de projeto. Foi utilizada uma constante  $L_d$  unitária, em se tratando de ruído branco na entrada do filtro [12], [16]. A Fig. 10 mostra as trajetórias dos algoritmos no plano de aprendizado. A Fig. 11 mostra claramente a superioridade do algoritmo VSS, quando corretamente projetado.

#### 4.4 Caso Não-Estacionário

O exemplo anterior será repetido com os mesmos parâmetros. Os algoritmos são liberados à convergência quando, na iteração  $n=1000$ , manifesta-se uma mudança abrupta na solução de Wiener,  $W_o' = -W_o$  [7], [11], provocando um “salto” no erro médio quadrático. A Fig. 12 apresenta os resultados da simulação Monte Carlo. Verifica-se que mesmo projetado para um desajuste pequeno,  $M=1\%$ , o algoritmo VSS corrigido ainda se mantém superior, mesmo na presença de não-estacionaridades, afirmando a vantagem da metodologia desenvolvida sobre o procedimento proposto em [6].

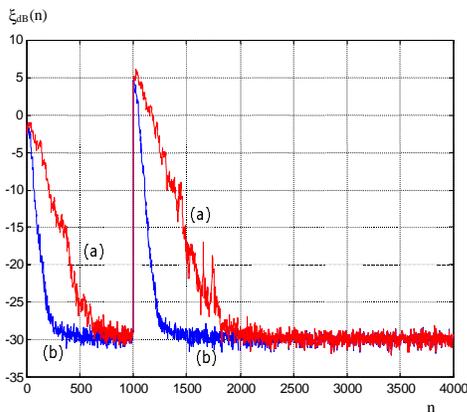


Fig. 12 – Caso não-estacionário. Algoritmos: (a) NASS; (b) VSS corrigido. Simulação com 100 realizações

## 5. CONCLUSÕES

Os exemplos explorados indicam a necessidade de uma avaliação mais sistemática dos algoritmos, que empregue critérios analíticos de desempenho. A proposição de novos algoritmos deve levar isto em conta. A metodologia proposta fornece uma base simples e eficiente para efetuar a inspeção dos algoritmos, também permitindo o projeto em certos casos. Revelou-se a potencialidade de algoritmos simples que, corretamente projetados, apresentam desempenho superior a seus concorrentes mais complexos, inclusive em ambientes não-estacionários. A metodologia pode ainda ser estendida a outros algoritmos, como o LMS normalizado, entre outros [16].

## 6. REFERÊNCIAS

- [1] HAYKIN, S. *Adaptive Filter Theory*. 3. ed., Prentice Hall, 1996.
- [2] HOROWITZ, L.; SENNE, K. Performance Advantage of Complex LMS for Controlling Narrow-Band Adaptive Arrays. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. ASSP-29, n. 3, p. 722-736, June 1981.
- [3] GARDNER, W. A. Learning Characteristics of Stochastic-Gradient-Descent Algorithms: A General Study, Analysis,

- and Critique. *Signal Processing*, North Holland, v. 6, p. 113-133, 1984.
- [4] SLOCK, D. On the Convergence Behavior of the LMS and the Normalized LMS Algorithms. *IEEE Trans. on Signal Processing*, v. 41, n. 9, p. 2811-2825, Sept. 1993.
- [5] EVANS, J. B.; XUE, P.; LIU, B. Analysis and Implementation of Variable Step Size Adaptive Algorithms. *IEEE Trans. on Signal Processing*, v. 41, n. 8, p. 2517-2535, Aug. 1993.
- [6] KWONG, R. H.; JONHSTON, E. W. A Variable Step Size LMS Algorithm. *IEEE Trans. Signal Processing*, v. 40, n. 7, p. 1633-1642, July 1992.
- [7] ABULNASR, T.; MAYYAS, K. A Robust Variable Step-Size LMS-Type Algorithm: Analysis and Simulations. *IEEE Trans. Signal Processing*, vol. 45, n.3, p 631-639, Mar. 1997.
- [8] SHAN, T. J.; KAILATH, T. Adaptive Algorithms with an Automatic Gain Control Feature. *IEEE Trans. on Circuits and Systems*, v. 35, n.1, p 122-127, Jan. 1988
- [9] OKELLO, J.; ITOH, Y.; FUKUI, Y.; NAKANISHI, I. *et al.* A New Modified Variable Step Size for the LMS Algorithm. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (May 1998, Monterey). Monterey, CA, v. 5, pp 170-173.
- [10] KUSHNER, H. J.; YANG, J. Analysis of Adaptive Step-Size SA Algorithms for Parameter Tracking. *IEEE Trans. Automatic Control*, v. 40, n. 8, p 1403-1410, Aug. 1995.
- [11] MATHEWS, J. V.; XIE, Z. A Stochastic Gradient Adaptive Filter with Gradient Adaptive Step Size. *IEEE Trans. Signal Processing*, v. 41, n.6, p 2075-2087, June 1993.
- [12] KOIKE, S. A Novel Adaptive Step Size Control Algorithm for Adaptive Filters. In: INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING (Mar.1999: Phoenix), Phoenix, Arizona, v. 4, paper no. 1054, 1999.
- [13] GAN, W. S. Fuzzy step-size adjustment for the LMS algorithm. *Signal Processing*, 49, p. 145-149, 1996.
- [14] \_\_\_\_\_. Designing a fuzzy step size LMS algorithm, *IEE Proc. Vis. Image Signal Processing*, v. 144, n. 5, p. 261-266, Oct. 1997.
- [15] KARNI, S.; ZENG, G. A New Convergence Factor for Adaptive Filters. *IEEE Trans. on Circuits and Systems*, v. 36, n. 7, July 1989.
- [16] LOPES, Cássio G. “Uma Metodologia de Avaliação e Projeto de Algoritmos Adaptativos de Passo Variável”. Dissertação de Mestrado, UFSC, Nov. 1999.