

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
PROGRAMA DE PÓS-GRADUAÇÃO EM  
ENGENHARIA ELÉTRICA**

José Gil Fausto Zipf

**CLASSIFICAÇÃO, ANÁLISE ESTATÍSTICA E NOVAS  
ESTRATÉGIAS DE ALGORITMOS LMS DE PASSO VARIÁVEL**

Tese submetida ao Programa de Pós-graduação em Engenharia Elétrica, Área de Concentração Comunicações e Processamento de Sinais, da Universidade Federal de Santa Catarina para a obtenção do grau de Doutor em Engenharia Elétrica  
Orientador: Prof. Dr. Rui Seara  
Co-orientador: Prof. Dr. Orlando José Tobias

Florianópolis, SC

2011

Catálogo na fonte pela Biblioteca Universitária  
da  
Universidade Federal de Santa Catarina

Z79c Zipf, José Gil Fausto  
Classificação, análise estatística e novas estratégias de  
algoritmos LMS de passo variável [tese] / José Gil Fausto  
Zipf ; orientador, Rui Seara. - Florianópolis, SC, 2011.  
167 p.: grafs., tabs.

Tese (doutorado) - Universidade Federal de Santa Catarina,  
Centro Tecnológico. Programa de Pós-Graduação em Engenharia  
Elétrica.

Inclui referências

1. Engenharia elétrica. 2. Algoritmos - Classificação.  
3. Estatística - Análise. 4. Processamento de sinal adaptativo  
- Teses. 5. Ruído - Teses. I. Seara, Rui. II. Universidade  
Federal de Santa Catarina. Programa de Pós-Graduação em  
Engenharia Elétrica. III. Título.

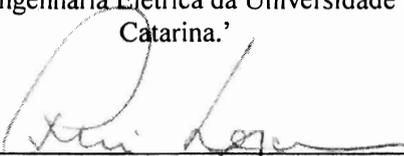
CDU 621.3



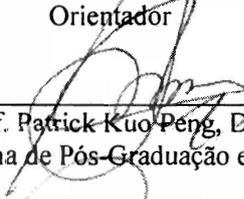
# CLASSIFICAÇÃO, ANÁLISE ESTATÍSTICA E NOVAS ESTRATÉGIAS DE ALGORITMOS LMS DE PASSO VARIÁVEL

José Gil Fausto Zipf

‘Esta Tese foi julgada adequada para a obtenção do Título de Doutor em Engenharia Elétrica, na área de concentração Comunicações e Processamento de Sinais, e aprovada em sua forma final pelo Programa de Pós Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.’



Prof. Rui Seara, Dr.  
Orientador

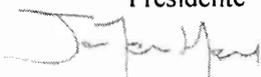


Prof. Patrick Kuo-Peng, Dr.  
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

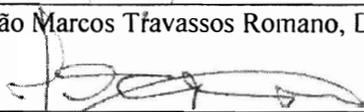
Banca Examinadora:



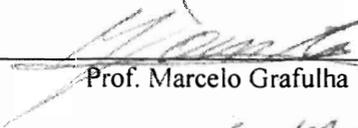
Prof. Rui Seara, Dr.  
Presidente



Prof. João Marcos Travassos Romano, Dr.



Prof. João César Moura Mota, Dr.



Prof. Marcelo Grafulha Vanti, Dr.



Prof. Eduardo Luiz Ortiz Batista, Dr.



Prof. Hans Helmut Zürn, Ph.D.



Dedico este trabalho à minha esposa  
Nicole e a meus filhos Mariah, Pedro e  
Bruno. Amo vocês mais do que tudo.



## AGRADECIMENTOS

Gostaria de deixar aqui meus sinceros agradecimentos a todos que contribuíram de alguma forma para a realização deste trabalho, em especial:

Ao meu orientador Prof. Rui Seara que, com sua paciência, dedicação e entusiasmo, me passou ensinamentos que guardarei para o resto de minha vida.

Ao meu amigo e co-orientador Prof. Orlando José Tobias, que infelizmente nos deixou de forma prematura, mas que certamente estará para sempre nas nossas lembranças.

A toda minha família, em especial à minha esposa e companheira de todas as horas Nicole, pelo seu amor e dedicação a mim nos difíceis momentos em que minha saúde esteve frágil, mas era preciso continuar.

Ao pessoal do LINSE, em especial ao Elton, pelas suas valorosas contribuições na formatação final desta tese e de todos os artigos submetidos no decorrer do curso.

À Universidade Regional de Blumenau (FURB), pelo suporte financeiro que viabilizou esta formação, em especial aos colegas do departamento de Engenharia Elétrica e de Telecomunicações, pelo apoio e amizade em todas as horas.

Aos professores do programa de Pós-Graduação em Engenharia Elétrica da UFSC, pela rica convivência e pelos conhecimentos transmitidos durante o primeiro ano do curso.

Aos professores membros da banca examinadora, por suas contribuições para o aprimoramento deste trabalho.

A Deus, por me amparar nos momentos difíceis e pela graça de ter me permitido concluir este trabalho.



## RESUMO

Este trabalho de pesquisa visa o estudo de algoritmos LMS de passo variável (VSSLMS) objetivando: classificá-los segundo uma nomenclatura unificada, realizar uma análise estatística dos mais importantes algoritmos e desenvolver novas estratégias de ajuste do passo de adaptação originando, com isso, novos algoritmos. Os algoritmos VSSLMS têm grande importância prática, visto que apresentam um melhor desempenho em relação aos de passo fixo, permitindo a obtenção simultânea de uma rápida convergência com menor desajuste. A literatura técnica apresenta um grande número de trabalhos que tratam de algoritmos VSSLMS. A classificação dos principais algoritmos VSSLMS apresentada neste trabalho baseia-se no estudo dos diferentes princípios de ajuste do passo de adaptação utilizados. A partir desse estudo, uma modificação em um algoritmo VSSLMS existente, baseado no gradiente do erro quadrático, e dois novos algoritmos VSSLMS, com base na autocorrelação do erro são propostos, sendo um deles não-paramétrico e com elevada imunidade a variações no nível do ruído de medição. Também, são desenvolvidos modelos estocásticos de seis importantes algoritmos VSSLMS, permitindo uma análise mais aprofundada de cada estratégia e fornecendo uma ferramenta de predição de desempenho de tais algoritmos. Resultados de simulação numérica comparam o desempenho das estratégias estudadas em um cenário comum, comprovam o desempenho dos novos algoritmos propostos e atestam a precisão dos modelos desenvolvidos.

**Palavras-chave:** Algoritmos LMS de passo variável. Análise estatística. Classificação de algoritmos. Filtragem adaptativa. Imunidade ao ruído.



## ABSTRACT

This research work discusses and studies variable step-size LMS (VSSLMS) algorithms aiming to classify them according to a unified nomenclature, to perform a statistical analysis of the most important algorithms, and to develop new strategies for updating the step size, resulting thereby in novel algorithms. The VSSLMS algorithms have large practical importance, since they exhibit a better performance as compared with those using fixed step size, allowing to obtain, simultaneously, both higher convergence speed and smaller error misadjustment. The open literature presents a large number of different approaches for VSSLMS algorithms. The classification of VSSLMS algorithms presented in this work is based on the study of different principles for updating the step size. From this study, a modification of an existing VSSLMS algorithm based on the squared error gradient, and two new VSSLMS algorithms, based on the error autocorrelation, are proposed, being one of them non-parametric and presenting high immunity to variations of the measurement noise power. In addition, stochastic models of six relevant VSSLMS algorithms are developed, allowing a detailed analysis of each strategy and providing a useful tool to predict the performance of such algorithms. Numerical simulation results compare the performance of the VSSLMS algorithms studied here in a common scenario, assess the performance of the new proposed algorithms, as well as verify the accuracy of the developed models.

**Keywords:** Variable step-size LMS algorithms. Statistical analysis. Algorithm classification. Adaptive filtering. Noise immunity.



## Lista de Figuras

2.1. Diagrama em blocos de um esquema de identificação de sistemas. ....	34
3.1. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Okello [18] para sinal de entrada branco. ....	59
3.2. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Okello [18] para sinal de entrada branco. ....	60
3.3. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Okello [18] para sinal de entrada branco. ....	60
3.4. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor gradiente estimado $\mathbf{g}(n)$ do algoritmo de Okello [18] para sinal de entrada branco. ....	61
3.5. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Okello [18] para sinal de entrada correlacionado. ....	62
3.6. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Okello [18] para sinal de entrada correlacionado. ....	62
3.7. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Okello [18] para sinal de entrada correlacionado. ....	63
3.8. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor gradiente estimado $\mathbf{g}(n)$ do algoritmo de Okello [18] para sinal de entrada correlacionado. ....	63
3.9. Evolução de $E[\mu^2(n)]$ (cinza) e $E^2[\mu(n)]$ (preto) para o algoritmo de Okello. ....	64
3.10. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Aboulnasr [28] para sinal de entrada branco. ....	69

3.11.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Aboulnasr [28] para sinal de entrada branco.....	70
3.12.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Aboulnasr [28] para sinal de entrada branco.....	70
3.13.	Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.....	71
3.14.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.....	72
3.15.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.....	72
3.16.	Evolução de $E[\mu^2(n)]$ (cinza) e $E^2[\mu(n)]$ (preto) para o algoritmo de Aboulnasr [28]. .....	73
3.17.	Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do Algoritmo de Wei [37] para sinal de entrada branco.....	83
3.18.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Wei [37] para sinal de entrada branco.....	83
3.19.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Wei [37] para sinal de entrada branco.....	84
3.20.	Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do Algoritmo de Wei [37] para sinal de entrada correlacionado. ....	85
3.21.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Wei [37] para sinal de entrada correlacionado.....	85
3.22.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Wei [37] para sinal de entrada correlacionado. ....	86

4.1. Evolução do EQM. (a) Algoritmo original [17]. (b) Algoritmo modificado (proposto).....	93
4.2. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Shan modificado [42] para sinal de entrada branco.....	94
4.3. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Shan modificado [42] para sinal de entrada branco.....	95
4.4. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Shan modificado [42] para sinal de entrada branco.....	95
4.5. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.....	96
4.6. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.....	97
4.7. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.....	97
4.8. Evolução de $E[\mu^2(n)]$ (cinza) e $E^2[\mu(n)]$ (preto) para o algoritmo de Shan modificado [42].....	98
4.9. Evolução do passo variável para sinal de entrada branco.....	101
4.10. Evolução do coeficiente $w_1(n)$ para sinal de entrada branco.....	102
4.11. Comportamento do EQM em excesso para sinal de entrada branco.....	102
4.12. Comportamento do EQM em excesso para sinal de entrada branco considerando uma mudança abrupta no sistema a ser identificado na iteração 20000.....	103
4.13. Evolução do passo variável para sinal de entrada correlacionado.....	104
4.14. Evolução do coeficiente $w_1(n)$ para sinal de entrada correlacionado.....	105

4.15. Comportamento do EQM em excesso para sinal de entrada correlacionado.....	105
4.16. Comportamento do EQM em excesso para sinal de entrada correlacionado considerando uma mudança abrupta no sistema a ser identificado na iteração 20000.....	106
4.17. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto API para sinal de entrada branco.....	107
4.18. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo proposto API para sinal de entrada branco.....	107
4.19. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto API para sinal de entrada branco.....	108
4.20. Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto API para sinal de entrada correlacionado.....	109
4.21. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo proposto API para sinal de entrada correlacionado.....	109
4.22. Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto API para sinal de entrada correlacionado.....	110
4.23. Evolução de $E[\mu^2(n)]$ (cinza) e $E^2[\mu(n)]$ (preto) para o algoritmo proposto API.....	110
4.24. Resultados de simulação para $\sigma_{\eta}^2 = 0,001$ (SNR = 30 dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle $\alpha(n)$ e sua aproximação.....	117
4.25. Resultados de simulação para $\sigma_{\eta}^2 = 0,0001$ (SNR = 40 dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle $\alpha(n)$ e sua aproximação.....	119

4.26.	Resultados de simulação para $\sigma_{\eta}^2 = 0,01$ (SNR = 20 dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle $\alpha(n)$ e sua aproximação. ....	121
4.27.	Resultados de simulação considerando dois níveis de ruído $\sigma_{\eta}^2 = 0,001$ e $0,0013$ , e algoritmo de Benesty [25] operando com valor fixo de $\hat{\sigma}_{\eta}^2 = 0,001$ . (a) Curva de EQM em excesso. (b) Evolução do passo variável. ....	122
4.28.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP2 para sinal de entrada branco. ....	123
4.29.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo proposto AP2 para sinal de entrada branco. ....	124
4.30.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP2 para sinal de entrada branco. ....	124
4.31.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP2 para sinal de entrada correlacionado. ....	125
4.32.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor $\mathbf{w}(n)$ do algoritmo proposto AP2 para sinal de entrada correlacionado. ....	126
4.33.	Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP2 para sinal de entrada correlacionado. ....	126
4.34.	Evolução de $E[\mu^2(n)]$ (cinza) e $E^2[\mu(n)]$ (preto) para o algoritmo proposto AP2. ....	127
5.1.	Comportamento do EQM em excesso dos algoritmos baseados no gradiente do erro quadrático. ....	131
5.2.	EQM em excesso dos algoritmos baseados no gradiente do erro quadrático considerando uma mudança abrupta no sistema a ser identificado. ....	132

5.3. Comportamento do EQM em excesso dos algoritmos baseados no erro quadrático.....	132
5.4. EQM em excesso dos algoritmos baseados no erro quadrático considerando uma mudança abrupta no sistema a ser identificado.....	133
5.5. Comportamento do EQM em excesso dos algoritmos baseados na autocorrelação do erro.....	134
5.6. EQM em excesso dos algoritmos baseados na autocorrelação do erro considerando uma mudança abrupta no sistema a ser identificado.....	134
5.7. Comportamento do EQM em excesso dos algoritmos baseados no valor absoluto do erro.....	135
5.8. EQM em excesso dos algoritmos baseados no valor absoluto do erro considerando uma mudança abrupta no sistema a ser identificado.....	135
5.9. Comportamento do EQM em excesso dos algoritmos baseados no ajuste proporcional ao vetor $\mathbf{w}(n)$ .....	136
5.10. EQM em excesso dos algoritmos baseados no ajuste proporcional ao vetor $\mathbf{w}(n)$ considerando uma mudança abrupta na planta.....	137
5.11. Comportamento do EQM em excesso dos algoritmos baseados na normalização do vetor de erros, teoria do passo ótimo e otimização com restrição do ruído de medição.....	137
5.12. Comportamento do EQM em excesso dos algoritmos baseados na normalização do vetor de erros, teoria do passo ótimo e otimização com restrição do ruído de medição considerando uma mudança abrupta na planta.....	138
5.13. Comparação do EQM em excesso dos algoritmos de melhor desempenho.....	139
5.14. Comparação do EQM em excesso dos algoritmos de melhor desempenho considerando uma mudança abrupta na planta.....	139

## **Lista de Tabelas**

2.1. Principais Passos do Algoritmo LMS .....	37
4.1. Parâmetros Usados para Simulação – Sinal de Entrada Branco .....	101
4.2. Parâmetros Usados na Simulação – Sinal de Entrada Correlacionado .....	104
5.1. Quadro Comparativo da Complexidade Computacional de Algoritmos VSSLMS .....	130



# SUMÁRIO

<b>1. INTRODUÇÃO.....</b>	<b>25</b>
<b>1.1 Objetivos do Trabalho.....</b>	<b>30</b>
<b>1.2 Organização do Trabalho .....</b>	<b>31</b>
<b>1.3 Publicações da Tese .....</b>	<b>31</b>
<b>2. FUNDAMENTOS DO ALGORITMO LMS.....</b>	<b>33</b>
<b>2.1 Introdução .....</b>	<b>33</b>
<b>2.2 Problema de Identificação de Sistemas.....</b>	<b>33</b>
<b>2.3 Algoritmo LMS .....</b>	<b>34</b>
2.3.1 Derivação do Algoritmo LMS .....	35
2.3.2 Teoria da Independência (TI).....	37
2.3.3 Modelo Estocástico do Algoritmo LMS .....	38
2.3.4 Análise de Estabilidade do Algoritmo LMS .....	43
<b>2.4 Algoritmo LMS Normalizado – NLMS.....</b>	<b>45</b>
<b>2.5 Algoritmos VSSLMS .....</b>	<b>46</b>
<b>2.6 Conclusões .....</b>	<b>47</b>
<b>3. CLASSIFICAÇÃO E MODELAGEM DE ALGORITMOS VSSLMS.....</b>	<b>49</b>
<b>3.1 Introdução .....</b>	<b>49</b>
<b>3.2 Aspectos Relevantes para a Modelagem de Algoritmos VSSLMS .....</b>	<b>50</b>
<b>3.3 Algoritmos Baseados no Gradiente do Erro Quadrático .</b>	<b>51</b>
3.3.1 Algoritmo de Richards [13] .....	52
3.3.2 Algoritmo de Mathews [14].....	53
3.3.3 Algoritmo de Wee Peng Ang [19] .....	53
3.3.4 Algoritmo de Farhang [16] .....	54
3.3.5 Algoritmo de Okello [18].....	55
3.3.6 Algoritmo de Shin [20] .....	64
<b>3.4 Algoritmos Baseados no Erro Quadrático.....</b>	<b>65</b>
3.4.1 Algoritmo de Kwong [22].....	65
3.4.2 Algoritmo de Nakanishi [23] .....	66

3.4.3	Algoritmo de Costa [24] .....	66
3.4.4	Algoritmo de Benesty [25] .....	66
<b>3.5</b>	<b>Algoritmos Baseados na Autocorrelação do Erro .....</b>	<b>67</b>
3.5.1	Algoritmo de Aboulnasr [28] .....	68
<b>3.6</b>	<b>Algoritmos Baseados no Valor Absoluto do Erro.....</b>	<b>73</b>
3.6.1	Algoritmo de Kim [31] .....	73
3.6.2	Algoritmo de Gollamudi [32] .....	74
<b>3.7</b>	<b>Algoritmos Baseados na Normalização do Vetor de Erros .....</b>	<b>74</b>
3.7.1	Algoritmo de Ramadan [33] .....	75
<b>3.8</b>	<b>Algoritmos Baseados nos Valores Absolutos do Vetor de Coeficientes .....</b>	<b>75</b>
3.8.1	Algoritmo de Rohani [34].....	75
3.8.2	Algoritmo de Benesty (IPNLMS) [36] .....	76
<b>3.9</b>	<b>Algoritmos Baseados na Teoria do Passo Ótimo .....</b>	<b>77</b>
3.9.1	Algoritmo de Yan-Bin [38].....	77
<b>3.10</b>	<b>Algoritmos Baseados na Otimização com Restrição do Ruído de Medição.....</b>	<b>78</b>
3.10.1	Algoritmo de Wei [37] .....	78
<b>3.11</b>	<b>Conclusões .....</b>	<b>86</b>
<b>4.</b>	<b>NOVOS ALGORITMOS VSSLMS .....</b>	<b>89</b>
<b>4.1</b>	<b>Introdução .....</b>	<b>89</b>
<b>4.2</b>	<b>Algoritmo de Shan Modificado [42].....</b>	<b>90</b>
4.2.1	Modelo Estocástico para o Algoritmo de Shan Modificado [42].....	91
4.2.2	Resultados de Simulação .....	92
<b>4.3</b>	<b>Algoritmo Proposto #1 (AP1) [29].....</b>	<b>98</b>
4.3.1	Modelo Estocástico para o Algoritmo AP1 .....	99
4.3.2	Resultados de Simulação .....	100
<b>4.4</b>	<b>Algoritmo Proposto #2 (AP2) [30].....</b>	<b>111</b>
4.4.1	Modelo Estocástico para o Algoritmo AP2 .....	113
4.4.2	Resultados de Simulação .....	114
<b>4.5</b>	<b>Conclusões .....</b>	<b>127</b>

<b>5. RESULTADOS COMPARATIVOS DE ALGORITMOS</b>	
VSSLMS .....	129
<b>5.1 Introdução .....</b>	<b>129</b>
<b>5.2 Complexidade Computacional de Algoritmos</b>	
VSSLMS .....	129
<b>5.3 Resultados de Simulação – Comparação dos</b>	
<b>Algoritmos VSSLMS .....</b>	<b>129</b>
5.3.1 Algoritmos Baseados no Gradiente do Erro	
Quadrático.....	131
5.3.2 Algoritmos Baseados no Erro Quadrático.....	131
5.3.3 Algoritmos Baseados na Autocorrelação do Erro.....	133
5.3.4 Algoritmos Baseados no Valor Absoluto do Erro.....	133
5.3.5 Algoritmos Baseados no Ajuste Proporcional aos	
Valores Absolutos do Vetor de Coeficientes .....	136
5.3.6 Algoritmos Baseados na Normalização do Vetor	
de Erros, na Teoria do Passo Ótimo e na	
Otimização com Restrição do Ruído de Medição.....	136
5.3.7 Comparação entre os Algoritmos de Melhor	
Desempenho.....	138
<b>5.4 Conclusões .....</b>	<b>140</b>
<b>6. CONCLUSÕES E COMENTÁRIOS FINAIS .....</b>	<b>141</b>
<b>APÊNDICE A .....</b>	<b>145</b>
<b>APÊNDICE B .....</b>	<b>153</b>
<b>APÊNDICE C .....</b>	<b>157</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>163</b>



## Introdução

Nas últimas décadas, a filtragem adaptativa vem se tornando um tema de pesquisa cada vez mais relevante na área de processamento digital de sinais. Tal relevância é corroborada pela popularização dos processadores digitais de sinais de alto desempenho, que possibilitam a implementação de diferentes estruturas de filtragem adaptativa para as mais diversas aplicações. Nos casos em que não se tem conhecimento *a priori* das estatísticas dos sinais envolvidos no processo, o uso de filtros lineares ótimos, denominados filtros de Wiener, se torna inadequado. Isso é devido à impossibilidade de estimação tanto da matriz de autocorrelação do sinal de entrada quanto do vetor de correlação cruzada entre o sinal de entrada e o sinal desejado. Por outro lado, uma solução prática pode ser viabilizada pela utilização de filtros adaptativos, cuja escolha tem se mostrado atrativa em uma vasta gama de aplicações, dentre as quais destacam-se cancelamento de eco [1], [2], equalização de canal [1], [3], cancelamento de ruído [1], [4], e identificação de sistemas propriamente dita [1], [5].

A pesquisa em filtragem adaptativa vem despertando grande interesse devido ao seu vasto campo de utilização como também pela contínua busca de novos métodos que incorporem, simultaneamente, as características mais desejáveis em um filtro adaptativo: simplicidade de implementação, traduzida por uma baixa complexidade computacional; robustez a erros de quantização quando implementado em processadores com precisão finita; elevada imunidade a ruído; capacidade de aquisição e rastreamento, quando se opera em ambientes não estacionários; alta velocidade de convergência; e adequada precisão e estabilidade em regime permanente. Todo filtro adaptativo tem a ele associado um algoritmo de adaptação que, através de um mecanismo de aprendizagem, ajusta recursivamente seus coeficientes de forma a minimizar uma determinada função custo, usualmente o erro quadrático médio (EQM). Com base em tal função custo, o algoritmo faz com que os coeficientes do filtro converjam, iterativamente, para a solução ótima no sentido médio quadrático do erro, (solução de Wiener) [1]. Essa convergência pode ocorrer em ambientes estacionários e não-estacionários, contanto que, neste último caso, as estatísticas dos sinais

envolvidos variem de forma relativamente lenta ou pouco freqüente, possibilitando ao filtro adaptativo exercer sua capacidade de rastreamento. O algoritmo mais utilizado em aplicações práticas de filtragem adaptativa é o *least-mean-square* (LMS) [6]. Tal algoritmo atualiza os coeficientes do filtro na direção do negativo do gradiente instantâneo do sinal de erro ao quadrado, sendo largamente utilizado em comunicação digital [2], [3], sistemas de geoposicionamento [7], identificação de sistemas [5], arranjos de antenas [8], cancelamento de eco [2], controle adaptativo [9], sistemas ativos de supressão de vibrações e ruídos [10], dentre outras aplicações de grande importância comercial e tecnológica. A popularidade do algoritmo LMS é devida, principalmente, à sua robustez numérica e baixa complexidade computacional, características extremamente atrativas em aplicações práticas. Por outro lado, ele tem como principal desvantagem a forte dependência de sua velocidade de convergência com o grau de correlação do sinal de entrada. Sinais de entrada que apresentem uma elevada dispersão de autovalores de sua matriz de autocorrelação, como, por exemplo, sinais de voz, podem acarretar séria degradação na velocidade de convergência do algoritmo [3].

Pertencente à classe dos algoritmos baseados no gradiente estocástico, o algoritmo LMS pode ser derivado a partir do método *steepest descent*, pela substituição do vetor gradiente real por uma estimativa ruidosa, baseada em informações instantâneas do sistema, visando sua convergência (na média) para a solução de Wiener. Dessa forma, a utilização do algoritmo LMS não requer conhecimento prévio das estatísticas dos sinais envolvidos, como ocorre no algoritmo *steepest descent* [11].

O algoritmo LMS convencional opera com passo de adaptação fixo. O valor do passo de adaptação é determinante para o desempenho de convergência do filtro adaptativo. Observa-se que quanto maior for o valor do passo de adaptação, desde que dentro do limite de estabilidade, maior será a velocidade de convergência do algoritmo, o que é desejável; porém, maior será também o erro em excesso em regime permanente. Já um valor de passo de adaptação reduzido proporciona desajustes menores em regime permanente, ao preço de uma convergência mais lenta. Dessa forma, a escolha do valor do passo de adaptação deve levar em conta o compromisso existente entre velocidade de convergência e desajuste em regime permanente.

Em aplicações envolvendo o algoritmo LMS, é usualmente desejável que o filtro adaptativo apresente simultaneamente alta

velocidade de convergência e reduzido desajuste em regime permanente. É nesse contexto que os algoritmos LMS de passo variável (*variable step-size LMS – VSSLMS*) são requeridos, visando ajustar dinamicamente, durante o processo de convergência, o valor do passo de adaptação [12]-[40]. Assim, esses algoritmos usam um valor de passo de adaptação elevado no início do processo de convergência, reduzindo gradualmente o valor do passo (segundo algum critério pré-estabelecido) até ser atingido o regime permanente. Dessa forma, várias estratégias de ajuste de passo vêm sendo propostas na literatura, as quais são baseadas em diferentes princípios, tais como gradiente do erro quadrático [12]-[21], erro quadrático [22]-[27], autocorrelação do erro [28]-[30], valor absoluto do erro [31], filtragem *set-membership* [32], normalização dos dados de erros [33], valores absolutos dos elementos do vetor de coeficientes [34]-[36] e ainda outros métodos menos conhecidos [37]-[40]. Com isso, surge um problema decorrente da quantidade de trabalhos neste tópico; nos quais, a grande maioria, não apresenta uma padronização que permita estabelecer comparações de desempenho frente a cenários comuns de aplicação.

Como não há na literatura técnica artigos que apresentem um estudo com um número significativo de algoritmos VSSLMS, a escolha do algoritmo de melhor desempenho em uma determinada aplicação pode não ser uma tarefa trivial. Assim, este trabalho inicialmente apresenta os principais algoritmos VSSLMS, estabelecendo uma padronização da nomenclatura utilizada e classificando-os em diferentes categorias, segundo a idéia central de cada algoritmo [41]. Um ponto interessante a se destacar é que todos os algoritmos são estudados usando um mesmo cenário, permitindo então estabelecer uma comparação de desempenho mais confortável. Resultados de simulação comparam todos os algoritmos estudados, considerando como aplicação comum um problema de identificação de sistema.

Um certo número de algoritmos VSSLMS propostos na literatura apresentam alguns problemas em sua operação, como é o caso do aqui denominado algoritmo de Shan [17], o qual se baseia no gradiente médio do erro quadrático. Tal algoritmo se mostra bastante simples e eficaz, apresentando, porém, um problema elementar de convergência que surge quando o gradiente médio estimado tem valor negativo. Isso leva o algoritmo LMS a exibir um comportamento inadequado. Assim, também é proposta neste trabalho uma modificação no algoritmo de Shan [17], pela utilização do valor absoluto do gradiente médio estimado como sendo uma medida mais confiável de distância em

relação ao valor do erro quadrático mínimo [42]. Essa alteração, então, contorna o problema de convergência do referido algoritmo, como também preserva sua simplicidade, esta última caracterizada por um ligeiro aumento de complexidade computacional quando comparado ao algoritmo LMS convencional.

Em algoritmos de passo variável, tanto o método de ajuste do valor do passo quanto o desajuste do algoritmo são afetados pelo ruído de medição. Em [22], é introduzido um algoritmo VSSLMS baseado no erro quadrático instantâneo. Esse algoritmo, denominado aqui algoritmo de Kwong, apresenta um desempenho adequado para um grande número de aplicações práticas. No entanto, o processo de ajuste do passo de adaptação, como também o desajuste do algoritmo são fortemente afetados pelo ruído de medição. Em [28], é então proposta uma modificação ao algoritmo de Kwong, originando o aqui denominado algoritmo de Aboulnasr, que apresenta uma considerável melhora de desempenho em ambientes fortemente contaminados por ruído. Tal algoritmo é baseado na medida de autocorrelação entre amostras adjacentes [ $lag(1)$ ] do sinal de erro. No entanto, em função do grau de correlação do sinal de entrada, podem ocorrer situações em que a função de autocorrelação do sinal de erro  $lag(1)$  reduza excessivamente o passo de adaptação, degradando, com isso, a velocidade de convergência do algoritmo. Assim, é proposto neste trabalho um novo algoritmo VSSLMS, baseado na função de autocorrelação do erro considerando diferentes atrasos. Esse algoritmo é indicado para ambientes com baixa razão sinal-ruído (SNR) [29], se apresentando como uma versão melhorada do algoritmo de Aboulnasr [28], acelerando o seu processo de convergência. Nele, a idéia básica para ajustar o parâmetro de passo é considerar as diversas funções de autocorrelação do sinal de erro  $lag(1), lag(2), \dots, lag(N)$ , com  $N$  denotando a ordem do filtro adaptativo. É importante ressaltar ainda que o uso de vários  $lags$  aumenta apenas marginalmente a complexidade computacional do algoritmo. Através de simulações numéricas, é verificado o desempenho do algoritmo aqui proposto.

Embora o algoritmo proposto com  $Nlags$  [29] represente uma versão melhorada do algoritmo de Aboulnasr [28], ainda persiste um problema comum à maioria dos algoritmos VSSLMS, que é a baixa imunidade ao ruído de medição. Esse problema implica degradação de desempenho dos algoritmos sempre que ocorram mudanças na variância do ruído de medição, a menos que sejam efetuados reajustes de parâmetros. Tal ação não é uma tarefa trivial de ser realizada na prática.

Neste trabalho, é também proposto um novo algoritmo LMS normalizado de passo variável (VSS-NLMS), não-paramétrico, baseado na autocorrelação do sinal de erro. O objetivo desse algoritmo é aumentar a imunidade ao ruído de medição do algoritmo de Aboulnasr [28], sem a necessidade de ajuste de parâmetros, mesmo em situações em que ocorrem alterações na variância desse ruído. No algoritmo proposto também não há necessidade de conhecer o nível de ruído *a priori*, tal como exigido pelo algoritmo não-paramétrico proposto em [25].

Observa-se ainda que a maioria dos trabalhos publicados nesta área não apresenta modelos estocásticos que possibilitem uma predição de desempenho dos algoritmos em uma dada aplicação. A modelagem estocástica em filtragem adaptativa visa descrever o comportamento médio das principais variáveis envolvidas na operação de um algoritmo adaptativo. Através dessa modelagem é possível prever (na média) o desempenho do algoritmo adaptativo em uma determinada condição de operação. Além disso, com o modelo do algoritmo em mãos, pode-se entender melhor o seu comportamento, assim como propor modificações que melhorem o desempenho dos algoritmos para certas condições e/ou aplicações.

Os modelos estocásticos permitem ainda uma análise mais aprofundada, estabelecendo relações de causa e efeito entre os parâmetros de cada algoritmo e os índices de desempenho relevantes para os mais diferentes cenários de aplicação. Especialmente, no caso de algoritmos VSSLMS, a modelagem estocástica constitui-se em uma poderosa ferramenta de auxílio ao projetista, permitindo a comparação das diversas estratégias e fornecendo subsídios para a melhor escolha em uma dada aplicação. Assim, a modelagem estocástica de algoritmos VSSLMS é também um assunto abordado neste trabalho de pesquisa. São propostos modelos analíticos para seis algoritmos VSSLMS, sendo dois deles baseados no gradiente do erro quadrático, um na otimização com restrição do ruído de medição e três na autocorrelação do erro [42]-[45]. Tais modelos são desenvolvidos considerando algumas hipóteses simplificativas, uma vez que a análise estatística de algoritmos LMS, especificamente os de passo variável, requer um tratamento matemático bastante complexo. Na apresentação dos modelos desenvolvidos, as hipóteses simplificativas consideradas são devidamente ressaltadas e justificadas. Simulações numéricas, comparando os resultados obtidos a partir do método de Monte Carlo e através dos modelos propostos, permitem avaliar a qualidade das predições aqui obtidas.

Em resumo, este trabalho trata do estudo e da modelagem estocástica de algoritmos VSSLMS. São estudadas, avaliadas e classificadas as principais estratégias de ajuste do passo de adaptação de algoritmos VSSLMS recentemente apresentadas na literatura e a correção de um problema elementar de convergência de um desses algoritmos é proposta. São apresentados também dois novos algoritmos VSSLMS baseados na autocorrelação do erro, sendo que em um deles não há necessidade de ajuste de parâmetros. Como a maioria das publicações não apresenta os modelos estocásticos dos algoritmos, são também realizadas análises estatísticas de seis importantes algoritmos VSSLMS, sendo dois desses algoritmos propostos neste trabalho de pesquisa.

## 1.1 Objetivos do Trabalho

Os principais objetivos deste trabalho são:

- i) Desenvolver novos algoritmos LMS de passo variável visando melhorar o desempenho de filtros adaptativos em diferentes aplicações.
- ii) Realizar a modelagem estocástica de algoritmos já existentes, fornecendo com isso ferramentas de auxílio ao projeto de filtros adaptativos considerando algoritmos LMS de passo variável.

Como objetivos específicos, enumeramos:

- i) Realizar uma revisão bibliográfica e avaliar, através de simulação numérica, as principais estratégias de diversos algoritmos VSSLMS já publicados.
- ii) Descrever cada algoritmo utilizando uma nomenclatura unificada.
- iii) Estabelecer uma classificação dos algoritmos VSSLMS estudados.
- iv) Propor novas estratégias de ajuste do passo de adaptação de algoritmos VSSLMS.
- v) Determinar modelos estocásticos para seis importantes algoritmos VSSLMS, criando com isso uma ferramenta para predição do desempenho de tais algoritmos.
- vi) Realizar um estudo comparativo de algoritmos VSSLMS, utilizando cenários comuns de aplicação.

## 1.2 Organização do Trabalho

Este trabalho está organizado em seis capítulos. Neste Capítulo 1, são apresentados introdução, objetivos, organização da tese e uma lista de publicações decorrentes deste trabalho de tese. Como base teórica deste trabalho de pesquisa, são apresentados no Capítulo 2 os fundamentos do algoritmo LMS, com ênfase à sua derivação e modelagem estocástica. Também são brevemente discutidos o algoritmo LMS normalizado (NLMS) e generalidades dos algoritmos VSSLMS. No Capítulo 3, é apresentada uma classificação das principais estratégias de algoritmos VSSLMS encontradas na literatura e são propostos modelos estocásticos de três algoritmos de grande importância. O Capítulo 4 introduz os novos algoritmos VSSLMS propostos neste trabalho de pesquisa. Nele uma modificação em um algoritmo existente e dois novos algoritmos baseados na autocorrelação do erro são discutidos. Também são apresentadas análises estatísticas dos algoritmos propostos. O Capítulo 5 traz os resultados comparativos das diversas estratégias de algoritmos VSSLMS em um cenário comum de aplicação. Finalmente, no Capítulo 6, são destacados as conclusões e comentários finais deste trabalho de tese.

## 1.3 Publicações da Tese

Este trabalho de pesquisa originou as seguintes publicações:

- i) J. G. F. Zipf, O. J. Tobias, R. Seara, “New insights on the noise constrained LMS algorithm,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, Hawaii, USA, Apr. 2007, pp. 1365-1368.
- ii) J. G. F. Zipf, O. J. Tobias, R. Seara, “Estudo e modelo estocástico do algoritmo LMS restrito ao ruído de medição,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBRT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- iii) J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo estocástico para o algoritmo de Shan de passo variável modificado,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBRT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6

- iv) J. G. F. Zipf, O. J. Tobias, R. Seara, “Estudo comparativo e classificação de algoritmos LMS de passo variável,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- v) J. G. F. Zipf, O. J. Tobias, R. Seara, “A new VSSLMS algorithm based on error autocorrelation,” in *Proc. Europ. Signal Process. Conf. (EUSIPCO)*, Lausanne, Switzerland, Aug. 2008, pp. 1-5.
- vi) J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo estocástico do algoritmo LMS de passo variável baseado no gradiente do sinal de erro quadrático,” in *Anais do XXVI Simpósio Brasileiro de Telecomunicações (SBrT)*, Rio de Janeiro, RJ, Brasil, Set. 2008, pp. 1-6.
- vii) J. G. F. Zipf, O. J. Tobias, R. Seara, “Algoritmo LMS de passo variável para ambientes com razão sinal ruído baixa,” in *Anais do XXVI Simpósio Brasileiro de Telecomunicações (SBrT)*, Rio de Janeiro, RJ, Brasil, Set. 2008, pp. 1-6.
- viii) J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo analítico de um algoritmo LMS de passo variável baseado na autocorrelação do sinal de erro,” in *Anais do XXVII Simpósio Brasileiro de Telecomunicações (SBrT)*, Blumenau, SC, Brasil, Set. 2009, pp. 1-6.
- ix) J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo estocástico do algoritmo LMS de passo variável baseado na autocorrelação do erro com N lags,” in *Anais do XXVII Simpósio Brasileiro de Telecomunicações (SBrT)*, Blumenau, SC, Brasil, Set. 2009, pp. 1-6.
- x) J. G. F. Zipf, O. J. Tobias, R. Seara, “Non-parametric VSS-NLMS algorithm with control parameter based on the error correlation,” in *Proc. 7th International Telecommunications Symposium (ITS)*, Manaus, AM, Brazil, Sep. 2010, pp. 1-6.

## Fundamentos do Algoritmo LMS

### 2.1 Introdução

Os algoritmos adaptativos baseados no gradiente estocástico, tal como o LMS, apresentam características vantajosas como a sua simplicidade e o fato de não necessitarem de conhecimento prévio a respeito das estatísticas dos sinais envolvidos no processo [1]. Tais vantagens, aliadas à sua robustez a erros numéricos, fazem do algoritmo LMS um dos mais utilizados em aplicações de filtragem adaptativa.

Este capítulo trata dos conceitos básicos relacionados ao algoritmo LMS, objetivando fornecer uma base teórica para o desenvolvimento deste trabalho de pesquisa. Aqui é discutida a derivação do algoritmo LMS, considerando aplicação em um problema de identificação de sistemas, assim como são apresentados uma análise de estabilidade e um modelo estocástico. Variações do algoritmo LMS convencional, tais como o LMS normalizado (NLMS) e o LMS de passo variável (VSSLMS) são também brevemente discutidas neste capítulo.

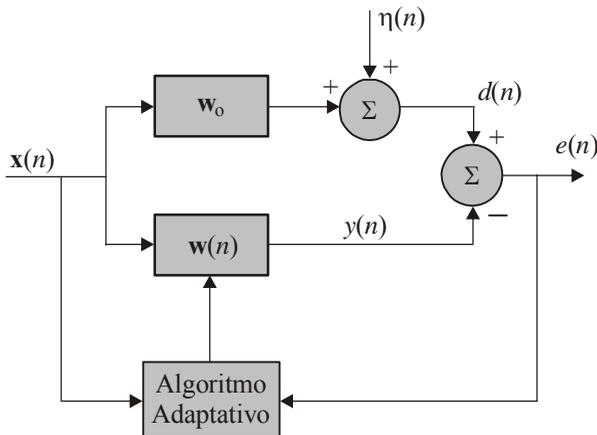
### 2.2 Problema de Identificação de Sistemas

Para estudar os algoritmos adaptativos, é considerado um problema de identificação de sistemas [1], cujo esquema está ilustrado na Figura 2.1. Nessa figura, o sinal de saída do sistema desconhecido, corrompido por ruído aditivo de medição, é dado por

$$d(n) = \mathbf{w}_o^T \mathbf{x}(n) + \eta(n) \quad (2.1)$$

onde  $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T$  denota o vetor de entrada, sendo  $\{x(n)\}$  um processo Gaussiano estacionário de média zero e variância  $\sigma_x^2$ ,  $\eta(n)$  é um ruído de medição independente e identicamente distribuído (i.i.d) com média zero e variância  $\sigma_\eta^2$ ,  $\mathbf{w}(n) = [w(n) \ w(n-1) \ \dots \ w(n-N+1)]^T$ , o vetor de coeficientes e  $\mathbf{w}_o = [w_0 \ w_1 \ \dots \ w_{N-1}]^T$ , a planta do sistema a ser identificado. O sinal de erro é

$$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n). \quad (2.2)$$



**Figura 2.1.** Diagrama em blocos de um esquema de identificação de sistemas.

Nesta configuração, a utilização do filtro adaptativo  $\mathbf{w}(n)$  visa obter uma estimativa de  $\mathbf{w}_0$  a partir de amostras dos sinais de entrada e saída [1]. Tal estimativa é obtida pela minimização de uma função custo, que no caso do algoritmo LMS é dada pelo erro quadrático instantâneo. Cancelamento de eco acústico e de eco de linha, proveniente de aplicações em telefonia, são dois dos muitos exemplos de um problema de identificação de sistemas [1], [5].

### 2.3 Algoritmo LMS

O algoritmo LMS [6] é caracterizado por sua simplicidade e robustez, sendo o algoritmo mais popular utilizado em filtragem adaptativa. Nesse algoritmo, a adaptação do vetor de coeficientes  $\mathbf{w}(n)$  do filtro se dá na direção do negativo do gradiente instantâneo do sinal de erro ao quadrado [com respeito a  $\mathbf{w}(n)$ ], obtendo-se, na média, a convergência para a solução de Wiener [1].

Nesta seção, são apresentados a derivação do algoritmo LMS e o seu modelo estocástico baseado na TI [1], sendo também discutidos aspectos relacionados à sua estabilidade.

### 2.3.1 Derivação do Algoritmo LMS

Na maioria das aplicações práticas de filtragem adaptativa, as estatísticas dos sinais envolvidos não são conhecidas *a priori*, impossibilitando, com isso, a obtenção em tempo real do vetor gradiente exato a cada iteração. Assim, ao invés de atualizar o vetor de coeficientes  $\mathbf{w}(n)$  com base no verdadeiro gradiente da função custo definida por  $J_w(n) = e^2(n)$ , o algoritmo LMS utiliza uma aproximação estocástica do vetor gradiente  $\hat{\nabla}J_w(n)$ , obtida a partir de dados instantâneos disponíveis no sistema. Dessa forma, o vetor gradiente do sinal de erro ao quadrado em relação ao vetor de coeficientes é escrito como

$$\hat{\nabla}J_w(n) = \frac{\partial e^2(n)}{\partial \mathbf{w}(n)} = 2e(n) \frac{\partial e(n)}{\partial \mathbf{w}(n)}. \quad (2.3)$$

Derivando então (2.2) em relação ao vetor  $\mathbf{w}(n)$ , tem-se

$$\frac{\partial e(n)}{\partial \mathbf{w}(n)} = -\mathbf{x}(n) \quad (2.4)$$

substituindo (2.4) em (2.3), resulta em

$$\hat{\nabla}J_w(n) = \frac{\partial e^2(n)}{\partial \mathbf{w}(n)} = -2e(n)\mathbf{x}(n) \quad (2.5)$$

e, finalmente, substituindo (2.2) em (2.5), chega-se à expressão do gradiente instantâneo em função do sinal de entrada, do sinal desejado e do vetor de coeficientes do filtro, dada por

$$\hat{\nabla}J_w(n) = 2[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n) - d(n)\mathbf{x}(n)]. \quad (2.6)$$

Objetivando determinar o comportamento médio do gradiente estimado do sinal de erro ao quadrado, toma-se o valor esperado de (2.6), resultando em

$$E[\hat{\nabla}J_w(n)] = 2\{E[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)] - E[d(n)\mathbf{x}(n)]\}. \quad (2.7)$$

Definindo agora a matriz de autocorrelação  $\mathbf{R}$  do sinal de entrada e o vetor de correlação cruzada  $\mathbf{p}$  entre o sinal de entrada e o sinal desejado, dados por

$$\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)] \quad (2.8)$$

$$\mathbf{p} = E[\mathbf{x}(n)d(n)] \quad (2.9)$$

e usando a TI, (2.7) pode ser reescrita como

$$E[\hat{\nabla}J_{\mathbf{w}}(n)] = 2[\mathbf{R}\mathbf{w}(n) - \mathbf{p}] = \nabla J_{\mathbf{w}}(n). \quad (2.10)$$

A expressão (2.10) corresponde ao gradiente exato do EQM, usado para atualização do vetor de coeficientes no algoritmo *steepest descent*. Com isso, observa-se que, na média, o algoritmo LMS produz o mesmo resultado do que o algoritmo *steepest descent*.

Como no algoritmo LMS o vetor  $\mathbf{w}(n)$  é atualizado na direção contrária à estimativa do gradiente  $\hat{\nabla}J_{\mathbf{w}}(n)$ , dada por (2.5), a expressão recursiva de ajuste dos coeficientes do filtro é obtida por

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n). \quad (2.11)$$

Substituindo  $2\mu$  por  $\mu$ , sem perda de generalidade, pode-se reescrever (2.11) como

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}(n). \quad (2.12)$$

A expressão (2.12) realiza, então, a atualização do vetor de coeficientes do filtro, sendo calculada a cada iteração, assumindo um valor inicial  $\mathbf{w}(0)$ . Um resumo da seqüência de passos do algoritmo LMS é apresentado na Tabela 2.1.

Tabela 2.1. Principais Passos do Algoritmo LMS

<p>Definições de Projeto:</p> <ul style="list-style-type: none"> <li>• Ordem do filtro <math>N</math></li> <li>• Passo de adaptação <math>\mu</math></li> <li>• Vetor de coeficientes inicial</li> </ul> $\mathbf{w}(0) = [0 \quad 0 \quad \dots \quad 0]^T$ <p>Execução:</p> <p>Para cada iteração <math>n</math>, calcule</p> $\mathbf{x}(n) = [x(n) \quad x(n-1) \quad \dots \quad x(n-N+1)]^T$ $e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n)$ $\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e(n)\mathbf{x}(n)$
---

### 2.3.2 Teoria da Independência (TI)

Os modelos estocásticos do algoritmo LMS são derivados com base na TI [1], na qual é assumido que o vetor de entrada  $\mathbf{x}(n)$  e o sinal desejado  $d(n)$  satisfazem as seguintes condições:

- i) Os vetores  $\mathbf{x}(n)$ ,  $\mathbf{x}(n-1)$ , ...,  $\mathbf{x}(1)$  constituem uma seqüência de vetores estatisticamente independentes.
- ii) Em um instante  $n$ , o vetor  $\mathbf{x}(n)$  é estatisticamente independente dos valores passados do sinal desejado  $d(n-1)$ ,  $d(n-2)$ , ...,  $d(1)$ .
- iii)  $d(n)$  é estatisticamente independente de seus valores passados  $d(n-1)$ ,  $d(n-2)$ , ...,  $d(1)$ .

Observa-se que em uma estrutura de filtragem FIR com linha de retardo,  $\mathbf{x}(n)$  não pode ser estatisticamente independente de  $\mathbf{x}(n-1)$ , ainda que  $\{x(n)\}$  seja ruído branco. Contudo, o uso da TI simplifica sobremaneira a análise estatística de algoritmos adaptativos. Sem o uso das suposições (i) a (iii), a análise estatística do algoritmo LMS se

tornaria bastante complexa. Resultados disponíveis na literatura respaldam o uso da TI, indicando que, na maioria dos casos de análise estatística de filtros adaptativos, os modelos desenvolvidos com base na TI seguem o comportamento médio dos algoritmos, sobretudo quando o passo de adaptação é pequeno.

### 2.3.3 Modelo Estocástico do Algoritmo LMS

Apesar de ser um algoritmo bastante simples, a determinação de um modelo estocástico para o algoritmo LMS não é uma tarefa trivial. Para determinar um modelo que descreva adequadamente o comportamento médio do algoritmo LMS, para o vetor de coeficientes e curva de aprendizagem, quase sempre se recorre à utilização da TI [11]. Assim, com base nessa teoria e usando (2.12), verifica-se que o vetor de coeficientes  $\mathbf{w}(n)$  depende de:

- i) Valores passados do vetor de entrada  $\mathbf{x}(n-1)$ ,  $\mathbf{x}(n-2)$ , ...; assim,  $\mathbf{w}(n)$  pode ser considerado estatisticamente independente de  $\mathbf{x}(n)$ .
- ii) Valores passados do sinal desejado  $d(n-1)$ ,  $d(n-2)$ , ...; assim,  $\mathbf{w}(n)$  é também considerado estatisticamente independente de  $d(n)$ .

Estas suposições são utilizadas para simplificar a análise de algoritmo LMS, objetivando obter um modelo estocástico que descreva o comportamento médio do vetor de coeficientes e curva de aprendizagem.

#### A. Modelo para o comportamento médio do vetor de coeficientes

Este modelo descreve a evolução do valor esperado do vetor de coeficientes  $\mathbf{w}(n)$ , aqui para o caso do algoritmo LMS. Para determiná-lo, substitui-se (2.2) em (2.12), resultando em

$$\begin{aligned}\mathbf{w}(n+1) &= \mathbf{w}(n) + \mu[d(n) - \mathbf{x}^T(n)\mathbf{w}(n)]\mathbf{x}(n) \\ &= \mathbf{w}(n) + \mu[d(n)\mathbf{x}(n) - \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)].\end{aligned}\quad (2.13)$$

Agora, tomando o valor esperado em ambos os lados de (2.13)

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu\{E[d(n)\mathbf{x}(n)] - E[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)]\} \quad (2.14)$$

e utilizando a TI, tem-se

$$E[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)] = E[\mathbf{x}(n)\mathbf{x}^T(n)]E[\mathbf{w}(n)]. \quad (2.15)$$

Dessa forma, (2.14) é então reescrita como

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu\{E[d(n)\mathbf{x}(n)] - E[\mathbf{x}(n)\mathbf{x}^T(n)]E[\mathbf{w}(n)]\}. \quad (2.16)$$

Usando (2.8) e (2.9) em (2.16) finalmente chega-se a uma expressão que descreve o comportamento médio do vetor de coeficientes  $\mathbf{w}(n)$  em função da matriz de autocorrelação  $\mathbf{R}$  e do vetor de correlação cruzada  $\mathbf{p}$ . Assim,

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu\{\mathbf{p} - \mathbf{R}E[\mathbf{w}(n)]\}. \quad (2.17)$$

Algumas vezes é interessante determinar o modelo descrito por (2.17) em função do vetor de erro nos coeficientes definido por

$$\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_o \quad (2.18)$$

onde  $\mathbf{w}_o$  é a solução de wiener do sistema a ser identificado (planta), considerando que a ordem do sistema é definida. Então, isolando  $\mathbf{w}(n)$  em (2.18), substituindo a expressão resultante em (2.17) e tomando o valor esperado de ambos os lados da expressão obtida, tem-se

$$E[\mathbf{v}(n+1)] = (\mathbf{I} - \mu\mathbf{R})E[\mathbf{v}(n)]. \quad (2.19)$$

### B. Curva de aprendizagem

A curva de aprendizagem descreve a evolução do erro quadrático médio do algoritmo LMS. Para determiná-la, considera-se o sinal de erro em função do vetor de erro nos coeficientes dado por

$$e(n) = d(n) - \mathbf{v}^T(n)\mathbf{x}(n) - \mathbf{w}_o^T\mathbf{x}(n). \quad (2.20)$$

Assumindo ordem do sistema conhecida e definindo  $e_o(n)$  como o mínimo erro de estimação, escrito por

$$e_o(n) = \eta(n) = d(n) - \mathbf{w}_o^T\mathbf{x}(n) \quad (2.21)$$

e substituindo (2.21) em (2.20), resulta em

$$e(n) = e_o(n) - \mathbf{v}^T(n)\mathbf{x}(n). \quad (2.22)$$

Dessa forma, uma expressão para o sinal de erro quadrático pode ser encontrada pela determinação do quadrado de (2.22), resultando em

$$\begin{aligned}
 e^2(n) &= e_o^2(n) - 2e_o(n)\mathbf{v}^T(n)\mathbf{x}(n) + \mathbf{v}^T(n)\mathbf{x}(n)\mathbf{v}^T(n)\mathbf{x}(n) \\
 &= e_o^2(n) - 2e_o(n)\mathbf{v}^T(n)\mathbf{x}(n) + \mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n).
 \end{aligned} \tag{2.23}$$

A curva de aprendizagem, denotada por  $J(n)$ , descreve a evolução do EQM e pode ser obtida determinando-se o valor esperado de (2.23). Assim,

$$\begin{aligned}
 J(n) &= E[e^2(n)] = E[e_o^2(n)] - 2E[e_o(n)\mathbf{v}^T(n)\mathbf{x}(n)] \\
 &\quad + E[\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)].
 \end{aligned} \tag{2.24}$$

Considerando agora  $\mathbf{x}(n)$  e  $\mathbf{v}(n)$  independentes, decorrente do uso da TI, e utilizando o princípio da ortogonalidade, (2.24) pode ser reescrita como

$$\begin{aligned}
 J(n) &= E[e_o^2(n)] - 2E[e_o(n)\mathbf{x}^T(n)]E[\mathbf{v}(n)] + E[\mathbf{v}^T(n)E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{v}(n)] \\
 &= E[e_o^2(n)] - 2\underbrace{E[e_o(n)\mathbf{x}^T(n)]}_0 E[\mathbf{v}(n)] + E[\mathbf{v}^T(n)\underbrace{E[\mathbf{x}(n)\mathbf{x}^T(n)]}_R\mathbf{v}(n)] \\
 &= E[e_o^2(n)] + E[\mathbf{v}^T(n)\mathbf{R}\mathbf{v}(n)].
 \end{aligned} \tag{2.25}$$

A expressão (2.25) descreve a evolução do EQM. O primeiro termo à direita em (2.25) representa valor mínimo da função custo e é obtido quando o vetor de coeficientes  $\mathbf{w}(n)$  assume o valor de  $\mathbf{w}_o$ , tornando nulo o vetor de erro nos coeficientes. Nesse caso, o EQM mínimo depende apenas da variância do ruído aditivo. Portanto,

$$J_{\min} = E[e_o^2(n)] = \sigma_\eta^2. \tag{2.26}$$

O último termo à direita de (2.25) corresponde ao EQM em excesso  $J_{\text{ex}}(n)$ , definido como  $J(n) - J_{\min}$ . A expressão para  $J_{\text{ex}}(n)$ , convenientemente manipulada, resulta em

$$\begin{aligned}
 J_{\text{ex}}(n) &= E[\mathbf{v}^T(n)\mathbf{R}\mathbf{v}(n)] \\
 &= E\{\text{tr}[\mathbf{v}^T(n)\mathbf{R}\mathbf{v}(n)]\} \\
 &= E\{\text{tr}[\mathbf{R}\mathbf{v}(n)\mathbf{v}^T(n)]\} \\
 &= \text{tr}\{E[\mathbf{R}\mathbf{v}(n)\mathbf{v}^T(n)]\} \\
 &= \text{tr}[\mathbf{R}\mathbf{K}(n)]
 \end{aligned} \tag{2.27}$$

onde  $\mathbf{K}(n)$  é a matriz de correlação do vetor de erro nos coeficientes, dada por  $E[\mathbf{v}(n)\mathbf{v}^T(n)]$ . Agora, substituindo (2.27) em (2.25), obtém-se

$$J(n) = J_{\min} + \text{tr}[\mathbf{R}\mathbf{K}(n)]. \quad (2.28)$$

Observa-se que a expressão da curva de aprendizagem dada por (2.28) depende da matriz  $\mathbf{K}(n)$ . Para determinar então  $\mathbf{K}(n)$ , substituiu-se (2.18) em (2.12) de forma a se obter uma expressão recursiva para o vetor de erro nos coeficientes. Assim,

$$\begin{aligned} \mathbf{v}(n+1) &= \mathbf{v}(n) + \mu e(n)\mathbf{x}(n) \\ &= \mathbf{v}(n) + \mu[d(n)\mathbf{x}(n) - \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n) - \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}_o]. \end{aligned} \quad (2.29)$$

Substituindo (2.21) em (2.29), tem-se

$$\mathbf{v}(n+1) = \mathbf{v}(n) + \mu[e_o(n)\mathbf{x}(n) - \mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)]. \quad (2.30)$$

Calculando agora a transposta em ambos os lados de (2.30), obtém-se

$$\mathbf{v}^T(n+1) = \mathbf{v}^T(n) + \mu[e_o(n)\mathbf{x}^T(n) - \mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)] \quad (2.31)$$

com isso, o produto  $\mathbf{v}(n+1)\mathbf{v}^T(n+1)$  pode ser determinado. Assim,

$$\begin{aligned} &\mathbf{v}(n+1)\mathbf{v}^T(n+1) \\ &= \mathbf{v}(n)\mathbf{v}^T(n) + \mu\mathbf{v}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n) + \mu e_o(n)\mathbf{v}(n)\mathbf{x}^T(n) \\ &\quad - \mu\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{v}^T(n) + \mu^2\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n) \\ &\quad - \mu^2 e_o(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{x}^T(n) + \mu e_o(n)\mathbf{x}(n)\mathbf{v}^T(n) \\ &\quad - \mu^2 e_o(n)\mathbf{x}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n) + \mu^2 e_o^2(n)\mathbf{x}(n)\mathbf{x}^T(n). \end{aligned} \quad (2.32)$$

Reagrupando os termos de (2.32), tem-se

$$\begin{aligned} &\mathbf{v}(n+1)\mathbf{v}^T(n+1) \\ &= \mathbf{v}(n)\mathbf{v}^T(n) - \mu[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{v}^T(n) \\ &\quad + \mathbf{v}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)] + \mu^2[\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)] \\ &\quad + \mu^2 e_o^2(n)\mathbf{x}(n)\mathbf{x}^T(n) + \mu e_o(n)\{[\mathbf{x}(n)\mathbf{v}^T(n) - \mu\mathbf{x}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)] \\ &\quad + [\mathbf{x}(n)\mathbf{v}^T(n) - \mu\mathbf{x}(n)\mathbf{v}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)]^T\}. \end{aligned} \quad (2.33)$$

Determinando então o valor esperado de ambos os lados de (2.33) e com base na TI, obtém-se uma expressão recursiva descrevendo a evolução da matriz  $\mathbf{K}(n)$ . Assim,

$$\begin{aligned} \mathbf{K}(n+1) &= E\{\mathbf{v}(n+1)\mathbf{v}(n+1)^T\} = \mathbf{K}(n) - \mu[\mathbf{R}\mathbf{K}(n) + \mathbf{K}(n)\mathbf{R}] \\ &\quad + \mu^2\{\mathbf{R}\text{tr}[\mathbf{R}\mathbf{K}(n)] + 2\mathbf{R}\mathbf{K}(n)\mathbf{R}\} + \mu^2\mathbf{R}\mathbf{J}_{w\min}. \end{aligned} \quad (2.34)$$

Portanto, o comportamento da curva de aprendizagem do algoritmo LMS é descrito por (2.28) e (2.34).

### C. Análise em regime permanente do algoritmo LMS

Esta análise objetiva a determinação do EQM em excesso e do vetor de coeficientes do filtro adaptativo em regime permanente. Então, determinando o valor de (2.17) quando  $n \rightarrow \infty$  e assumindo que o filtro converge, tem-se

$$\mathbf{p} - \mathbf{R}\mathbf{w}(\infty) = 0 \quad (2.35)$$

tendo como única solução para o valor em regime permanente do vetor de coeficientes

$$\mathbf{w}(\infty) = \mathbf{R}^{-1}\mathbf{p} = \mathbf{w}_0. \quad (2.36)$$

Note que, para o caso de identificação de sistemas, o vetor de coeficientes do filtro  $\mathbf{w}(n)$  converge para a solução de Wiener, i.e.,  $\mathbf{w}_0$ .

Como determinado em [1], o EQM em excesso final  $J_{\text{ex}}(\infty)$  é obtido pela seguinte expressão:

$$J_{\text{ex}}(\infty) = J_{\min} \frac{\sum_{i=1}^N \frac{\mu\lambda_i}{2(1-\mu\lambda_i)}}{1 - \sum_{i=1}^N \frac{\mu\lambda_i}{2(1-\mu\lambda_i)}} \quad (2.37)$$

onde  $\lambda_i$  corresponde ao  $i$ -ésimo autovalor da matriz de autocorrelação  $\mathbf{R}$  do sinal de entrada.

Considerando  $\sum_{i=1}^N \lambda_i = \text{tr}(\mathbf{R})$  e valores do passo de adaptação pequenos, de tal forma que  $\mu\lambda_i \ll 1$ ,  $\forall i$ , tem-se

$$\sum_{i=1}^N \frac{\mu \lambda_i}{2(1 - \mu \lambda_i)} \cong \frac{\mu}{2} \text{tr}(\mathbf{R}). \quad (2.38)$$

Agora, substituindo (2.38) em (2.37), resulta em

$$J_{\text{ex}}(\infty) = J_{\text{min}} \frac{\frac{\mu}{2} \text{tr}(\mathbf{R})}{1 - \frac{\mu}{2} \text{tr}(\mathbf{R})}. \quad (2.39)$$

Novamente, considerando que o passo de adaptação seja pequeno, assume-se

$$\frac{\mu}{2} \text{tr}(\mathbf{R}) \ll 1. \quad (2.40)$$

Então, (2.39) pode ser aproximada por

$$J_{\text{ex}}(\infty) \cong \frac{\mu}{2} J_{\text{min}} \text{tr}(\mathbf{R}). \quad (2.41)$$

A expressão (2.41) determina, de forma aproximada, o valor final do EQM em excesso em regime permanente, considerando valores pequenos do passo de adaptação. Observa-se que tal resultado é linearmente dependente do valor do passo de adaptação.

### 2.3.4 Análise de Estabilidade do Algoritmo LMS

Existem algumas abordagens para determinação dos valores limites do passo de adaptação  $\mu$  que garantem a convergência do algoritmo LMS [1], [11].

A primeira delas considera a decomposição da matriz de autocorrelação do sinal de entrada dada por

$$\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T \quad (2.42)$$

onde  $\mathbf{Q}$  é a matriz dos autovetores de  $\mathbf{R}$  e  $\mathbf{\Lambda}$  é uma matriz diagonal contendo os autovalores de  $\mathbf{R}$ . Então, substituindo (2.18) e (2.42) em (2.12), obtém-se

$$\mathbf{v}(n+1) = (\mathbf{I} - \mu\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T)\mathbf{v}(n). \quad (2.43)$$

Agora, pré-multiplicando (2.43) por  $\mathbf{Q}^T$  e definindo  $\mathbf{v}'(n) = \mathbf{Q}^T \mathbf{v}(n)$  como a representação do vetor de erro nos coeficientes na base dos autovetores da matriz  $\mathbf{R}$ , (2.43) pode ser reescrita como

$$\mathbf{v}'(n+1) = (\mathbf{I} - \mu\mathbf{\Lambda})\mathbf{v}'(n). \quad (2.44)$$

Em (2.44), para cada elemento  $k$  do vetor  $\mathbf{v}'(n)$ , tem-se a seguinte equação recursiva:

$$v'_k(n+1) = (1 - \mu\lambda_k)v'_k(n) \quad (2.45)$$

que descreve o comportamento transitório de cada elemento do vetor de erro nos coeficientes, representado na base dos autovetores da matriz  $\mathbf{R}$ . O termo  $(1 - \mu\lambda_k)$ , em (2.45), é chamado modo de convergência. O algoritmo LMS possui  $N$  modos de convergência, sendo que o modo de maior magnitude é determinante na velocidade de convergência do algoritmo.

Assim, para que todos os componentes do vetor  $\mathbf{v}(n)$  decaiam a zero é necessário que

$$-1 < (1 - \mu\lambda_k) < 1, \quad \forall k. \quad (2.46)$$

Considerando ainda que  $\lambda_k$  seja real e positivo, tem-se

$$0 < \mu < \frac{2}{\lambda_k}, \quad \forall k. \quad (2.47)$$

Então, tomando-se o pior caso, i.e., considerando o maior autovalor da matriz  $\mathbf{R}$  em (2.47), os limites do valor do passo para a convergência do algoritmo LMS é restrito a

$$0 < \mu < \frac{2}{\lambda_{\max}}. \quad (2.48)$$

Adicionalmente, os limites de variação do passo de adaptação  $\mu$  que garantem a convergência do algoritmo também podem ser obtidos pela análise do EQM em regime permanente. Analisando o denominador de (2.37), conclui-se que, para operação estável do algoritmo, é necessário que

$$0 < \sum_{i=1}^N \frac{\mu\lambda_i}{2(1 - \mu\lambda_i)} < 1. \quad (2.49)$$

Resultando, agora, em uma condição mais restritiva de estabilidade do algoritmo LMS, dada por

$$0 < \mu < \frac{2}{3\text{tr}(\mathbf{R})}. \quad (2.50)$$

Portanto, para garantir a estabilidade do algoritmo LMS, adota-se neste trabalho (2.50) como os limites permitidos de variação para o passo de adaptação.

#### 2.4 Algoritmo LMS Normalizado – NLMS

Observa-se de (2.12) que variações na potência do sinal  $x(n)$  podem causar variações significativas no ajuste de  $\mathbf{w}(n)$ . Para evitar tal inconveniente, é incluído no algoritmo LMS o termo  $\mathbf{x}^T(n)\mathbf{x}(n)$  no denominador da parcela de ajuste da equação de adaptação do vetor de coeficientes. Assim,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{e(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \mathbf{x}(n). \quad (2.51)$$

Tal algoritmo é denominado algoritmo LMS normalizado (NLMS). Observa-se que o NLMS pode ser considerado um algoritmo LMS com passo de adaptação variável definido por

$$\mu(n) = \frac{\mu}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (2.52)$$

onde  $\mu(n)$  depende apenas do vetor de entrada  $\mathbf{x}(n)$  e do parâmetro de controle de adaptação  $\mu$ . A diferença do algoritmo NLMS para os demais algoritmos VSSLMS abordados neste trabalho é que, no NLMS, o passo variável não é ajustado considerando qualquer variável relacionada ao processo de convergência do algoritmo.

Para se evitar problemas de instabilidade quando  $\mathbf{x}^T(n)\mathbf{x}(n) \rightarrow 0$ , uma constante  $\varepsilon$  (parâmetro de regularização) de pequeno valor é incluída em (2.51). Dessa forma, tem-se

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{e(n)}{\mathbf{x}^T(n)\mathbf{x}(n) + \varepsilon} \mathbf{x}(n). \quad (2.53)$$

Em [46], são apresentados modelos estocásticos descrevendo o comportamento médio do algoritmo NLMS.

## 2.5 Algoritmos VSSLMS

O algoritmo LMS convencional considera um passo de adaptação fixo. A escolha desse parâmetro afeta sobremaneira o desempenho do algoritmo. Assim, observa-se de (2.41) e (2.45) que um valor elevado do passo de adaptação leva a uma velocidade de convergência maior associada também a um maior desajuste do algoritmo. Por outro lado, um valor pequeno do passo de adaptação resulta em um menor desajuste do algoritmo às custas de uma taxa de convergência mais baixa. Para obter alta velocidade de convergência aliada a um reduzido desajuste, são propostos na literatura um grande número de algoritmos VSSLMS [12]-[40]. A idéia central nesse tipo de algoritmo é utilizar um valor de passo de adaptação grande no início do processo de convergência (respeitado o limite para estabilidade) e gradualmente (segundo algum critério dado) ir reduzindo o valor do passo à medida que o regime permanente se aproxime.

Nos algoritmos VSSLMS a equação de adaptação do vetor de coeficientes opera com um passo variável função da variável independente  $n$ . Nesse caso, (2.12) é redefinida como

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n)e(n)\mathbf{x}(n). \quad (2.54)$$

Algumas estratégias utilizam passos individuais de adaptação, um para cada elemento do vetor de coeficientes. Assim, o passo de adaptação  $\mu(n)$  é substituído por uma matriz diagonal  $\mathbf{D}(n)$ , cuja diagonal principal é composta pelos passos individuais de adaptação. Dessa forma, (2.54) é modificada como segue:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + e(n)\mathbf{D}(n)\mathbf{x}(n) \quad (2.55)$$

com

$$\mathbf{D}(n) = \begin{bmatrix} \mu_0(n) & 0 & 0 & \dots & 0 \\ 0 & \mu_1(n) & 0 & \dots & 0 \\ 0 & 0 & \mu_2(n) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mu_{N-1}(n) \end{bmatrix}. \quad (2.56)$$

A utilização de um ou mais passos variáveis de adaptação pode então, nas mais diferentes aplicações, levar a uma melhoria significativa no desempenho do algoritmo LMS.

Então, devido à natureza dinâmica do passo de adaptação decorrente dos algoritmos VSSLMS, faz-se necessário um controle sobre a faixa de variação do passo, de modo a prevenir eventuais problemas de instabilidade [47]. Nos algoritmos VSSLMS, a limitação da variação do passo de adaptação para a garantia de estabilidade, tanto para passo comum quanto para passos individuais, deve satisfazer a seguinte desigualdade, adaptada de (2.50). Assim,

$$0 < \mu(n) < \frac{2}{3\text{tr}(\mathbf{R})}. \quad (2.57)$$

## 2.6 Conclusões

Este capítulo abordou os conceitos básicos e fundamentais relacionados ao algoritmo LMS, visando estabelecer uma base teórica adequada para este trabalho de pesquisa. Nele foi apresentada a derivação do algoritmo LMS assim como seu modelo estocástico, tanto para a evolução do vetor de coeficientes quanto para a curva de aprendizagem. Analisando o comportamento médio do vetor de coeficientes se observa que o algoritmo LMS apresenta (na média) resultado similar ao do algoritmo *steepest descent*. Também é apresentada uma análise de estabilidade do algoritmo LMS, visando definir os valores limites do passo de adaptação que assegurem a convergência do algoritmo.

As características gerais dos algoritmos NLMS e VSSLMS foram brevemente descritas. No próximo capítulo, as principais estratégias de ajuste do passo de adaptação variável em algoritmos LMS são apresentadas e classificadas.



## Classificação e Modelagem de Algoritmos VSSLMS

### 3.1 Introdução

Os algoritmos adaptativos de passo variável têm grande importância prática, visto que apresentam um melhor desempenho em relação aos de passo fixo, por possibilitarem a obtenção simultânea de elevadas taxas de convergência com um desajuste reduzido. Nesse contexto, a literatura técnica apresenta um grande número de algoritmos com tal característica, indicando ser essa uma linha de pesquisa bastante ativa na área de filtragem adaptativa.

Neste capítulo, é realizada uma breve descrição e uma classificação dos principais algoritmos LMS de passo variável (VSSLMS), baseada na idéia central de cada algoritmo [41]. Além disso, são desenvolvidos modelos estocásticos para três algoritmos de grande impacto. A classificação aqui proposta facilita o estudo dos algoritmos VSSLMS, padronizando a nomenclatura das diversas estratégias recentemente publicadas e possibilitando uma comparação considerando um cenário comum de aplicação. Para tal, são estabelecidas oito categorias, conforme o critério utilizado para ajuste do passo de adaptação. Tais critérios podem ser baseados no(a):

- Gradiente do erro quadrático.
- Erro quadrático.
- Autocorrelação do erro.
- Valor absoluto do erro.
- Normalização do vetor de erros.
- Ajuste proporcional aos valores absolutos do vetor de coeficientes.
- Teoria do passo ótimo.
- Otimização com restrição do ruído de medição.

Cada categoria aqui proposta é brevemente discutida e um resumo de cada algoritmo estudado é apresentado. Para todos os algoritmos estudados, o vetor de coeficientes do filtro é ajustado seja por (2.54) ou (2.55), conforme a utilização de um passo de adaptação comum ou passos individuais, respectivamente.

Neste capítulo também são apresentados modelos estocásticos de três algoritmos estudados. Os modelos estocásticos em filtragem adaptativa objetivam a determinação do comportamento médio das principais variáveis envolvidas na operação do algoritmo, possibilitando com isso, a predição de seu desempenho em uma dada condição de operação. Além do mais, os modelos estocásticos permitem estabelecer relações de causa e efeito entre parâmetros de cada algoritmo e seus principais índices de desempenho.

Especificamente no caso dos algoritmos VSSLMS, a modelagem estocástica constitui-se em uma ferramenta fundamental, auxiliando no processo de escolha de um determinado algoritmo para uma dada aplicação. Como existe um grande número de estratégias usadas em algoritmos VSSLMS, a escolha de um algoritmo se torna uma tarefa bastante difícil quando não se dispõe de modelos estocásticos para tais algoritmos.

### 3.2 Aspectos Relevantes para a Modelagem de Algoritmos VSSLMS

A modelagem estocástica de algoritmos VSSLMS proposta neste trabalho, visa determinar o comportamento médio do vetor de coeficientes e/ou do vetor de erro nos coeficientes e a correspondente curva de aprendizagem. Em alguns casos, também é realizada uma análise em regime permanente.

Os modelos dos algoritmos VSSLMS aqui discutidos são determinados considerando as seguintes hipóteses simplificativas:

- i) Teoria da Independência (TI).
- ii) O passo variável  $\mu(n)$  é assumido independente de qualquer outro sinal no sistema. Essa hipótese é válida sob condições de adaptação lenta.

A determinação dos modelos estocásticos necessita do cálculo dos valores esperados do passo variável  $E[\mu(n)]$  e do quadrado do passo variável  $E[\mu^2(n)]$ . Em alguns modelos, utiliza-se ainda uma terceira hipótese, i.e.,

$$\text{iii) } E[\mu^2(n)] = E^2[\mu(n)]. \quad (3.1)$$

A hipótese (iii), quando utilizada, é devidamente verificada por resultados de simulação.

Após a determinação de  $E[\mu(n)]$  e  $E[\mu^2(n)]$  e considerando as hipóteses (i) e (ii), os modelos estocásticos para o comportamento médio do vetor de coeficientes podem ser obtidos considerando a seguinte expressão:

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + E[\mu(n)]\{\mathbf{p} - \mathbf{R}E[\mathbf{w}(n)]\} \quad (3.2)$$

e para a curva de aprendizagem

$$J(n) = E[e^2(n)] = \sigma_\eta^2 + \text{tr}[\mathbf{R}\mathbf{K}(n)] \quad (3.3)$$

com a evolução de  $\mathbf{K}(n)$  dada por

$$\begin{aligned} \mathbf{K}(n+1) &= \mathbf{K}(n) - E[\mu(n)][\mathbf{R}\mathbf{K}(n) + \mathbf{K}(n)\mathbf{R}] \\ &\quad + E[\mu^2(n)][\mathbf{R}\text{tr}(\mathbf{R}\mathbf{K}(n)) + 2\mathbf{R}\mathbf{K}(n)\mathbf{R}] \\ &\quad + E[\mu^2(n)]\mathbf{R}\sigma_\eta^2. \end{aligned} \quad (3.4)$$

Portanto, a diferença entre os modelos dos diversos algoritmos VSSLMS se encontra na determinação dos valores esperados  $E[\mu(n)]$  e  $E[\mu^2(n)]$ .

### 3.3 Algoritmos Baseados no Gradiente do Erro Quadrático

Nesta categoria, são incluídos os algoritmos VSSLMS cujo ajuste do passo de adaptação é realizado de maneira proporcional ao gradiente do valor quadrático do erro de estimação. Alguns algoritmos consideram o gradiente em relação ao vetor de coeficientes, enquanto em outros é utilizado o gradiente em relação ao passo variável de adaptação.

Os primeiros trabalhos que utilizaram essa abordagem baseavam-se apenas nas trocas de sinais, observadas nos componentes do gradiente do erro quadrático a cada iteração [12]. Nesses algoritmos, a frequência com que essas trocas de sinais ocorrem é usada como medida de proximidade do ponto ótimo na superfície de desempenho. Posteriormente, foram propostos algoritmos que utilizavam o valor numérico do gradiente para efetuar um ajuste proporcional do passo de adaptação.

A idéia central dos algoritmos dessa categoria é que quanto maior o valor do gradiente do erro quadrático, maior a distância do EQM mínimo na superfície de desempenho e, portanto, maior deve ser o passo

de adaptação para que a convergência seja acelerada. Por outro lado, quanto menor for o gradiente, menor deve ser o valor do passo de forma a reduzir o desajuste em regime permanente. Em todos os casos, utiliza-se uma estimativa do gradiente; sendo assim, a presença de ruído afeta essa estimativa levando a um aumento do desajuste do algoritmo. Os principais algoritmos incluídos nessa categoria são brevemente discutidos a seguir.

### 3.3.1 Algoritmo de Richards [13]

Este algoritmo opera com passos individuais de adaptação, considerando a potência média do gradiente do sinal de erro ao quadrado em relação ao vetor de coeficientes para o cálculo do passo de adaptação. Assim, a matriz diagonal de passos de adaptação é obtida a partir da potência dos componentes da estimativa do vetor gradiente, o qual é dado por

$$\hat{\nabla}(n) = \mathbf{g}(n) = \begin{bmatrix} \frac{\partial e^2(n)}{\partial w_0} \\ \vdots \\ \frac{\partial e^2(n)}{\partial w_{N-1}} \end{bmatrix} = -2e(n)\mathbf{x}(n). \quad (3.5)$$

Para obter o valor médio do gradiente, é utilizado um filtro auto-regressivo, expresso como

$$E\{g_i(n+1)\} = \beta E\{g_i(n)\} + (1-\beta)e(n)x(n-i) \quad (3.6)$$

onde  $g_i(n)$  representa o  $i$ -ésimo elemento do vetor gradiente estimado e  $\beta$  é um parâmetro que controla a seletividade do filtro.

Dessa forma, a matriz  $\mathbf{D}(n)$  é então dada por

$$\mathbf{D}(n) = \gamma \begin{bmatrix} E^2[g_0(n)] & 0 & \cdots & 0 \\ 0 & E^2[g_1(n)] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E^2[g_{N-1}(n)] \end{bmatrix} \quad (3.7)$$

onde  $\gamma$  é um parâmetro positivo de controle do passo de adaptação.

### 3.3.2 Algoritmo de Mathews [14]

A idéia central deste algoritmo é ajustar o passo de adaptação de forma proporcional ao gradiente do sinal de erro ao quadrado em relação aos passos anteriores. A equação de ajuste do passo de adaptação é

$$\mu(n) = \mu(n-1) - \frac{\rho}{2} \frac{\partial e^2(n)}{\partial \mu(n-1)} \quad (3.8)$$

podendo ser alternativamente escrita como

$$\mu(n) = \mu(n-1) - \frac{\rho}{2} \frac{\partial^T e^2(n)}{\partial \mathbf{w}(n)} \frac{\partial \mathbf{w}(n)}{\partial \mu(n-1)}. \quad (3.9)$$

Após a determinação das derivadas parciais, chega-se à expressão final de ajuste do passo comum de adaptação do algoritmo. Assim,

$$\mu(n) = \mu(n-1) + \rho e(n) e(n-1) \mathbf{x}^T(n-1) \mathbf{x}(n) \quad (3.10)$$

onde  $\rho$  é um parâmetro positivo de controle do passo de adaptação. Na abordagem de Mathews, é também possível utilizar passos individuais para cada coeficiente do filtro, resultando na seguinte equação de ajuste:

$$\mu_i(n) = \mu_i(n-1) + \rho e(n) e(n-1) x_i(n) x_i(n-1), \quad i = 0, \dots, N-1. \quad (3.11)$$

O algoritmo de Mathews é um dos mais representativos desta categoria, uma vez que vários algoritmos recentemente propostos são derivados a partir de sua formulação inicial.

### 3.3.3 Algoritmo de Wee-Peng Ang [19]

Este algoritmo, baseado no algoritmo de Mathews, utiliza uma forma recursiva para determinação do termo  $\frac{\partial \mathbf{w}(n)}{\partial \mu(n-1)}$  em (3.9), utilizando um filtro auto-regressivo, dado por

$$\psi_i(n) = \frac{\partial w_i(n)}{\partial \mu_i(n-1)} = \alpha \psi_i(n-1) + x_i(n-1) e(n-1) \quad (3.12)$$

onde  $\alpha$  é um parâmetro de controle de seletividade do filtro. Assim, a equação de ajuste dos passos individuais de adaptação é

$$\mu_i(n) = \mu_i(n-1) + \rho e(n) x_i(n) \psi_i(n) \quad (3.13)$$

onde  $\rho$  é um parâmetro de ajuste do passo de adaptação do algoritmo.

Observa-se que o algoritmo de Wee-Peng realiza uma suavização da estimativa ruidosa do gradiente considerado no algoritmo de Mathews, gerando em contrapartida um aumento de complexidade computacional.

Note que o algoritmo de Mathews pode ser obtido a partir do algoritmo de Wee-Peng Ang fazendo  $\alpha = 0$  em (3.12). Neste último algoritmo, também é possível a utilização de um passo comum de adaptação, sendo que, nesse caso, as expressões de ajuste são dadas por

$$\boldsymbol{\psi}(n) = \alpha \boldsymbol{\psi}(n-1) + e(n-1) \mathbf{x}(n-1) \quad (3.14)$$

e

$$\mu(n) = \mu(n-1) + \rho e(n) \mathbf{x}^T(n) \boldsymbol{\psi}(n) \quad (3.15)$$

onde  $\alpha$  e  $\rho$  são parâmetros de controle positivos.

### 3.3.4 Algoritmo de Farhang [16]

Este algoritmo utiliza passos individuais de adaptação, os quais são obtidos a partir de modificações do algoritmo de Mathews. Nessa abordagem, o valor do passo anterior de adaptação é multiplicado pela estimativa do gradiente do sinal de erro ao quadrado em relação ao passo variável de adaptação. Além disso, são utilizados parâmetros  $\delta_i$  individuais para cada passo de adaptação, no lugar da constante  $\rho$  utilizada no algoritmo de Mathews. Esses parâmetros controlam individualmente a taxa de convergência de cada passo de adaptação. Dessa forma, a expressão de ajuste de cada passo de adaptação é escrita como

$$\mu_i(n+1) = [1 + \delta_i e(n) x_i(n) e(n-1) x_i(n-1)] \mu_i(n) \quad (3.16)$$

onde o parâmetro de controle  $\delta_i$  é obtido através da normalização de uma constante  $\delta_o$  com respeito à potência média do gradiente. Assim,  $\delta_i$  é dado por

$$\delta_i = \frac{\delta_o}{\varepsilon_i} \quad (3.17)$$

onde  $\delta_0$  é uma constante positiva e  $\varepsilon_i$  denota o valor médio de  $|e(n)x_i(n)|^2$ .

### 3.3.5 Algoritmo de Okello [18]

Neste algoritmo, o ajuste do passo variável comum também é baseado na potência da estimativa do gradiente do sinal de erro ao quadrado em relação ao vetor de coeficientes. Dessa forma, a equação de atualização do passo variável é dada por

$$\mu(n) = \alpha\mu(n-1) + \sigma \sum_{i=0}^{N-1} g_i^2(n) \quad (3.18)$$

com

$$g_i(n) = \beta g_i(n-1) + e(n)x_i(n) \quad (3.19)$$

onde  $g_i(n)$  é a estimativa do gradiente do erro quadrático em relação ao  $i$ -ésimo componente do vetor de coeficientes, sendo  $\alpha$  e  $\beta$  parâmetros de controle do processo de adaptação.

#### A. Modelo estocástico

Uma análise estatística do algoritmo de Okello [45] é proposta neste trabalho de pesquisa e apresentada a seguir.

O ajuste do passo variável, em (3.18), pode ser expresso em termos do vetor gradiente  $\mathbf{g}(n)$ , obtido como

$$\mu(n) = \alpha\mu(n-1) + \sigma \mathbf{g}^T(n)\mathbf{g}(n) \quad (3.20)$$

com o vetor gradiente dado por

$$\mathbf{g}(n) = \beta\mathbf{g}(n-1) + e(n)\mathbf{x}(n). \quad (3.21)$$

Agora, com o objetivo de se determinar  $E[\mu(n)]$ , toma-se o valor esperado de ambos os lados de (3.20), resultando em

$$E[\mu(n)] = \alpha E[\mu(n-1)] + \sigma E[\mathbf{g}^T(n)\mathbf{g}(n)]. \quad (3.22)$$

Para resolver (3.22), o segundo termo no lado direito deve ser determinado. Assim, utilizando (3.21), obtém-se

$$\begin{aligned} \mathbf{g}^T(n)\mathbf{g}(n) &= \beta^2 \mathbf{g}^T(n-1)\mathbf{g}(n-1) + \beta e(n)\mathbf{g}^T(n-1)\mathbf{x}(n) \\ &\quad + \beta e(n)\mathbf{x}^T(n)\mathbf{g}(n-1) + e^2(n)\mathbf{x}^T(n)\mathbf{x}(n). \end{aligned} \quad (3.23)$$

Então, rearranjando (3.23), tem-se

$$\begin{aligned} \mathbf{g}^T(n)\mathbf{g}(n) &= \beta^2 \mathbf{g}^T(n-1)\mathbf{g}(n-1) + 2\beta e(n)\mathbf{g}^T(n-1)\mathbf{x}(n) \\ &\quad + e^2(n)\mathbf{x}^T(n)\mathbf{x}(n). \end{aligned} \quad (3.24)$$

Finalmente, determinado o valor esperado de ambos os lados de (3.24),  $E[\mathbf{g}^T(n)\mathbf{g}(n)]$  pode ser obtido de forma recursiva por

$$\begin{aligned} E[\mathbf{g}^T(n)\mathbf{g}(n)] &= \beta^2 E[\mathbf{g}^T(n-1)\mathbf{g}(n-1)] \\ &\quad + 2\beta E[e(n)\mathbf{x}^T(n)\mathbf{g}(n-1)] + E[e^2(n)\mathbf{x}^T(n)\mathbf{x}(n)]. \end{aligned} \quad (3.25)$$

Em (3.25), o termo  $E[e^2(n)\mathbf{x}^T(n)\mathbf{x}(n)]$  pode ser aproximado pelo uso da TI por

$$\begin{aligned} E[e^2(n)\mathbf{x}^T(n)\mathbf{x}(n)] &\cong E[e^2(n)]E[\mathbf{x}^T(n)\mathbf{x}(n)] \\ &\cong J(n)\text{tr}(\mathbf{R}) \end{aligned} \quad (3.26)$$

onde  $J(n)$  representa a curva de aprendizagem do algoritmo. O termo  $E[e(n)\mathbf{x}^T(n)\mathbf{g}(n-1)]$  é obtido sob a condição de adaptação lenta, resultando na seguinte aproximação:

$$\begin{aligned} E[e(n)\mathbf{x}^T(n)\mathbf{g}(n-1)] &\cong E[e(n)\mathbf{x}^T(n)]E[\mathbf{g}(n-1)] \\ &\cong \{\mathbf{p} - \mathbf{R}E[\mathbf{w}(n)]\}^T E[\mathbf{g}(n-1)]. \end{aligned} \quad (3.27)$$

Então, para finalizar,  $E[\mathbf{g}(n-1)]$  pode ser obtido, de forma recursiva, tomando o valor esperado de ambos os lados de (3.21). Assim,

$$E[\mathbf{g}(n)] = \beta E[\mathbf{g}(n-1)] + \{\mathbf{p} - \mathbf{R}E[\mathbf{w}(n)]\}. \quad (3.28)$$

Neste modelo, utilizou-se a suposição (3.1) para a determinação de  $E[\mu^2(n)]$  e obtenção da curva de aprendizagem dada por (3.3).

*B. Análise em regime permanente*

A análise em regime permanente visa a determinação dos valores das principais variáveis envolvidas na operação de algoritmos VSSLMS quando  $n \rightarrow \infty$ . Essa análise está disponível na literatura para o algoritmo de Mathews [14], sendo também desenvolvida neste trabalho para os algoritmos de Okello [18] e Wei [37]. As análises em regime permanente apresentadas neste trabalho são elaboradas assumindo que o passo de adaptação e o vetor de coeficientes convergem.

Para o algoritmo de Okello [18], são determinados a seguir os valores em regime permanente da estimativa do vetor gradiente  $\mathbf{g}(n)$ , do passo variável  $\mu(n)$ , do vetor de coeficientes  $\mathbf{w}(n)$  e do EQM.

Vetor gradiente:

Considere-se (3.28) para  $n \rightarrow \infty$ , dessa forma

$$\begin{aligned} \mathbf{g}(\infty) &= \beta \mathbf{g}(\infty) + \{\mathbf{p} - \mathbf{RE}[\mathbf{w}(\infty)]\} \\ &= \frac{\mathbf{p} - \mathbf{RE}[\mathbf{w}(\infty)]}{(1 - \beta)} \end{aligned} \quad (3.29)$$

onde  $\mathbf{w}(\infty)$  é o valor final do vetor de coeficientes.

Tomando (3.25) para  $n \rightarrow \infty$ , tem-se

$$\mathbf{g}^T(\infty)\mathbf{g}(\infty) = \frac{2\beta J(\infty)\text{tr}(\mathbf{R})}{(1 - \beta^2)}. \quad (3.30)$$

Passo de adaptação:

O valor final do passo de adaptação pode ser obtido substituindo (3.30) em (3.22) para  $n \rightarrow \infty$ , resultando em

$$\mu(\infty) = \frac{2\beta\sigma J(\infty)\text{tr}(\mathbf{R})}{(1 - \beta^2)(1 - \alpha)}. \quad (3.31)$$

Vetor de coeficientes:

Considerando que  $\mathbf{w}(n)$  converge, seu valor final obtido de (3.2) para  $n \rightarrow \infty$  é dado por

$$\mathbf{w}(\infty) = \mathbf{w}(\infty) + \mu(\infty) [\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)]. \quad (3.32)$$

Agora, desenvolvendo (3.32)

$$\mu(\infty)[\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)] = 0 \quad (3.33)$$

que resulta em

$$[\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)] = 0 \quad (3.34)$$

tem-se finalmente como única solução

$$\mathbf{w}(\infty) = \mathbf{R}^{-1}\mathbf{p}. \quad (3.35)$$

### Erro Quadrático Médio:

Considerando que o passo de adaptação variável  $\mu(n)$  converge para  $\mu(\infty)$ , pode-se utilizar (2.41), válida para valores pequenos do passo de adaptação, para a determinação do valor do EQM em regime permanente. Assim,

$$J_{\text{ex}}(\infty) = \frac{\mu(\infty)J_{\text{min}}}{2} \text{tr}(\mathbf{R}) = \frac{\beta\sigma J(\infty)}{(1-\beta^2)(1-\alpha)} \sigma_{\eta}^2 \text{tr}^2(\mathbf{R}). \quad (3.36)$$

Como  $J_{\text{ex}}(\infty) = J(\infty) - J_{\text{min}}$  e  $J_{\text{min}} = \sigma_{\eta}^2$ , (3.36) pode ser reescrita como

$$J(\infty) = \frac{\sigma_{\eta}^2}{1 - \frac{\beta\sigma}{2(1-\beta^2)(1-\alpha)} \sigma_{\eta}^2 \text{tr}^2(\mathbf{R})}. \quad (3.37)$$

### *C. Resultados de simulação*

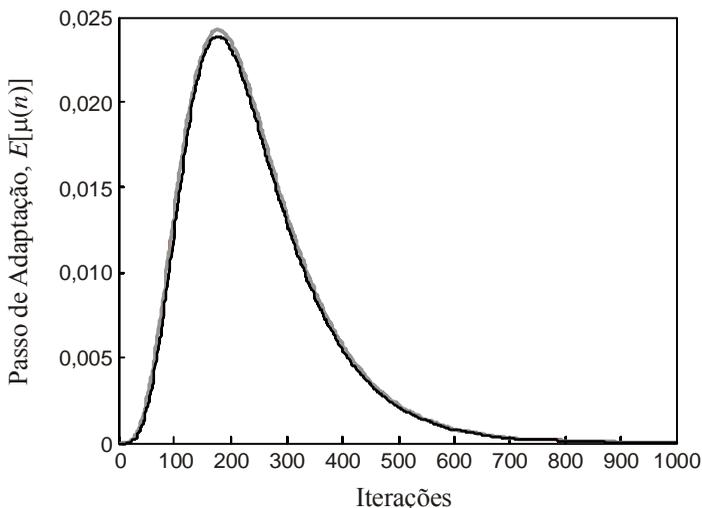
Aqui são apresentados os resultados de simulação numérica, verificando a precisão do modelo desenvolvido para o algoritmo de Okello [18].

Considera-se um problema de identificação de sistema, utilizando a planta dada por  $\mathbf{w}_0 = [0,2 \ 0,3 \ 0,5 \ 0,8 \ 0,9]$ , para sinais de entrada branco e correlacionado. Para ambos os casos, os sinais de entrada são gaussianos com média zero e variância  $\sigma_x^2 = 1$ , e a variância do ruído branco de medição é  $\sigma_{\eta}^2 = 0,01$  (SNR = 20 dB). O sinal correlacionado é obtido por um processo AR(1), dado por  $x(n) = 0,6x(n-1) + u(n)$

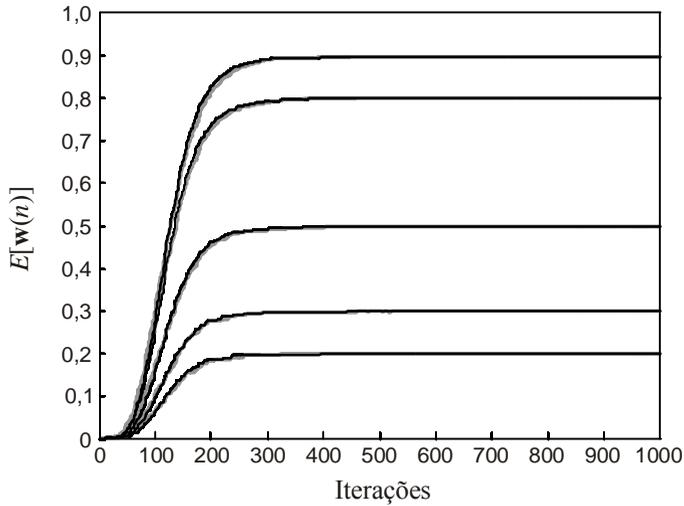
com  $u(n)$  sendo um ruído branco gaussiano com  $\sigma_u^2 = 0,65$ . Para o caso de sinal correlacionado, a dispersão dos autovalores da matriz de autocorrelação do sinal de entrada é  $\chi = 9,7$ . Os parâmetros dos algoritmos são ajustados de maneira a se obter o mesmo EQM em regime permanente ( $-20$  dB).

Dois exemplos apresentam os resultados de simulação para o modelo estocástico do algoritmo de Okello [45]. O Exemplo 3.1 considera sinal de entrada branco e o Exemplo 3.2, um sinal de entrada correlacionado.

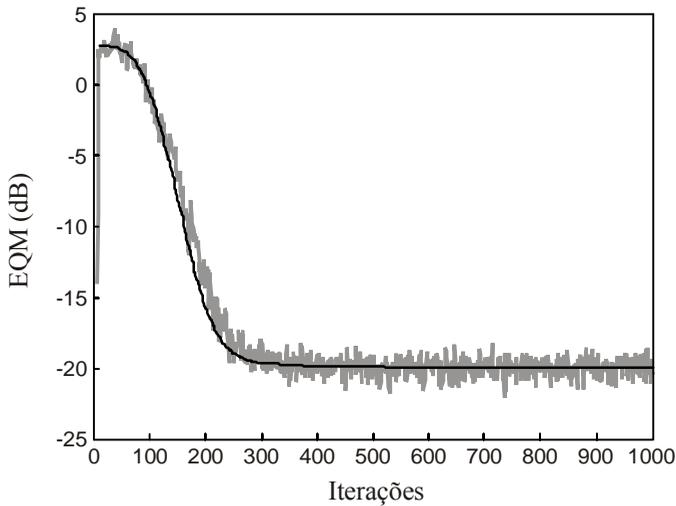
**Exemplo 3.1:** Neste exemplo, é usado sinal de entrada branco. As Figuras 3.1, 3.2, 3.3 e 3.4 mostram, respectivamente, o comportamento do passo de adaptação, do vetor de coeficientes, do EQM e do vetor gradiente estimado. Uma comparação dos resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto é apresentada.



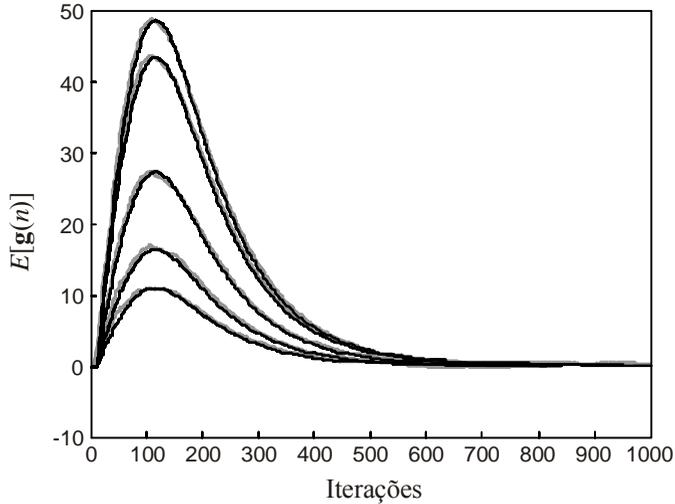
**Figura 3.1.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Okello [18] para sinal de entrada branco.



**Figura 3.2.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Okello [18] para sinal de entrada branco.



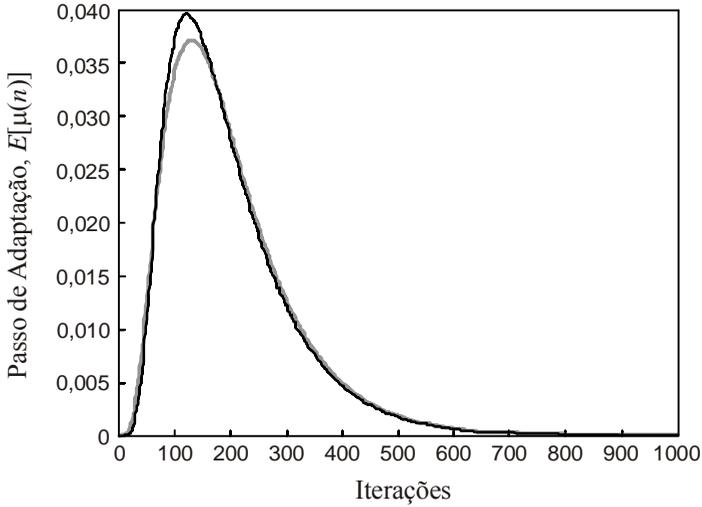
**Figura 3.3.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Okello [18] para sinal de entrada branco.



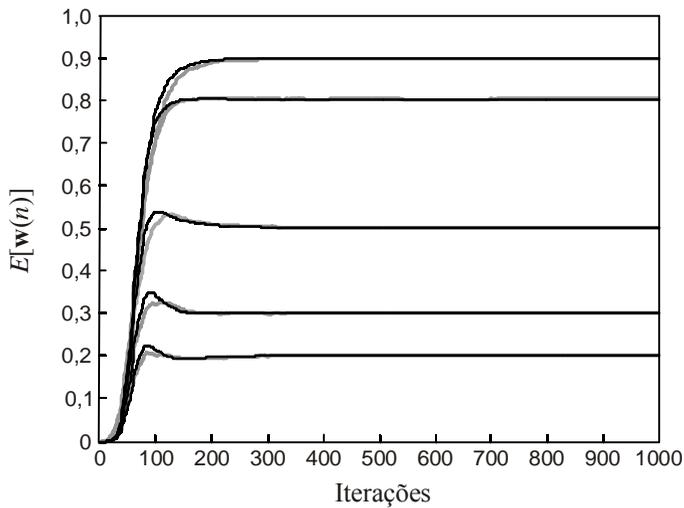
**Figura 3.4.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor gradiente estimado  $\mathbf{g}(n)$  do algoritmo de Okello [18] para sinal de entrada branco.

Observa-se das Figuras 3.1 a 3.4 um casamento muito bom entre os resultados obtidos a partir da simulação de MC e aqueles obtidos pelo modelo proposto do algoritmo de Okello [18], para sinal de entrada branco.

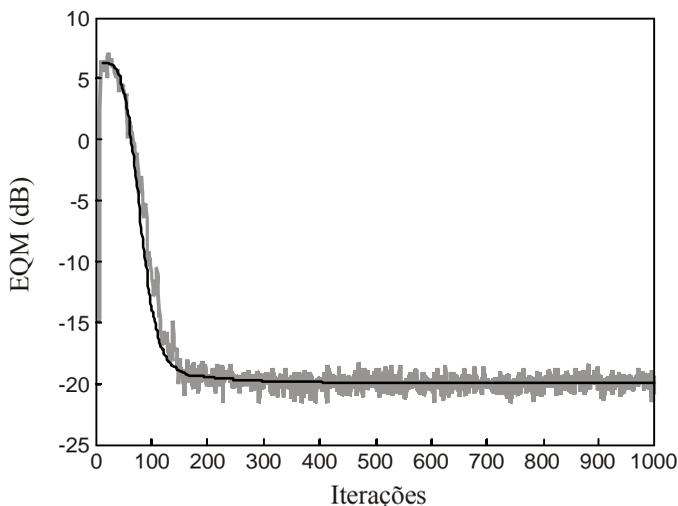
Exemplo 3.2: Neste exemplo, é considerado um sinal de entrada correlacionado. As Figuras 3.5, 3.6, 3.7 e 3.8 mostram, respectivamente, o comportamento do passo de adaptação, vetor de coeficientes, EQM e vetor gradiente estimado. Assim, é possível fazer uma comparação entre os resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto. A Figura 3.9 verifica por simulação a Aproximação (3.1), comparando as evoluções de  $E[\mu^2(n)]$  com  $E^2[\mu(n)]$ .



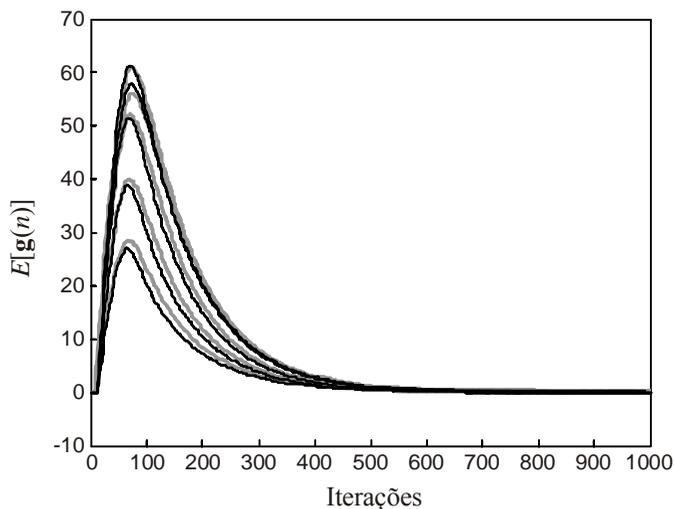
**Figura 3.5.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Okello [18] para sinal de entrada correlacionado.



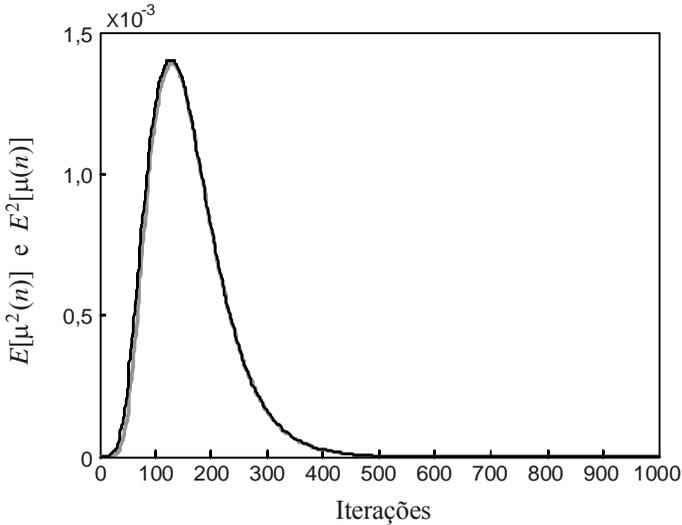
**Figura 3.6.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Okello [18] para sinal de entrada correlacionado.



**Figura 3.7.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Okello [18] para sinal de entrada correlacionado.



**Figura 3.8.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor gradiente estimado  $\mathbf{g}(n)$  do algoritmo de Okello [18] para sinal de entrada correlacionado.



**Figura 3.9.** Evolução de  $E[\mu^2(n)]$  (cinza) e  $E^2[\mu(n)]$  (preto) para o algoritmo de Okello.

Para o cenário considerado, observa-se a partir das Figuras 3.5 a 3.8 um casamento muito bom entre os resultados de simulação e do modelo proposto para o algoritmo de Okello [18], considerando um sinal de entrada correlacionado. As aproximações utilizadas no desenvolvimento do modelo estatístico justificam os pequenos desvios observados nas Figuras 3.5 a 3.8. Adicionalmente, a Figura 3.9 valida por simulação a Aproximação (3.1) para uma condição de adaptação lenta.

### 3.3.6 Algoritmo de Shin [20]

Esse algoritmo, semelhante ao algoritmo de Okello [18], utiliza uma estimativa normalizada do vetor gradiente do sinal de erro ao quadrado em relação ao vetor de coeficientes, obtida por

$$\mathbf{p}(n) = \alpha \mathbf{p}(n-1) + (1-\alpha) \frac{e(n)\mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (3.38)$$

onde o parâmetro  $\alpha$  controla a seletividade do filtro. O passo variável é então atualizado com base na seguinte expressão:

$$\mu(n) = \frac{\mu \mathbf{p}^T(n) \mathbf{p}(n)}{\mathbf{p}^T(n) \mathbf{p}(n) + c} \quad (3.39)$$

onde  $\mu$  e  $c$  são parâmetros de ajuste do algoritmo.

O algoritmo de Shin (aqui resumido) é um caso particular de um algoritmo de projeções afins de passo variável [20].

### 3.4 Algoritmos Baseados no Erro Quadrático

Os algoritmos desta categoria ajustam o passo utilizando o valor do sinal de erro ao quadrado. Pela própria natureza de tal ajuste, esses algoritmos possuem, em geral, uma forte dependência do ruído de medição. Assim, as diferentes estratégias propostas para essa classe de algoritmos buscam minimizar tal problema. Geralmente, o desempenho dessa família de algoritmos é afetado por ambientes com baixa razão sinal-ruído.

#### 3.4.1 Algoritmo de Kwong [22]

Neste algoritmo, o ajuste do passo de adaptação é controlado pelo sinal de erro ao quadrado, conforme a seguinte expressão:

$$\mu(n+1) = \alpha \mu(n) + \gamma e^2(n) \quad (3.40)$$

onde  $\alpha$  e  $\gamma$  são parâmetros de controle de ajuste do passo.

O algoritmo de Kwong é referência para todas as demais estratégias dessa categoria. A idéia principal para fundamentar esse tipo de algoritmo é que um erro de predição grande leva a um aumento no valor do passo, produzindo, assim, uma convergência mais rápida. Por outro lado, um valor reduzido de erro resulta em um valor de passo menor e, conseqüentemente, em um baixo desajuste.

## 3.4.2 Algoritmo de Nakanishi [23]

O ajuste do passo de adaptação desse algoritmo é baseado no erro quadrático instantâneo e na norma  $l_\infty$  [23] do vetor de erro quadrático, segundo a seguinte expressão:

$$\mu(n) = \mu \left[ 1 - \frac{e^2(n)_{\max}}{\alpha e^2(n)} \right] \quad (3.41)$$

onde  $e^2(n)_{\max}$  é o maior valor de erro quadrático entre a primeira e a  $n$ -ésima iteração, com  $\alpha$  e  $\mu$  parâmetros de ajuste do passo de adaptação.

## 3.4.3 Algoritmo de Costa [24]

Determinado a partir de uma modificação no algoritmo de Kwong, esse algoritmo é baseado na seguinte equação de recorrência:

$$\mu(n) = \alpha\mu(n-1) + \gamma [k \mathbf{x}^T(n)\mathbf{x}(n) - 1]e^2(n), \quad k = \frac{1}{N\sigma_x^2} \quad (3.42)$$

onde  $\alpha$  e  $\gamma$  são parâmetros de ajuste do passo de adaptação.

A inserção do fator  $[k \mathbf{x}^T(n)\mathbf{x}(n) - 1]$  em (3.40) para obter (3.42) tem por objetivo reduzir a influência do ruído de medição no ajuste do passo de adaptação, observada no algoritmo de Kwong. A motivação desse algoritmo pode ser melhor compreendida analisando seu modelo analítico [24].

## 3.4.4 Algoritmo de Benesty [25]

Observa-se que a maioria dos algoritmos VSSLMS necessita do ajuste de um ou mais parâmetros para a sua operação. O ajuste destes parâmetros nem sempre é uma tarefa fácil, sendo que na maioria dos casos é realizado de forma heurística. Esse algoritmo, teoricamente, não necessita do ajuste de parâmetros para a sua correta operação. A equação de atualização do passo variável é

$$\mu(n) = \frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \left[ 1 - \frac{\sigma_\eta}{\sigma_e(n)} \right] \quad (3.43)$$

onde  $\sigma_\eta$  é obtido através do conhecimento prévio ou a partir da estimativa da variância do ruído aditivo  $\sigma_\eta^2$ . O parâmetro  $\sigma_e(n)$  é determinado através de uma estimativa da potência do erro  $\sigma_e^2(n)$ , dada por

$$\hat{\sigma}_e^2(n) = \lambda \hat{\sigma}_e^2(n-1) + (1-\lambda)e^2(n). \quad (3.44)$$

onde  $\lambda$  é uma constante próxima, porém menor do que a unidade.

### 3.5 Algoritmos Baseados na Autocorrelação do Erro

Para esta classe de algoritmos, o valor do passo de adaptação é ajustado baseado na função de autocorrelação do sinal de erro. Tal medida se mostra um bom indicativo de proximidade do EQM mínimo. Assim, com a utilização da função de autocorrelação do erro, o algoritmo pode efetivamente realizar o ajuste do valor do passo de adaptação, melhorando a sua robustez ao ruído de medição não correlacionado (ruído branco).

#### 3.5.1 Algoritmo de Aboulnasr [28]

Como principal representante desta categoria, considera-se neste trabalho de pesquisa o algoritmo de Aboulnasr [28]. Para este algoritmo, o passo de adaptação é ajustado considerando o quadrado da estimativa da autocorrelação entre  $e(n)$  e  $e(n-1)$ , obtida através de um filtro auto-regressivo, dado por

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1) \quad (3.45)$$

onde  $\beta$  é um parâmetro que controla a seletividade do filtro. A equação utilizada para o ajuste do passo é

$$\mu(n+1) = \alpha \mu(n) + \gamma p^2(n) \quad (3.46)$$

onde  $\alpha$  e  $\gamma$  são parâmetros de controle do algoritmo.

A utilização de  $p(n)$  na atualização de  $\mu(n)$  atende a dois objetivos: a função de autocorrelação do erro é, em geral, uma medida adequada da proximidade do ponto ótimo e, em segundo lugar, fornece

maior robustez ao ruído de medição não correlacionado, se comparada à função erro quadrático utilizada no algoritmo de Kwong [22].

### A. Modelo estocástico

Aqui é apresentada uma análise estocástica do comportamento do passo variável de adaptação para o algoritmo de Aboulnasr [28]. Esta análise, desenvolvida neste trabalho de pesquisa, visa mostrar que tal algoritmo é fortemente dependente da variância do ruído de medição. Tal dependência é observada mesmo quando o ruído de medição é uma sequência descorrelacionada (ruído branco).

Tomando o valor esperado de ambos os lados de (3.46), tem-se

$$E[\mu(n+1)] = \alpha E[\mu(n)] + \gamma E[p^2(n)]. \quad (3.47)$$

Assumindo agora  $\mu(0) = 0$  e conseqüentemente  $E[\mu(0)] = 0$ , pode-se reescrever (3.47) como

$$E[\mu(n+1)] = \gamma \sum_{\ell=0}^n \alpha^\ell E[p^2(n-\ell)]. \quad (3.48)$$

Para se obter  $E[\mu(n+1)]$  é necessário determinar uma expressão para  $E[p^2(n)]$ . Tal determinação é feita no Apêndice A.

Agora, substituindo (A.26) em (3.48), pode-se aproximar o comportamento médio do passo de adaptação por

$$\begin{aligned} & E[\mu(n+1)] \\ & \cong \gamma(1-\beta)^2 \sum_{\ell=0}^n \alpha^\ell \left( \sum_{j=0}^{n-l-1} \beta^{2j} \{ \sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-l-j-1) + \mathbf{R}\mathbf{K}(n-l)] \} \right. \\ & \left. + \sum_{j=1}^{n-l-1} \beta^{2j-1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-l-j+1)] + \sum_{j=0}^{n-l-2} \beta^{2j+1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-l-j)] \right). \end{aligned} \quad (3.49)$$

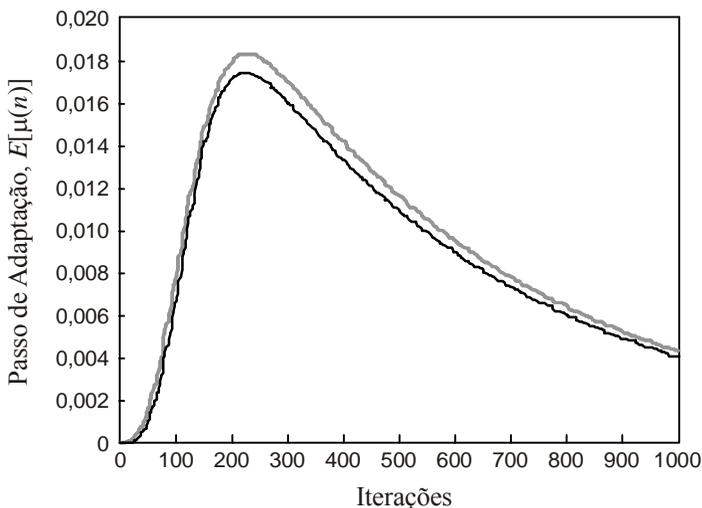
A partir dessa expressão, pode-se verificar a dependência do passo de adaptação em relação à variância do ruído de medição para o algoritmo proposto em [28]. É observado também que quanto maior o valor da variância  $\sigma_\eta^2$ , maior será o valor do passo de adaptação. Assim, quando qualquer alteração na potência do ruído de medição ocorrer, uma

mudança no parâmetro  $\gamma$  deve ser realizada, de forma a se obter um apropriado desempenho do algoritmo. Para obtenção da curva de aprendizagem do algoritmo, considera-se a Aproximação (3.1).

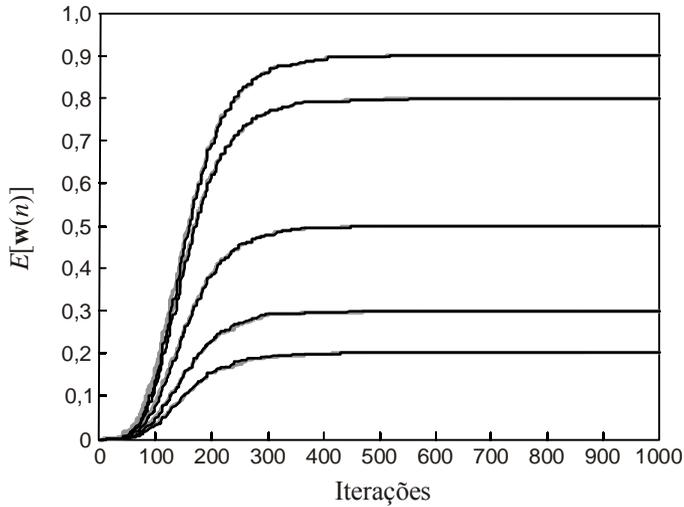
### B. Resultados de simulação

Dois exemplos apresentam os resultados de simulação para o modelo estocástico do algoritmo de Aboulnasr [28]. O cenário utilizado é o mesmo descrito nos Exemplos 3.1 e 3.2. O Exemplo 3.3 considera sinal de entrada branco e o Exemplo 3.4, sinal de entrada correlacionado.

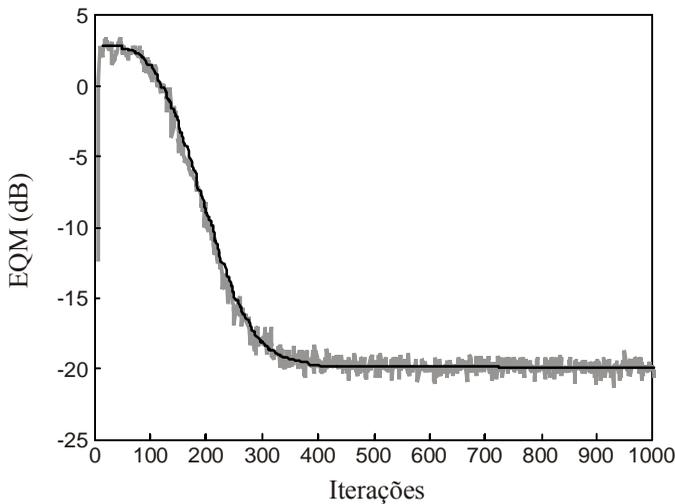
**Exemplo 3.3:** Neste exemplo, é considerado um sinal de entrada branco. As Figuras 3.10, 3.11 e 3.12 mostram, respectivamente, o comportamento do passo de adaptação, do vetor de coeficientes e do EQM. Uma comparação dos resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto é apresentada.



**Figura 3.10.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Aboulnasr [28] para sinal de entrada branco.



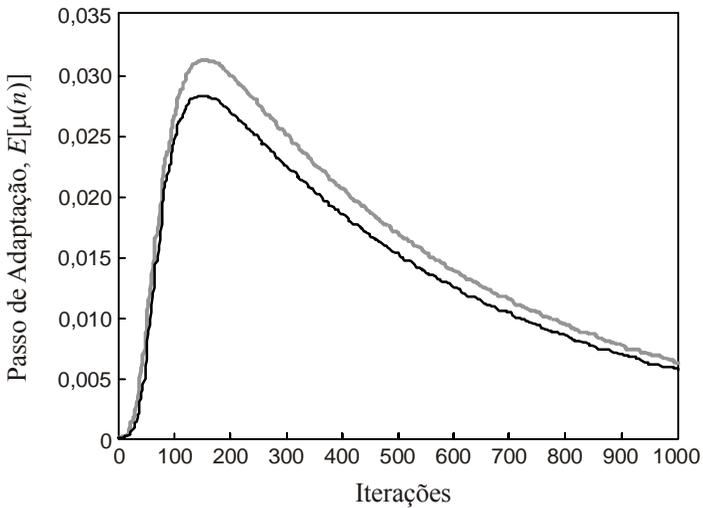
**Figura 3.11.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Aboulnasr [28] para sinal de entrada branco.



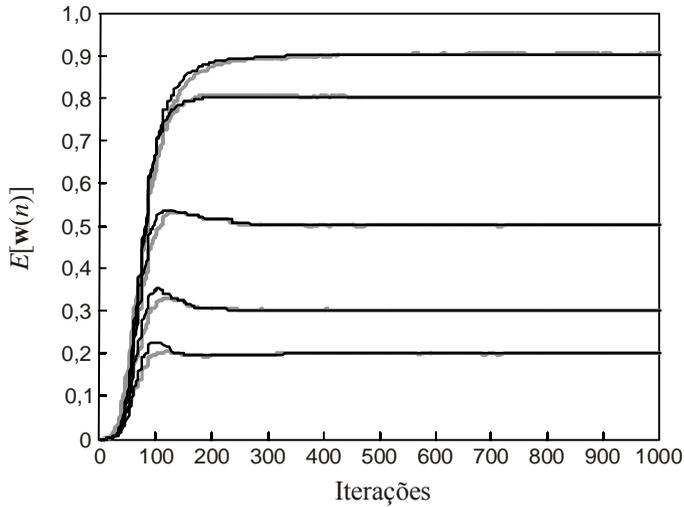
**Figura 3.12.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Aboulnasr [28] para sinal de entrada branco.

Observa-se das Figuras 3.10 a 3.12 um casamento muito bom entre os resultados obtidos a partir da simulação de MC e aqueles obtidos pelo modelo proposto do algoritmo de Aboulnasr [28], para sinal de entrada branco.

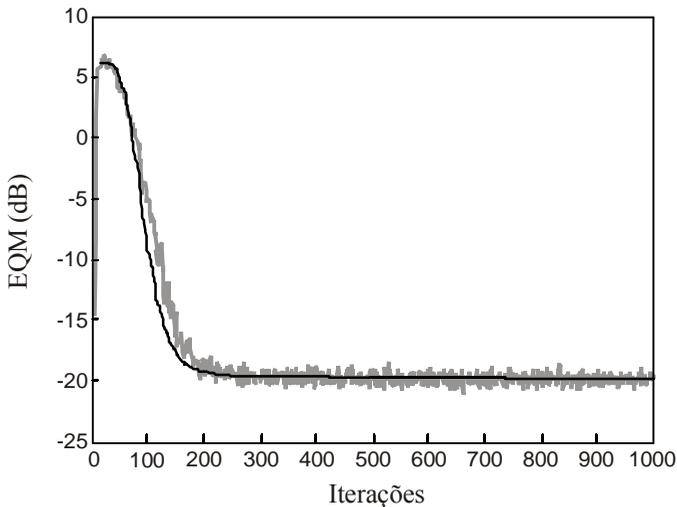
**Exemplo 3.4:** Neste exemplo, é considerado um sinal de entrada correlacionado. As Figuras 3.13, 3.14, e 3.15 mostram, respectivamente, o comportamento do passo de adaptação, vetor de coeficientes e EQM, possibilitando novamente uma comparação entre os resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto. A Figura 3.16 verifica por simulação a Aproximação (3.1), comparando as evoluções de  $E[\mu^2(n)]$  com  $E^2[\mu(n)]$ .



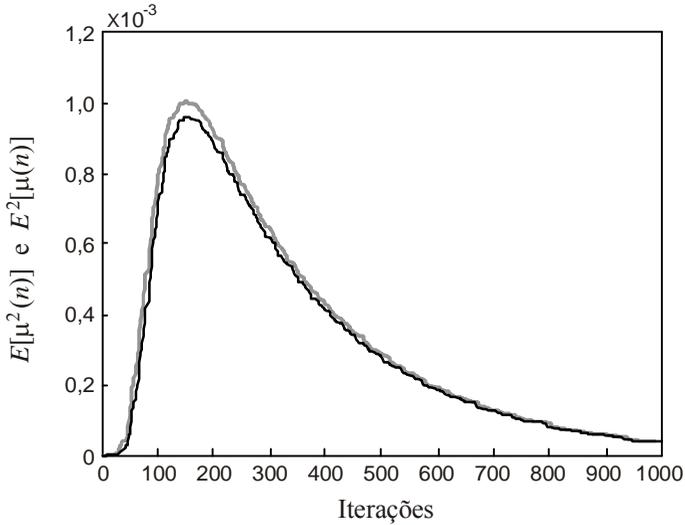
**Figura 3.13.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.



**Figura 3.14.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.



**Figura 3.15.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Aboulnasr [28] para sinal de entrada correlacionado.



**Figura 3.16.** Evolução de  $E[\mu^2(n)]$  (cinza) e  $E^2[\mu(n)]$  (preto) para o algoritmo de Aboulnasr [28].

Para o cenário considerado, se observa a partir das Figuras 3.13 a 3.15 um casamento muito bom entre os resultados de simulação e do modelo proposto para o algoritmo de Aboulnasr [28], considerando um sinal de entrada correlacionado. Algumas discrepâncias observadas entre modelo e simulação são devidas às aproximações utilizadas na determinação do modelo, especialmente no cálculo de  $p^2(n)$  (Apêndice A). A Figura 3.16 verifica a razoabilidade do uso da Aproximação (4.1).

### 3.6 Algoritmos Baseados no Valor Absoluto do Erro

Nesta categoria, o ajuste do passo variável de adaptação do algoritmo VSSLMS é realizado com base no valor absoluto do sinal de erro.

#### 3.6.1 Algoritmo de Kim [31]

Neste algoritmo, o passo de adaptação é ajustado de forma proporcional ao valor absoluto do erro de estimação. A equação de ajuste do passo de adaptação proposta leva em conta o valor absoluto do

sinal de erro normalizado em relação ao valor absoluto do sinal desejado. Assim,

$$\mu'(n) = \frac{|e(n-1)|}{|d(n-1)|} \mu_{\max} \quad (3.50)$$

com

$$\mu(n) = \begin{cases} \alpha\mu(n-1), & \mu(n) < \alpha\mu(n-1) \\ \frac{1}{\alpha}\mu(n-1), & \mu(n) > \frac{1}{\alpha}\mu(n-1) \\ \mu'(n), & \text{caso contrário} \end{cases} \quad (3.51)$$

onde  $0 < \alpha < 1$  é o fator de controle de convergência do passo de adaptação.

### 3.6.2 Algoritmo de Gollamudi [32]

Este algoritmo, baseado no valor absoluto do erro de adaptação, realiza o ajuste do passo variável de adaptação através de uma técnica de filtragem denominada *set membership* [32]. O objetivo dessa filtragem é estabelecer um valor limite para o valor absoluto do sinal de erro, abaixo do qual é assumido que o algoritmo convergiu. Para o algoritmo de Gollamudi, a expressão de atualização do passo variável é dada por

$$\mu(n) = \begin{cases} 1 - \frac{\gamma}{|e(n)|}, & \text{se } |e(n)| > \gamma \\ 0, & \text{caso contrário.} \end{cases} \quad (3.52)$$

O parâmetro  $\gamma$  controla o ajuste do passo de adaptação e representa o maior valor absoluto do sinal de erro aceitável para o regime permanente.

### 3.7 Algoritmos Baseados na Normalização do Vetor de Erros

Nesta categoria, a atualização do passo é realizada utilizando a norma do vetor de erro de estimação. O algoritmo de Ramadan [33], descrito a seguir, é o mais representativo dessa categoria.

### 3.7.1 Algoritmo de Ramadan [33]

Neste algoritmo, a estratégia de atualização do passo depende de uma normalização de vetores de erros. Tais normas são determinadas a partir de

$$\|\mathbf{e}(n)\|^2 = \sum_{k=0}^{n-1} |e(n-k)|^2 \quad (3.53)$$

e

$$\|\mathbf{e}_L(n)\|^2 = \sum_{k=0}^{L-1} |e(n-k)|^2. \quad (3.54)$$

A expressão de ajuste do passo desse algoritmo é então dada por

$$\mu(n) = \frac{\mu \|\mathbf{e}_L(n)\|^2}{[\alpha \|\mathbf{e}(n)\|^2 + (1-\alpha) \|\mathbf{x}(n)\|^2]} \quad (3.55)$$

onde  $\mu$  e  $\alpha$  são parâmetros constantes positivos que controlam o processo de adaptação. Uma constante de pequeno valor  $\varepsilon$  (parâmetro de regularização) pode ser adicionada ao denominador de (3.55), prevenindo a instabilidade do algoritmo para valores muito pequenos desse denominador.

## 3.8 Algoritmos Baseados nos Valores Absolutos do Vetor de Coeficientes

Nesta categoria, estão aqueles algoritmos cujo ajuste do valor do passo individual de adaptação é realizado de forma proporcional ao valor absoluto de cada componente do vetor de coeficientes  $\mathbf{w}(n)$ . A idéia central aqui é que, quanto maiores forem os valores das amostras da resposta ao impulso do sistema a ser identificado, maior deverá ser o valor do correspondente passo de adaptação.

### 3.8.1 Algoritmo de Rohani [34]

Este é o algoritmo mais simples desta categoria e utiliza passos individuais de adaptação. A equação de atualização dos passos individuais  $\mu_i(n)$  é dada por

$$\mu_i(n) = \beta |w_i(n)| + \mu \quad (3.56)$$

onde  $\beta$  e  $\mu$  são constantes positivas. O parâmetro  $\mu$  é usado como valor inicial para o processo de ajuste uma vez que todos os coeficientes do vetor  $\mathbf{w}(n)$  são geralmente inicializados com valor zero.

### 3.8.2 Algoritmo de Benesty (IPNLMS) [36]

O algoritmo LMS normalizado proporcional (PNLMS) é apresentado em [35], com o objetivo de melhorar o desempenho do algoritmo NLMS quando a resposta ao impulso da planta é do tipo esparsa. No processo de adaptação do PNLMS, o ajuste do passo de adaptação é proporcional ao valor do coeficiente. Entretanto, quando a resposta ao impulso do sistema desconhecido é do tipo dispersiva, este algoritmo apresenta um desempenho inferior ao NLMS [36]. Para contornar tal deficiência, em [36] é proposta uma modificação do algoritmo PNLMS padrão, originando o algoritmo PNLMS melhorado (IPNLMS). No algoritmo IPNLMS o ajuste dos passos individuais de adaptação considera um fator de ponderação  $k_i(n)$  dado por

$$k_i(n) = (1 - \alpha) \frac{\|\mathbf{w}(n)\|_1}{N} + (1 + \alpha) |w_i(n)| \quad (3.57)$$

Com  $\|\mathbf{w}(n)\|_1$  definido como

$$\|\mathbf{w}(n)\|_1 = \sum_{i=0}^{N-1} |w_i(n)|. \quad (3.58)$$

O ajuste do passo individual de adaptação  $\mu'_i(n)$  é então realizado com base nas seguintes expressões:

$$\mu'_i(n) = \frac{k_i(n)}{\|\mathbf{k}(n)\|_1} \quad (3.59)$$

onde  $\|\mathbf{k}(n)\|_1$  é definido como

$$\|\mathbf{k}(n)\|_1 = \sum_{i=0}^{N-1} |k_i(n)|. \quad (3.60)$$

A matriz de passos individuais  $\mathbf{D}(n)$  é então obtida de:

$$\mathbf{D}(n) = \frac{\mu \mathbf{D}'(n-1)}{\mathbf{x}^T(n) \mathbf{D}'(n-1) \mathbf{x}(n) + \varepsilon} \quad (3.61)$$

onde  $\mathbf{D}'(n)$  é uma matriz diagonal obtida dos passos individuais dada por

$$\mathbf{D}'(n) = \text{diag}\{\mu'_0(n), \dots, \mu'_{N-1}(n)\}. \quad (3.62)$$

Observa-se que quando  $\alpha = -1$  o algoritmo IPNLMS é idêntico ao algoritmo NLMS e para  $\alpha$  próximo de 1 o algoritmo se comporta como o PNLMS.

### 3.9 Algoritmos Baseados na Teoria do Passo Ótimo

Os algoritmos desta categoria são baseados no valor teórico do passo ótimo, o qual é obtido pela minimização do EQM em excesso  $J_{\text{ex}}(n)$ , com respeito ao passo variável de adaptação. O valor teórico do passo ótimo de adaptação é aquele que satisfaz

$$\frac{\partial J_{\text{ex}}(n)}{\partial \mu(n)} = 0. \quad (3.63)$$

O objetivo dos algoritmos baseados no passo ótimo é encontrar o valor do passo de adaptação que, à cada iteração, aproxime (3.63).

#### 3.9.1 Algoritmo de Yan-Bin [38]

Em [38], é proposto um algoritmo VSSLMS, baseado na teoria do passo ótimo. Nele, a expressão de ajuste do passo de adaptação é obtida de forma a satisfazer (3.63), consideradas algumas aproximações. Para este algoritmo, em cada iteração,  $\mu(n)$  é atualizado por

$$\mu(n) = \frac{a(n)}{b(n)} \quad (3.64)$$

onde  $a(n)$  e  $b(n)$  são obtidos recursivamente por

$$a(n) = (1 - \alpha)a(n-1) + \alpha e^2(n-1) \quad (3.65)$$

$$b(n) = (1 - \beta)b(n-1) + \beta[e^2(n)\mathbf{x}^T(n)\mathbf{x}(n)]. \quad (3.66)$$

As expressões (3.65) e (3.66) permitem a obtenção do passo de adaptação em (3.64) através de uma fórmula iterativa que aproxima a solução de (3.63) [38].

### 3.10 Algoritmos Baseados na Otimização com Restrição do Ruído de Medição

Os algoritmos desta categoria buscam ajustar o passo do algoritmo LMS segundo algum critério de otimização com restrição do ruído de medição (NCLMS) [37].

#### 3.10.1 Algoritmo de Wei [37]

Em algumas aplicações práticas pode-se ter o conhecimento prévio da potência do ruído, ou ainda estimá-la. Dessa forma, pode-se utilizar tal conhecimento com o objetivo de melhorar o desempenho de algoritmos adaptativos, sem aumentar significativamente a sua complexidade computacional ou diminuir sua robustez.

Seja o erro quadrático médio (EQM) dado por

$$E[e^2(n)] = E\{[d(n) - \mathbf{w}^T \mathbf{x}(n)]^2\}. \quad (3.67)$$

Considere a minimização de (3.67) em relação ao vetor  $\mathbf{w}$ . Assumindo o conhecimento prévio de  $\sigma_\eta^2$ , pode-se transformar (3.67) em um problema de otimização com restrições pela minimização de  $E[e^2(n)]$  sujeito a  $E[e^2(n)] = \sigma_\eta^2$  [37]. Dessa forma, a nova função-custo é agora dada por

$$\varepsilon_1(\mathbf{w}, \lambda) = E[e^2(n)] + \lambda[E[e^2(n)] - \sigma_\eta^2]. \quad (3.68)$$

Os valores críticos de (3.68) são  $\mathbf{w} = \mathbf{w}_0$  com  $\lambda$  podendo assumir qualquer valor. Para tentar se obter uma solução única, um termo com uma penalidade adicional  $\gamma\lambda^2$  (com  $\gamma > 0$ ) é subtraído de (3.68), resultando na seguinte expressão:

$$\varepsilon_2(\mathbf{w}, \lambda) = E[e^2(n)] + \lambda(E[e^2(n)] - \sigma_\eta^2) - \gamma\lambda^2. \quad (3.69)$$

Agora, os únicos valores críticos de (3.69) são  $\mathbf{w} = \mathbf{w}_0$  e  $\lambda = 0$ . Na minimização de  $\varepsilon_2(\mathbf{w}, \lambda)$ ,  $(\mathbf{w}, \lambda) = (\mathbf{w}_0, 0)$  é um ponto de sela de  $\varepsilon_2(\mathbf{w}, \lambda)$ . Usando o algoritmo de Robbins-Monro [48], obtêm-se as seguintes relações recursivas:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \alpha_f \frac{\partial \varepsilon_2}{\partial \mathbf{w}} \quad (3.70)$$

$$\lambda(n+1) = \lambda(n) + \beta \frac{\partial \varepsilon_2}{\partial \lambda} \quad (3.71)$$

onde  $\alpha_f$  e  $\beta$  são os passos de adaptação do algoritmo NCLMS.

As derivadas parciais em (3.70) e (3.71) são determinadas como segue:

$$\frac{\partial \varepsilon_2}{\partial \mathbf{w}} = -2[1 + \gamma\lambda(n)]e(n)\mathbf{x}(n) \quad (3.72)$$

e

$$\frac{\partial \varepsilon_2}{\partial \lambda} = \gamma[e(n)^2 - \sigma_\eta^2] - 2\gamma\lambda(n). \quad (3.73)$$

Então, substituindo (3.72) e (3.73) em (3.70) e (3.71), obtêm-se

$$\mathbf{w}(n) = \mathbf{w}(n) + 2\alpha_f[1 + \gamma\lambda(n)]e(n)\mathbf{x}(n) \quad (3.74)$$

e

$$\lambda(n+1) = \lambda(n) + \beta\gamma[e(n)^2 - \sigma_\eta^2] - 2\beta\gamma\lambda(n). \quad (3.75)$$

Substituindo agora  $\alpha_f$  por  $\frac{\alpha_f}{2}$  e  $\beta\gamma$  por  $\frac{\beta}{2}$ , obtêm-se as expressões para o algoritmo NCLMS. Assim,

$$\mu(n) = \alpha_f[1 + \gamma\lambda(n)] \quad (3.76)$$

e

$$\lambda(n+1) = \lambda(n) + \beta\left\{\frac{1}{2}[e^2(n) - \sigma_\eta^2] - \lambda(n)\right\}. \quad (3.77)$$

Note que  $\mu(n)$  é composto por dois termos, um constante  $\alpha_f$  e outro variável  $\lambda(n)$ . Os parâmetros  $0 < \beta < 1$  e  $0 < \gamma < 1$  são constantes de controle do algoritmo.

#### A. Modelo estocástico

Apresenta-se a seguir uma análise estatística do comportamento do passo variável de adaptação para o algoritmo de Wei. Tal análise foi desenvolvida neste trabalho de pesquisa.

Considerando  $\lambda(0) = 0$ , (3.77) pode ser reescrita como

$$\lambda(n) = \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} \frac{\beta}{2} [e^2(k) - \sigma_{\eta}^2]. \quad (3.78)$$

Então, substituindo (3.78) em (3.76), obtém-se

$$\mu(n) = \alpha_f + \frac{\alpha_f \beta \gamma}{2} \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} [e^2(k) - \sigma_{\eta}^2]. \quad (3.79)$$

Fazendo  $\delta = \frac{1}{2} \alpha_f \beta \gamma$ , tem-se que

$$\mu(n) = \alpha_f + \delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} [e^2(k) - \sigma_{\eta}^2]. \quad (3.80)$$

Definindo agora  $\alpha(n)$  como a parte variável do passo de adaptação

$$\alpha(n) = \delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} [e^2(k) - \sigma_{\eta}^2] \quad (3.81)$$

a expressão de atualização do vetor de coeficientes é escrita como

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(n)e(n)\mathbf{x}(n) = \mathbf{w}(n) + \alpha_f e(n)\mathbf{x}(n) + \alpha(n)e(n)\mathbf{x}(n). \quad (3.82)$$

Tomando o valor esperado de ambos os lados de (3.82), tem-se

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \alpha_f E[e(n)\mathbf{x}(n)] + E[\alpha(n)e(n)\mathbf{x}(n)]. \quad (3.83)$$

Arranjando os termos de (3.83), obtém-se

$$E[\mathbf{w}(n+1)] = \underbrace{E[\mathbf{w}(n)] + \alpha_f \{\mathbf{p} - \mathbf{RE}[\mathbf{w}(n)]\}}_{\text{LMS convencional}} + E[\alpha(n)e(n)\mathbf{x}(n)] \quad (3.84)$$

ou ainda, utilizando a hipótese (ii), tem-se

$$E[\mathbf{w}(n+1)] = \underbrace{E[\mathbf{w}(n)] + \alpha_f \{\mathbf{p} - \mathbf{RE}[\mathbf{w}(n)]\}}_{\text{LMS convencional}} + E[\alpha(n)]E[\mathbf{p} - \mathbf{RE}[\mathbf{w}(n)]]. \quad (3.85)$$

Nota-se que (3.85) é idêntica à expressão que descreve o valor médio do vetor de coeficientes para o algoritmo LMS convencional, exceto pelo último termo à direita, onde  $E[\alpha(n)]$  pode ser escrito como

$$\begin{aligned}
 & E[\alpha(n)] \\
 &= E\left[\delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} (e^2(k) - \sigma_\eta^2)\right] = \delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} [J(k) - J_{\min}] \quad (3.86)
 \end{aligned}$$

ou

$$\begin{aligned}
 & E[\alpha(n)] \\
 &= E\left[\delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} (e^2(k) - \sigma_\eta^2)\right] = \delta \sum_{k=0}^{n-1} (1-\beta)^{n-1-k} \text{tr}[\mathbf{R}\mathbf{K}(k)]. \quad (3.87)
 \end{aligned}$$

Assim, o valor esperado do passo variável de adaptação  $\mu(n) = \alpha_f + \alpha(n)$  pode ser obtido. Então,

$$E[\mu(n)] = \alpha_f + E[\alpha(n)]. \quad (3.88)$$

Da mesma forma, o valor esperado do passo de adaptação ao quadrado é

$$E[\mu^2(n)] = \alpha_f^2 + 2\alpha_f E[\alpha(n)] + E[\alpha^2(n)] \quad (3.89)$$

com

$$E[\alpha^2(n)] = \delta^2 \sum_{k=0}^{n-1} (1-\beta)^{2(n-1-k)} [J^2(k) - 2J(k)\sigma_\eta^2 + \sigma_\eta^4]. \quad (3.90)$$

### B. Análise em regime permanente

Realiza-se aqui uma análise do valor em regime permanente para o vetor  $\mathbf{w}(n)$ , passo variável  $\mu(n)$  e  $J(n)$  no algoritmo de Wei [37].

#### Vetor de coeficientes:

Considerando  $\mathbf{w}(n)$  convergente e utilizando (3.85) quando  $n \rightarrow \infty$ , tem-se

$$\mathbf{w}(\infty) = \mathbf{w}(\infty) + \alpha_f [\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)] + \alpha(\infty) [\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)]. \quad (3.91)$$

Visto que  $\alpha_f$  e  $\alpha(\infty)$  são positivos, a única solução de (3.91) é obtida para

$$[\mathbf{p} - \mathbf{R}\mathbf{w}(\infty)] = 0 \quad (3.92)$$

resultando em

$$\mathbf{w}(\infty) = \mathbf{R}^{-1}\mathbf{p}. \quad (3.93)$$

A partir de (3.93), constata-se que o valor em regime permanente do algoritmo NCLMS é o mesmo que o do algoritmo LMS convencional, convergindo para a solução de Wiener.

Passo de adaptação:

Calculando o valor de  $\lambda(n)$  em (3.77) para  $n \rightarrow \infty$ , tem-se

$$\lambda(\infty) = \frac{J_{\text{ex}}(\infty)}{2}. \quad (3.94)$$

Determinando agora o valor de  $\mu(n)$  em (3.76) para  $n \rightarrow \infty$  e utilizando (3.94), resulta em

$$\mu(\infty) = \alpha_f + \frac{1}{2} \alpha_f \gamma J_{\text{ex}}(\infty). \quad (3.95)$$

Erro quadrático médio em excesso:

Como, em regime permanente, o passo de adaptação converge para um valor constante, utiliza-se aqui, novamente, (2.41) para a determinação do valor do EQM em excesso em regime permanente. Assim,

$$J_{\text{ex}}(\infty) = \frac{\mu(\infty) J_{\min}}{2} \text{tr}(\mathbf{R}) = \frac{[2\alpha_f + \alpha_f \gamma J_{\text{ex}}(\infty)]}{4} \sigma_\eta^2 \text{tr}(\mathbf{R}) \quad (3.96)$$

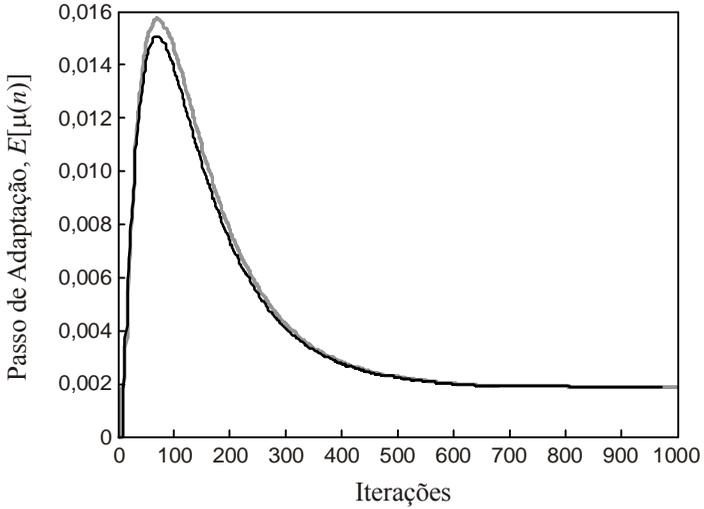
resultando finalmente em

$$J_{\text{ex}}(\infty) = \frac{2\alpha_f \sigma_\eta^2 \text{tr}(\mathbf{R})}{4 - \alpha_f \gamma \sigma_\eta^2 \text{tr}(\mathbf{R})}. \quad (3.97)$$

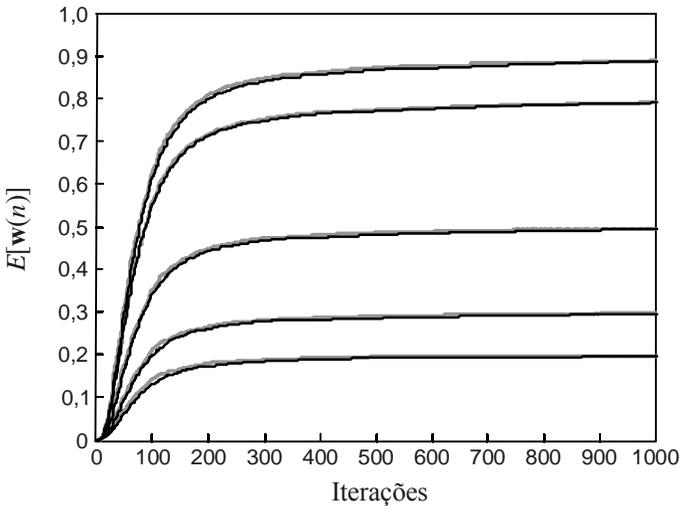
*C. Resultados de simulação*

Os resultados de simulação para o modelo estocástico do algoritmo de Wei são aqui mostrados em dois exemplos. O cenário utilizado é o mesmo descrito nos Exemplos 3.1 e 3.2. O primeiro exemplo, considera um sinal de entrada branco e, o segundo, um sinal de entrada correlacionado.

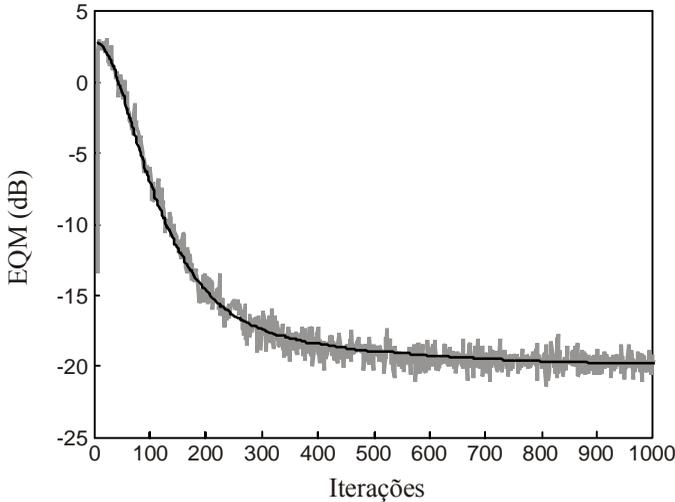
Exemplo 3.5: Neste exemplo, é utilizado um sinal de entrada branco. As Figuras 3.17, 3.18 e 3.19 mostram, respectivamente, a evolução do passo de adaptação, o vetor de coeficientes e o EQM, comparando os resultados através de simulação de MC (200 realizações independentes) e pelo modelo proposto.



**Figura 3.17.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do Algoritmo de Wei [37] para sinal de entrada branco.



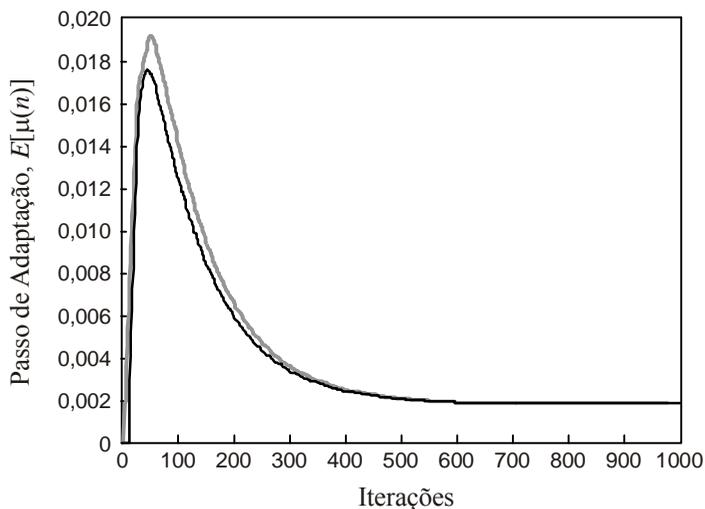
**Figura 3.18.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Wei [37] para sinal de entrada branco.



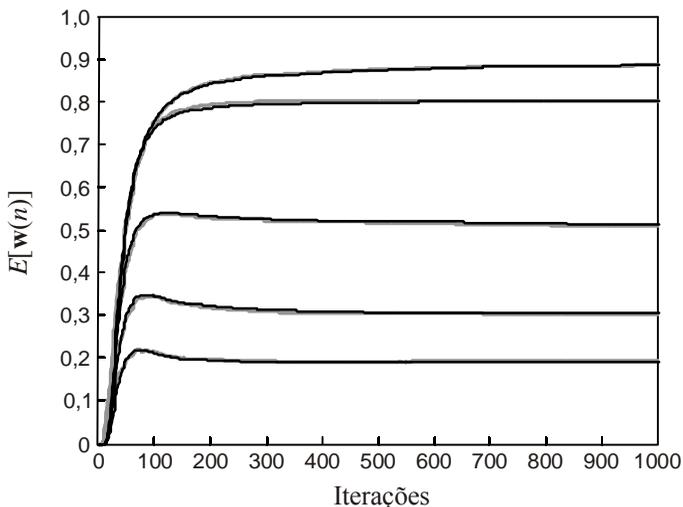
**Figura 3.19.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Wei [37] para sinal de entrada branco.

Observa-se nas Figuras 3.17, 3.18 e 3.19 um casamento muito bom entre os resultados obtidos através de simulação de MC e pelo modelo proposto, para o caso de sinal de entrada branco, no cenário de simulação considerado.

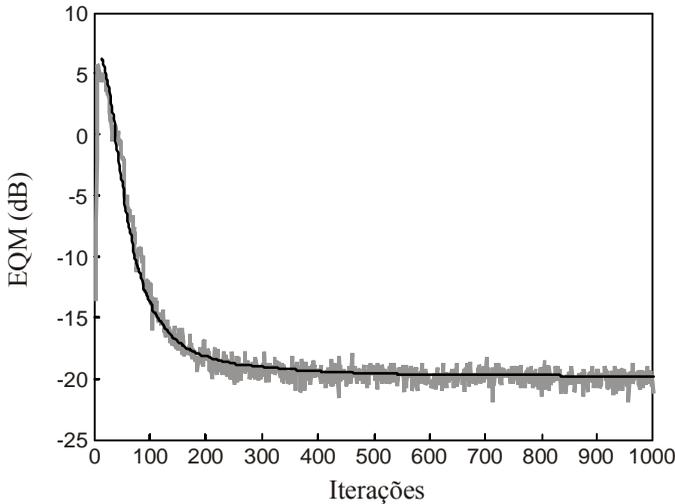
Exemplo 3.6: Este exemplo utiliza sinal de entrada correlacionado. As Figuras 3.20, 3.21 e 3.22 mostram, respectivamente, a evolução do passo de adaptação, o vetor de coeficientes e o EQM, comparando os resultados através de simulação de MC (200 realizações independentes) e pelo modelo proposto.



**Figura 3.20.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do Algoritmo de Wei [37] para sinal de entrada correlacionado.



**Figura 3.21.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $\mathbf{w}(n)$  do algoritmo de Wei [37] para sinal de entrada correlacionado.



**Figura 3.22.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Wei [37] para sinal de entrada correlacionado.

Observa-se das Figuras 3.20, 3.21 e 3.22 um casamento muito bom entre os resultados de simulação e aqueles obtidos através do modelo desenvolvido, para um sinal de entrada correlacionado. As pequenas discrepâncias existentes na fase transiente observados nas Figuras 3.20 e 3.21 são devidas às aproximações consideradas no desenvolvimento do modelo.

### 3.11 Conclusões

Este capítulo apresentou uma breve discussão de diversos algoritmos VSSLMS encontrados na literatura técnica levando em conta suas principais especificidades. Foi proposta uma classificação desses algoritmos, segundo o princípio de ajuste utilizado para o passo de adaptação. No total, foram consideradas oito categorias reunindo 18 algoritmos VSSLMS. Cada algoritmo foi descrito através de uma nomenclatura unificada, permitindo uma compreensão mais clara e facilitando uma análise comparativa das diversas estratégias utilizadas.

Também foram apresentados modelos estocásticos para predição do comportamento médio do vetor de coeficientes e da curva de aprendizagem para três dos algoritmos classificados. O desenvolvimento de modelos estocásticos para algoritmos adaptativos envolve um trabalho matemático intenso e a elaboração de algumas hipóteses simplificativas. Neste capítulo, foram abordadas tais hipóteses, além de outros aspectos comuns para a modelagem de algoritmos VSSLMS.

Pode-se concluir que existem grandes diferenças entre os diversos algoritmos estudados. A escolha de um dado algoritmo VSSLMS para uma determinada aplicação não é uma tarefa trivial. Portanto, é de fundamental importância o desenvolvimento de modelos estocásticos que permitam uma análise mais aprofundada do comportamento de cada algoritmo considerando os diversos cenários de aplicação. Resultados de simulação apresentados nesta seção comprovaram a precisão dos modelos estocásticos desenvolvidos.



## Novos Algoritmos VSSLMS

### 4.1 Introdução

Com o propósito de se obter taxas elevadas de convergência com um desajuste reduzido, várias estratégias para ajustar o valor do passo de adaptação em algoritmos VSSLMS vêm sendo propostas na literatura. O objetivo deste capítulo é apresentar novas propostas de algoritmos VSSLMS.

Os algoritmos baseados no gradiente ajustam o passo de adaptação a partir do valor numérico do gradiente do erro quadrático de estimação. Em [17] é apresentado um dos primeiros algoritmos utilizando tal abordagem, denominado aqui algoritmo de Shan. Tal algoritmo se mostra bastante simples e eficaz apresentando, porém, um problema elementar de convergência que ocorre quando o gradiente médio resultar em um valor negativo. Assim, com o objetivo de contornar o problema verificado, é aqui proposta uma modificação no algoritmo de Shan e apresentado um modelo estocástico do algoritmo modificado.

Neste capítulo, também são propostos dois novos algoritmos baseados na autocorrelação do erro. O primeiro deles, aqui denominado algoritmo AP1, tem seu ajuste baseado na função de autocorrelação do erro considerando  $N$  lags. Essa característica melhora o desempenho do algoritmo quando comparado ao algoritmo de Abounasr [28], aumentando a velocidade de convergência na presença de ruído de medição. Embora o algoritmo proposto AP1 represente uma evolução em relação ao algoritmo de Abounasr [28], ele ainda apresenta dois principais problemas inerentes aos algoritmos VSSLMS. São eles: a baixa imunidade a variações no nível de ruído de medição e a necessidade de ajuste de parâmetros, o que geralmente é feito de forma experimental. Com o objetivo de obter um algoritmo imune a variações no ruído de medição e que não necessite de ajuste de parâmetros, é também proposto neste capítulo um segundo algoritmo, baseado na autocorrelação do erro, aqui denominado algoritmo AP2.

Neste capítulo, são ainda apresentados os modelos estocásticos dos três algoritmos avaliados. Resultados de simulação comprovam o desempenho dos três algoritmos e verificam a precisão dos modelos desenvolvidos.

## 4.2 Algoritmo de Shan Modificado [42]

Nesta seção, é apresentada uma solução eficaz ao problema de convergência do algoritmo VSSLMS proposto em [17], assim como um modelo estocástico para o comportamento médio do vetor de coeficientes e para a curva de aprendizagem do algoritmo adaptativo modificado. A precisão do modelo analítico proposto é verificada através de resultados de simulação.

O algoritmo de Shan apresentado em [17] utiliza uma estimativa do gradiente médio do sinal de erro ao quadrado em função do vetor de coeficientes dada pela correlação média entre o sinal de erro de adaptação e o sinal de entrada, obtida por

$$g(n) = \lambda g(n-1) + (1-\lambda)e(n)\bar{x}(n) \quad (4.1)$$

com

$$\bar{x}(n) = \frac{1}{N} \sum_{k=n-N+1}^n x(k) \quad (4.2)$$

onde  $\lambda$  controla a estimação de  $g(n)$ . O passo comum de adaptação é então ajustado de forma proporcional ao valor do gradiente estimado. Portanto,

$$\mu(n) = \alpha g(n) \quad (4.3)$$

onde o fator  $\alpha$  é uma constante positiva menor do que a unidade.

Observa-se de (4.3) que um valor negativo do gradiente médio resulta em um passo de adaptação negativo, sendo que este está limitado a zero pela desigualdade dada em (2.57), levando o algoritmo LMS a ter um comportamento inadequado. Neste trabalho, é apresentada uma modificação no algoritmo de Shan, utilizando o valor absoluto do gradiente [42]. Dessa forma, (4.3) é redefinida como

$$\mu(n) = \alpha |g(n)|. \quad (4.4)$$

Esta alteração supera os problemas de convergência do algoritmo de Shan, anteriormente mencionados, mantendo sua simplicidade, originando apenas um pequeno aumento de complexidade computacional em comparação ao LMS convencional.

## 4.2.1 Modelo Estocástico para o Algoritmo de Shan Modificado [42]

Neste trabalho, é também proposta uma análise estatística do algoritmo de Shan modificado, a qual é apresentada a seguir.

Seja a estimativa do vetor gradiente  $\mathbf{g}(n)$  dada por

$$\mathbf{g}(n) = \lambda \mathbf{g}(n-1) + (1-\lambda)e(n)\mathbf{x}(n). \quad (4.5)$$

O gradiente médio (ou correlação média entre o erro de adaptação e o sinal de entrada), obtida em (4.1), pode ser reescrita como

$$g(n) = \frac{1}{N} \sum_{i=0}^{N-1} g_i(n). \quad (4.6)$$

Em (4.6),  $g_i(n)$  denota o  $i$ -ésimo componente do vetor  $\mathbf{g}(n)$ . Calculando o valor esperado em ambos os lados de (4.5), tem-se

$$E[\mathbf{g}(n)] = \lambda E[\mathbf{g}(n-1)] + (1-\lambda)\{\mathbf{p} - \mathbf{R}E[\mathbf{w}(n)]\}. \quad (4.7)$$

Obtém-se então o valor esperado de (4.6), dado por

$$E[g(n)] = \frac{1}{N} \sum_{i=0}^{N-1} E[g_i(n)] \quad (4.8)$$

e, dessa forma, o valor esperado do passo de adaptação é obtido utilizando (4.8) e aproximando o valor esperado de (4.4) na forma

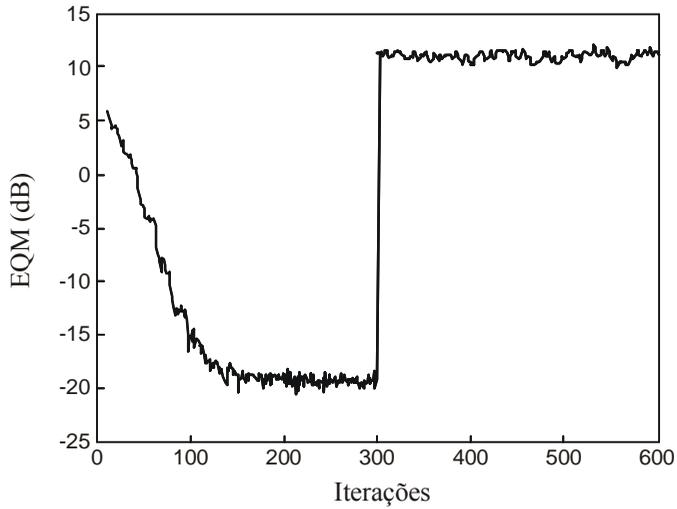
$$E[\mu(n)] \cong \alpha |E[g(n)]|. \quad (4.9)$$

Nesse modelo, utilizou-se a hipótese dada em (3.1) para a determinação de  $E[\mu^2(n)]$ .

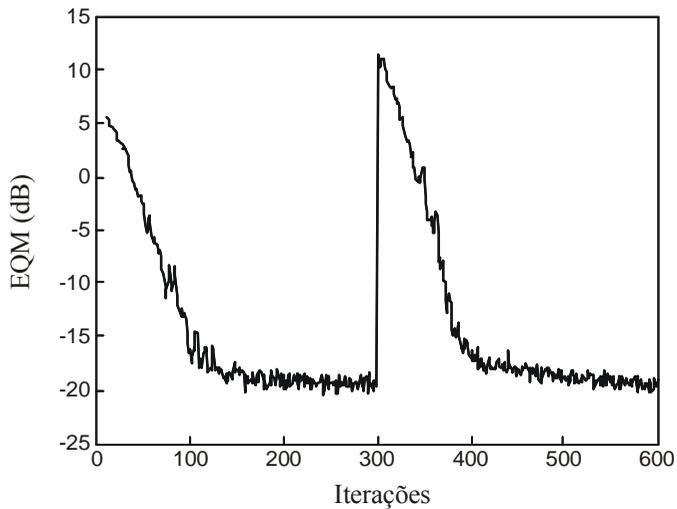
#### 4.2.2 Resultados de Simulação

Os resultados de simulação para o algoritmo de Shan modificado são apresentados considerando três exemplos. No Exemplo 4.1, o comportamento do algoritmo com a modificação incluída é avaliado. Nos Exemplos 4.2 e 4.3, a precisão do modelo estocástico desenvolvido é verificada para sinais de entrada branco e correlacionado, respectivamente.

Exemplo 4.1: Neste exemplo é considerado um problema de identificação de sistema. A planta utilizada é obtida a partir de uma janela de Hanning com 9 coeficientes dada por  $\mathbf{w}_o = [0,000 \ 0,1170 \ 0,4132 \ 0,7500 \ 0,9698 \ 0,7500 \ 0,4132 \ 0,1170 \ 0,000]^T$ . O sinal de entrada utilizado é correlacionado obtido de um processo AR(1), dado por  $x(n) = 0,4x(n-1) + u(n)$ , onde  $u(n)$  é um ruído branco Gaussiano com variância tal que  $\sigma_x^2 = 1$ . A dispersão dos autovalores da matriz de autocorrelação do sinal de entrada é  $\chi = 5$ . Também, é introduzido um ruído aditivo de medição com variância  $\sigma_\eta^2 = 0,01$  (SNR = 20 dB). Os resultados de simulação são obtidos através do método de Monte Carlo (MC), considerando uma média de 200 realizações independentes. Na iteração  $n = 300$ , é provocada uma alteração na planta, mudando o sinal de todos os seus coeficientes. Dessa forma, é verificado o comportamento dos algoritmos [Shan original [17] e Shan modificado (proposto)] frente uma perturbação. Da Figura 4.1(a), observa-se que o algoritmo original [17] não converge após a referida alteração da planta, visto que o gradiente médio se torna negativo. Na Figura 4.1(b), é mostrado o comportamento do algoritmo proposto que inclui a modificação de (4.3) para (4.4). O algoritmo proposto agora converge sem quaisquer problemas.



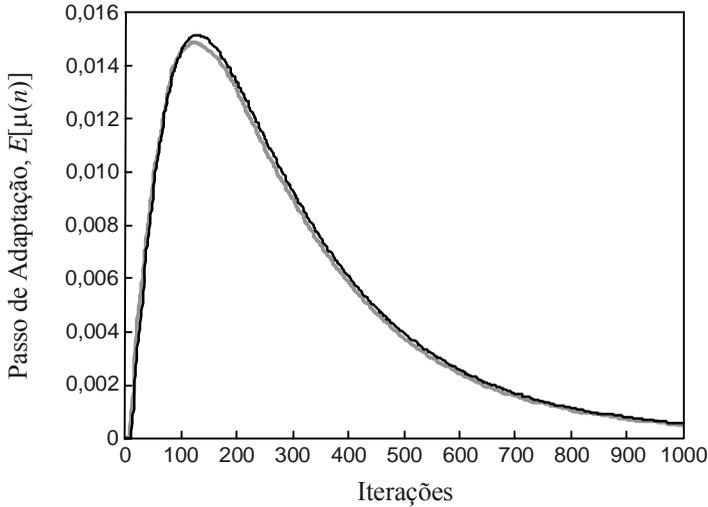
(a)



(b)

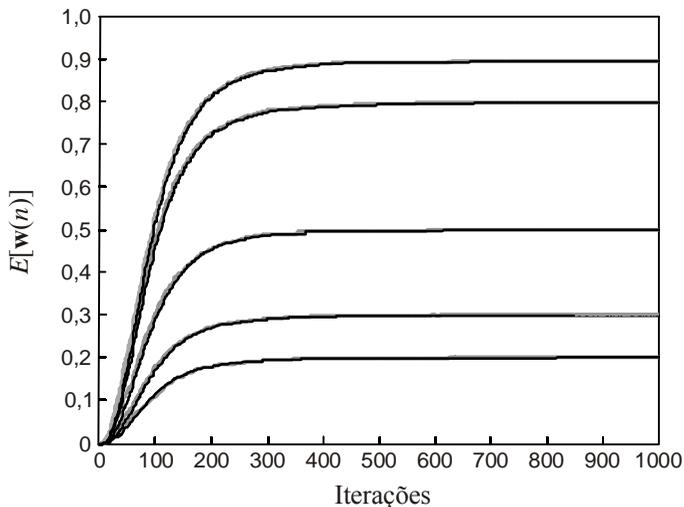
**Figura 4.1.** Evolução do EQM. (a) Algoritmo original [17]. (b) Algoritmo modificado (proposto).

Exemplo 4.2: Neste exemplo, a precisão do modelo é verificada considerando um sinal de entrada branco e o mesmo cenário descrito no Exemplo 3.1. As evoluções do passo de adaptação, do vetor de coeficientes e do EQM podem ser observadas, respectivamente, nas Figuras 4.2, 4.3 e 4.4. Uma comparação entre os resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto é apresentada.

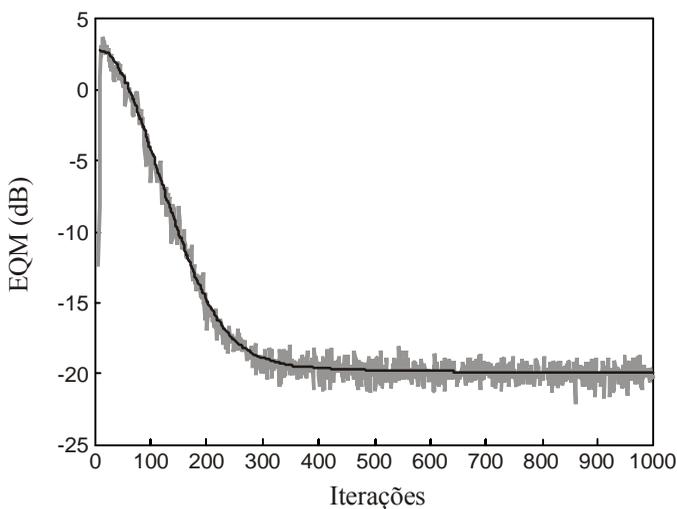


**Figura 4.2.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Shan modificado [42] para sinal de entrada branco.

Observa-se das Figuras 4.2, 4.3 e 4.4 um casamento muito bom entre os resultados de simulação e do modelo para o algoritmo de Shan modificado [42], tanto para a evolução do passo de adaptação quanto para o comportamento do vetor de coeficientes  $\mathbf{w}(n)$ , como também para o EQM, considerando sinal de entrada branco.

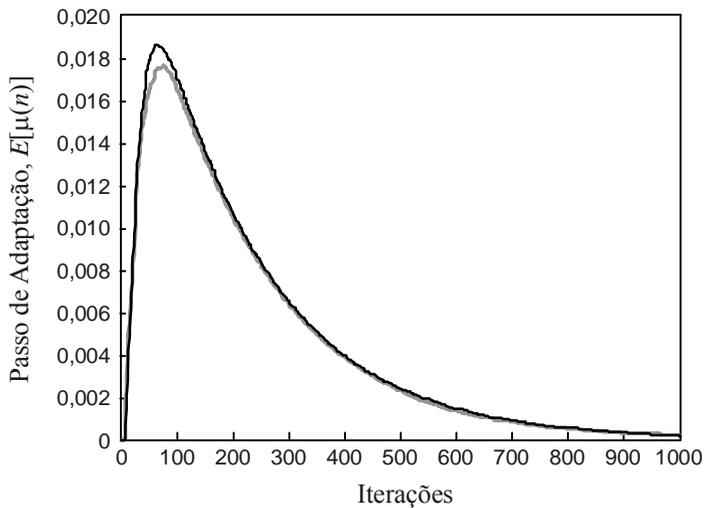


**Figura 4.3.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Shan modificado [42] para sinal de entrada branco.



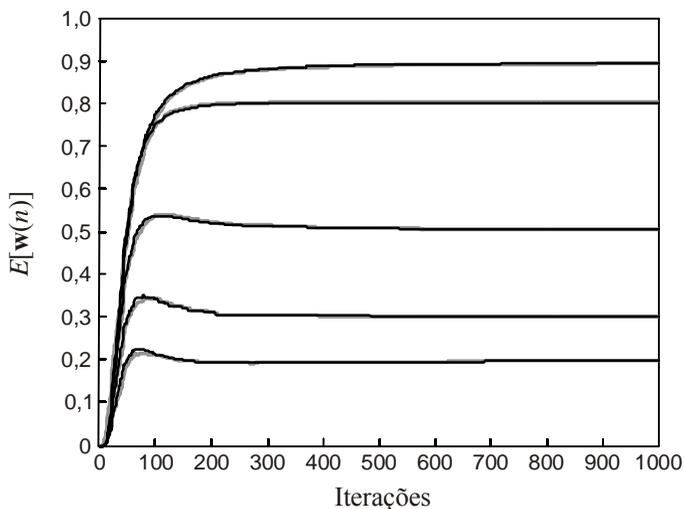
**Figura 4.4.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Shan modificado [42] para sinal de entrada branco.

**Exemplo 4.3:** Neste exemplo, é considerado um sinal de entrada correlacionado, no mesmo cenário descrito no Exemplo 3.2. As Figuras 4.5, 4.6 e 4.7 ilustram, respectivamente, o comportamento do passo de adaptação, do vetor de coeficientes e do EQM, para o algoritmo de Shan modificado [42], comparando os resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto. Na Figura 4.8, verifica-se através de simulação a aproximação (3.1), comparando os resultados de  $E[\mu^2(n)]$  e  $E^2[\mu(n)]$ .

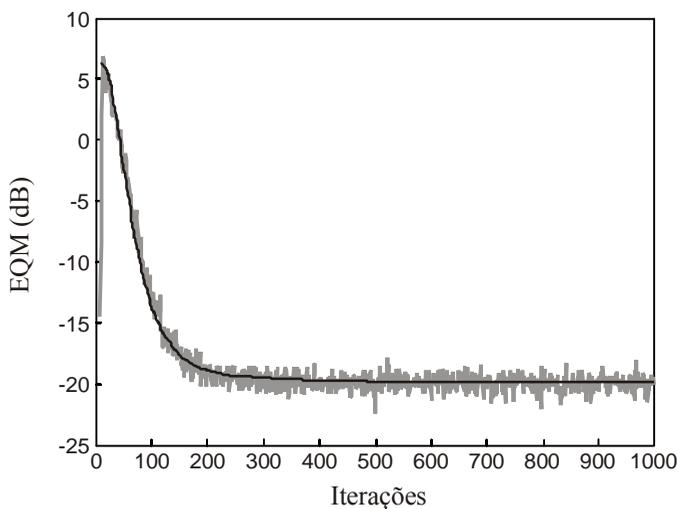


**Figura 4.5.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.

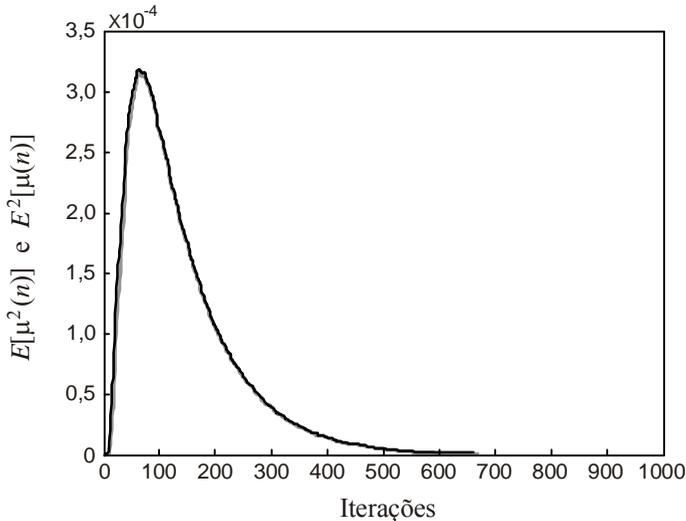
Novamente, se observa a partir das Figuras 4.5, 4.6 e 4.7 um casamento muito bom entre os resultados de simulação e do modelo proposto para o algoritmo de Shan modificado [42], para um sinal de entrada correlacionado. Observa-se também, que as aproximações utilizadas no modelo acarretam pequenos desajustes na fase transiente das evoluções mostradas nas Figuras 4.5 e 4.6. A Figura 4.8 apresenta a validação por simulação da Aproximação (3.1).



**Figura 4.6.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.



**Figura 4.7.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo de Shan modificado [42] para sinal de entrada correlacionado.



**Figura 4.8.** Evolução de  $E[\mu^2(n)]$  (cinza) e  $E^2[\mu(n)]$  (preto) para o algoritmo de Shan modificado [42].

### 4.3 Algoritmo Proposto #1 (AP1) [29]

A utilização da função de autocorrelação entre  $e(n)$  e  $e(n-1)$  considerada no algoritmo de Aboulnasr [28], aqui denominada  $lag(1)$ , pode, em algumas situações, não ser o melhor indicador de aproximação do erro mínimo. Dependendo da correlação do sinal de entrada e do tipo de resposta ao impulso do sistema a ser identificado, a autocorrelação entre  $e(n)$  e  $e(n-2)$ ,  $e(n)$  e  $e(n-3)$  ou outro valor de atraso, pode ser tão ou mais significativa do que a autocorrelação  $lag(1)$ , para medir a proximidade do ponto ótimo. No algoritmo de Aboulnasr [28], podem ocorrer situações em que a função de autocorrelação do sinal de erro  $lag(1)$  reduza o passo de adaptação desnecessariamente, tornando o processo de convergência mais lento.

O algoritmo proposto neste trabalho, denominado algoritmo proposto #1 (AP1) visa utilizar todas as possíveis funções de autocorrelação do sinal de erro  $lag(1), lag(2), \dots, lag(N)$ , com  $N$  denotando a ordem do filtro adaptativo. A utilização de vários  $lags$  pouco aumenta a complexidade computacional do algoritmo. Além

disso, o processo de convergência é acelerado, mantendo-se uma boa robustez do algoritmo com relação ao ruído branco de medição.

Então seja  $p(n)$  o valor médio da soma dos quadrados das funções de autocorrelação entre  $e(n)$  e  $e(n-1)$ ,  $e(n-2)$ , ...,  $e(n-N)$  estimado através de

$$p(n) = \beta p(n-1) + (1-\beta) \sum_{i=1}^N [e(n)e(n-i)]^2 \quad (4.10)$$

a equação de adaptação do passo variável é então dada por

$$\mu(n+1) = \alpha \mu(n) + \gamma p(n) \quad (4.11)$$

onde  $0 < \beta < 1$  controla a estimação de  $p(n)$ , e  $\alpha$  e  $\gamma$  são os parâmetros da equação de atualização do passo de adaptação.

#### 4.3.1 Modelo Estocástico para o algoritmo AP1

Nesta seção, é apresentado um modelo estocástico para o algoritmo proposto AP1, desenvolvido neste trabalho de pesquisa.

Tomando o valor esperado de ambos os lados de (4.11), tem-se

$$E[\mu(n+1)] = \alpha E[\mu(n)] + \gamma E[p(n)]. \quad (4.12)$$

Assumindo então  $\mu(0) = 0$  e conseqüentemente  $E[\mu(0)] = 0$ , pode-se reescrever (4.12) como

$$E[\mu(n+1)] = \gamma \sum_{\ell=0}^n \alpha^\ell E[p(n-\ell)]. \quad (4.13)$$

Para se obter  $E[\mu(n+1)]$  é necessário obter inicialmente  $E[p(n)]$ . Tal determinação é apresentada no Apêndice B.

Então, substituindo (B.13) em (4.13) obtém-se a expressão de modelo para  $E[\mu(n+1)]$  para o algoritmo proposto AP1. Assim,

$$\begin{aligned} & E[\mu(n+1)] \\ &= \gamma \sum_{\ell=0}^n \alpha^\ell (1-\beta) \sum_{k=0}^{n-l-1} \beta^k \sum_{i=1}^N (\sigma_\eta^4 + \sigma_\eta^2 \{ \text{tr}[\mathbf{R}\mathbf{K}(n-l-k-i) + \mathbf{R}\mathbf{K}(n-l-k)] \}) \\ & \quad + \text{tr} \{ \mathbf{R}\mathbf{K}_i^\top (n-l-k) \} \mathbf{R}\mathbf{K}_i (n-l-k) \}. \end{aligned} \quad (4.14)$$

Observa-se através desse modelo, que apesar do algoritmo proposto AP1 melhorar o desempenho em relação ao algoritmo de Aboulnasr, ele ainda não é capaz de obter imunidade a mudanças na variância do ruído de medição, visto que o comportamento médio do passo de adaptação depende fortemente de  $\sigma_{\eta}^2$ .

### 4.3.2 Resultados de Simulação

Nesta seção, são apresentados os resultados de simulação numérica comparando o desempenho do algoritmo AP1 com os algoritmos de Kwong [22] e de Aboulnasr [28] (Exemplos 4.4 e 4.5) e atestando a precisão do modelo estocástico desenvolvido (Exemplos 4.6 e 4.7).

Exemplo 4.4: Neste exemplo, um problema de identificação de sistema é considerado, usando o mesmo cenário descrito em [28]. Os resultados de simulação mostram o comportamento do passo de adaptação, a evolução de um coeficiente do vetor  $\mathbf{w}(n)$  e o erro quadrático médio (EQM) em excesso, definido por  $E\{[e(n) - \eta(n)]^2\}$ . Também é analisado o comportamento do algoritmo frente a uma mudança abrupta no sistema a ser identificado. A planta considerada possui 4 coeficientes, dada pelo vetor  $\mathbf{w}_0 = [5 \ 0 \ 1 \ 8]^T$ . O sinal de entrada é branco  $\sigma_x^2 = 1$  e a variância do ruído de medição aditivo é  $\sigma_{\eta}^2 = 1$ , caracterizando uma SNR de 0dB. Resultados numéricos obtidos através de simulações de Monte Carlo (MC) - 200 rodadas independentes - comparam os algoritmos de Kwong [22] e Aboulnasr [28] com o algoritmo proposto AP1. Para todos algoritmos, o passo máximo de adaptação é limitado em 0,1. As Figuras 4.9 a 4.11 mostram, respectivamente, o comportamento do passo de adaptação, do coeficiente  $w_1(n)$  e do EQM em excesso. Na Figura 4.12, o comportamento do EQM em excesso é analisado considerando uma mudança abrupta nos parâmetros da planta. Essa mudança é obtida multiplicando, na iteração 20000, todos os seus coeficientes por  $-1$ . Através dessas figuras, verifica-se um melhor desempenho do algoritmo AP1, para o cenário de simulação.

Em todas as simulações deste exemplo, os parâmetros dos três algoritmos são ajustados conforme descrito na Tabela 4.1 visando obter o mesmo EQM em excesso final igual a  $-33\text{dB}$ .

Tabela 4.1. Parâmetros Usados para Simulação – Sinal de Entrada Branco

Algoritmo	Parâmetros
Kwong[22]	$\alpha = 0,97$ $\gamma = 7 \times 10^{-6}$
Aboulnasr[28]	$\alpha = 0,97$ $\beta = 0,99$ $\gamma = 8 \times 10^{-4}$
AP1	$\alpha = 0,97$ $\beta = 0,99$ $\gamma = 2 \times 10^{-6}$

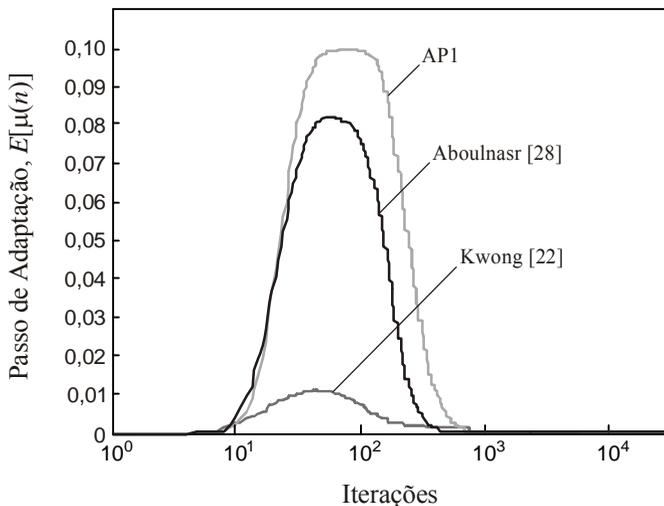
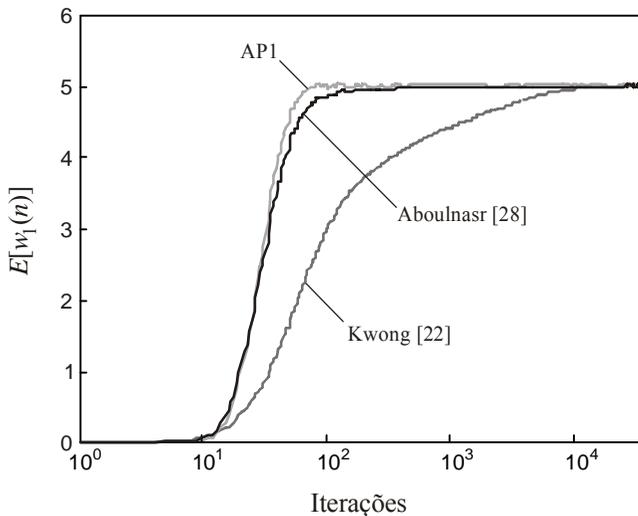
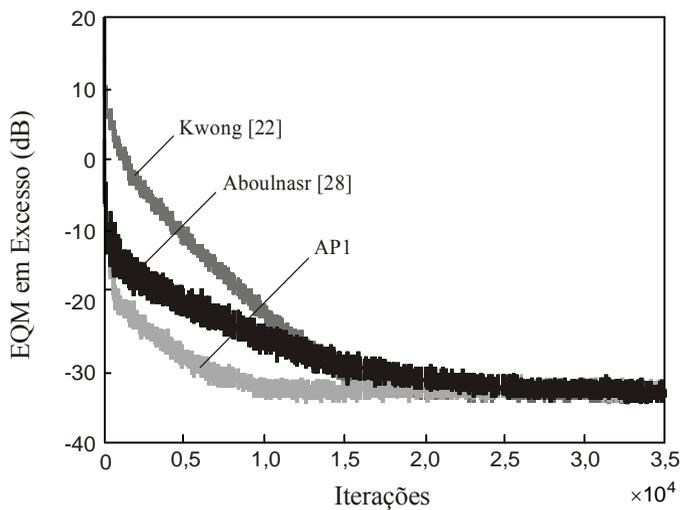


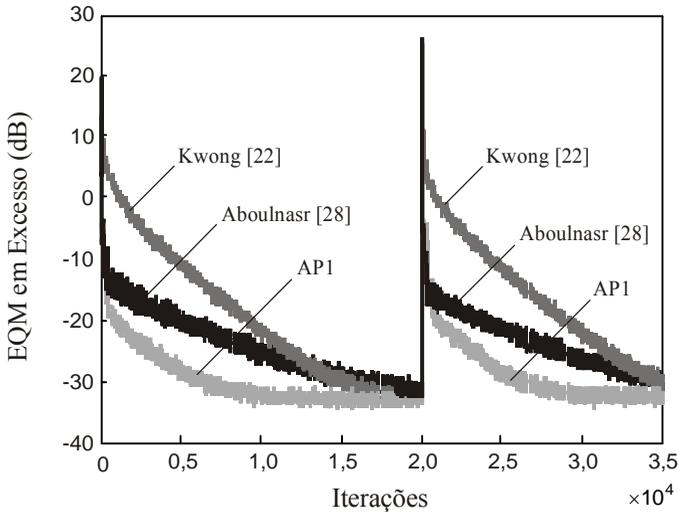
Figura 4.9. Evolução do passo variável para sinal de entrada branco.



**Figura 4.10.** Evolução do coeficiente  $w_1(n)$  para sinal de entrada branco.



**Figura 4.11.** Comportamento do EQM em excesso para sinal de entrada branco.



**Figura 4.12.** Comportamento do EQM em excesso para sinal de entrada branco considerando uma mudança abrupta no sistema a ser identificado na iteração 20000.

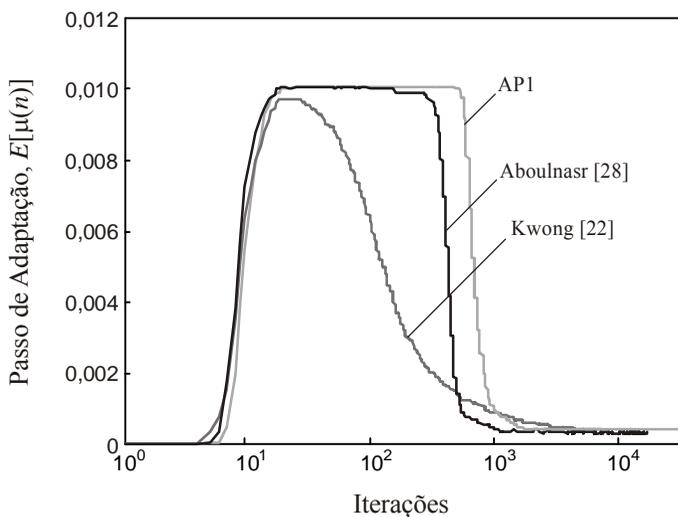
Exemplo 4.5: Neste exemplo, é utilizado o mesmo cenário do Exemplo 4.4, à exceção do sinal de entrada que agora é correlacionado, obtido de um processo AR(1) dado por  $x(n) = 0,9x(n-1) + u(n)$  com  $\sigma_x^2 = 5,26$ , sendo  $u(n)$  ruído branco gaussiano com variância  $\sigma_u^2 = 1$ . A dispersão dos autovalores da matriz de autocorrelação do sinal de entrada é  $\chi = 57,4$ . Novamente é realizada uma comparação entre o algoritmo proposto e os algoritmos de Kwong [22] e Aboulnasr [28]. Para todos os algoritmos, o passo máximo de adaptação é limitado em 0,01. As Figuras 4.13 a 4.15 mostram, respectivamente, a evolução do passo de adaptação, do coeficiente  $w_1(n)$  e do EQM em excesso. Na Figura 4.16, o comportamento do EQM em excesso é mostrado considerando uma mudança abrupta nos parâmetros da planta. Essa mudança é novamente obtida multiplicando, na iteração 20000, todos os seus coeficientes por  $-1$ . Os resultados numéricos obtidos comprovam o melhor desempenho do algoritmo AP1 em comparação com os algoritmos de Kwong [22] e Aboulnasr [28], no cenário simulado.

Nas simulações deste exemplo, os parâmetros dos três algoritmos são ajustados de maneira a se obter o mesmo EQM em excesso final

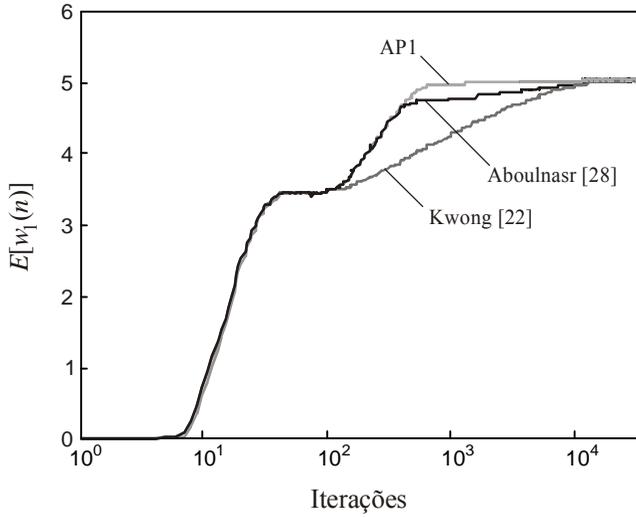
igual a  $-24$  dB. A Tabela 4.2 mostra os valores dos parâmetros considerados na simulação para os algoritmos avaliados.

**Tabela 4.2.** Parâmetros Usados na Simulação – Sinal de Entrada Correlacionado

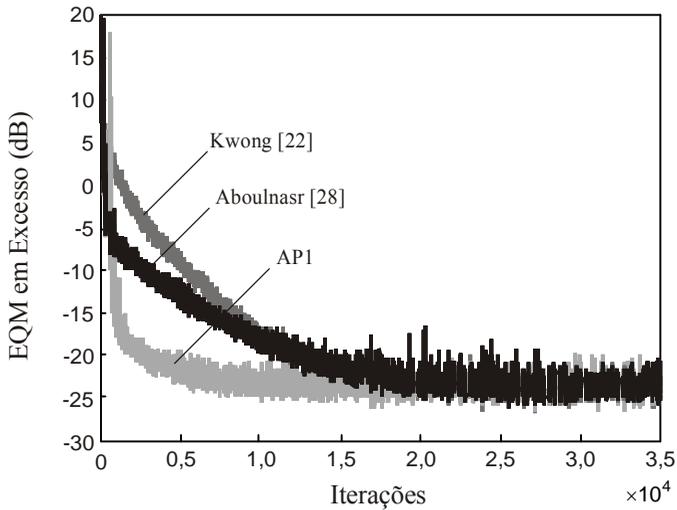
Algoritmo	Parâmetros
Kwong	$\alpha = 0,97$
	$\gamma = 1,1 \times 10^{-5}$
Aboulnasr	$\alpha = 0,97$
	$\beta = 0,99$
	$\gamma = 1,6 \times 10^{-3}$
AP1	$\alpha = 0,97$
	$\beta = 0,99$
	$\gamma = 3 \times 10^{-6}$



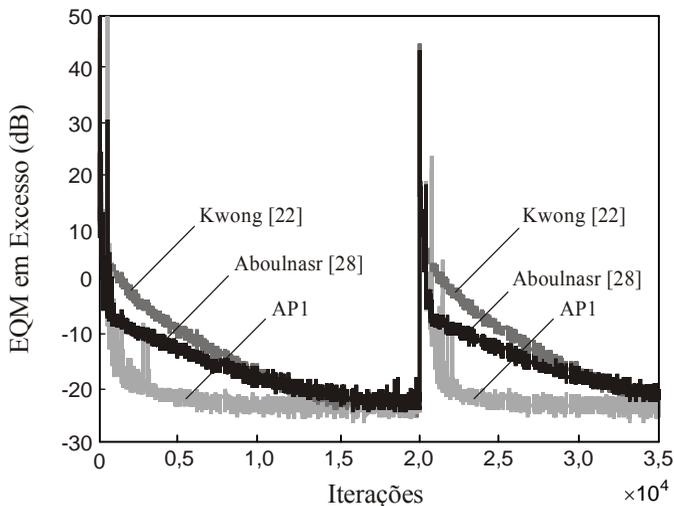
**Figura 4.13.** Evolução do passo variável para sinal de entrada correlacionado.



**Figura 4.14.** Evolução do coeficiente  $w_1(n)$  para sinal de entrada correlacionado.

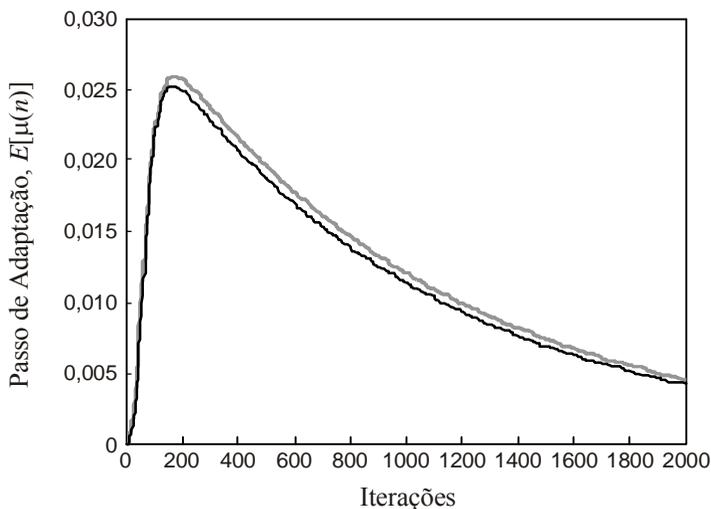


**Figura 4.15.** Comportamento do EQM em excesso para sinal de entrada correlacionado.

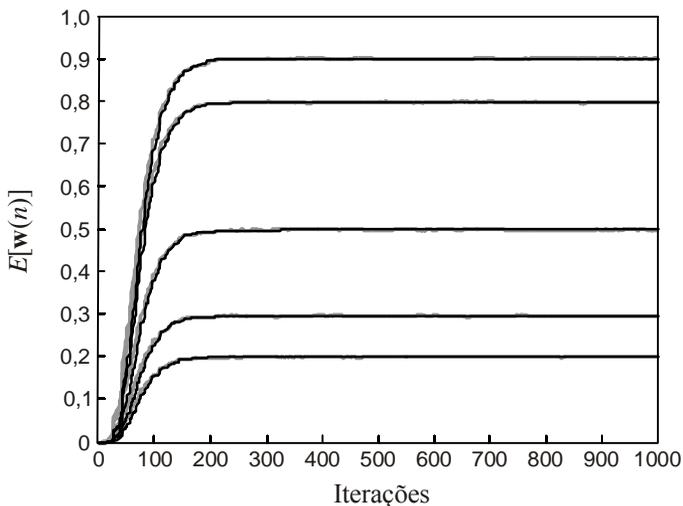


**Figura 4.16.** Comportamento do EQM em excesso para sinal de entrada correlacionado considerando uma mudança abrupta no sistema a ser identificado na iteração 20000.

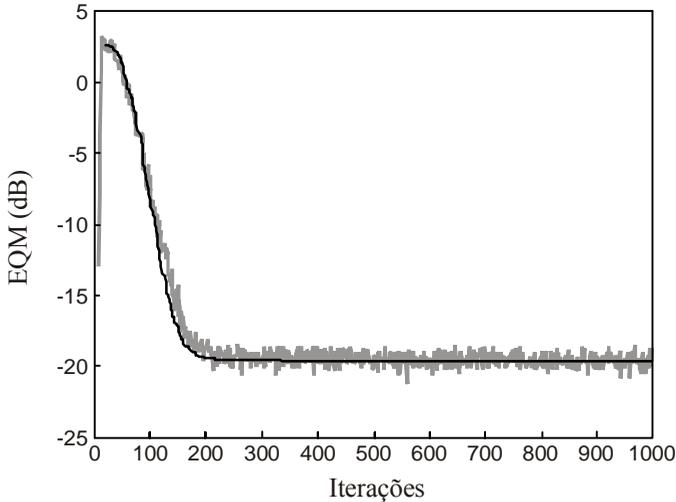
Exemplo 4.6: Neste exemplo, a precisão do modelo desenvolvido para o algoritmo proposto AP1 é verificada considerando um sinal de entrada branco e o mesmo cenário descrito no Exemplo 3.1. As Figuras 4.17, 4.18 e 4.19 mostram, respectivamente, o comportamento do passo de adaptação, do vetor de coeficientes e do EQM. Uma comparação dos resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto é novamente apresentada.



**Figura 4.17.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP1 para sinal de entrada branco.



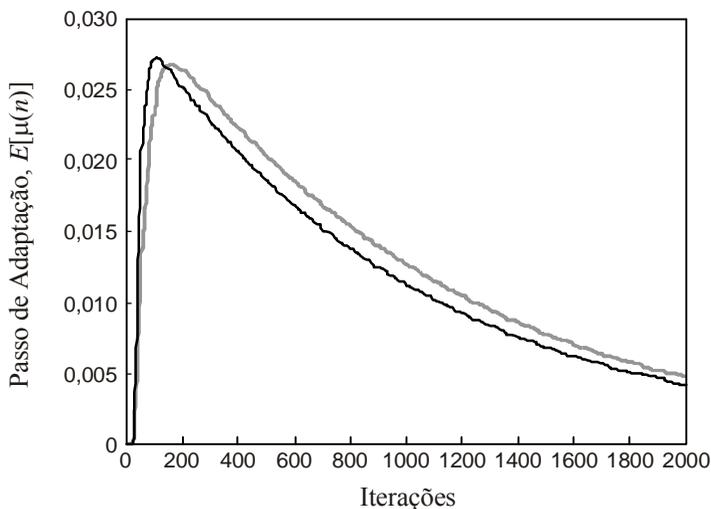
**Figura 4.18.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $\mathbf{w}(n)$  do algoritmo proposto AP1 para sinal de entrada branco.



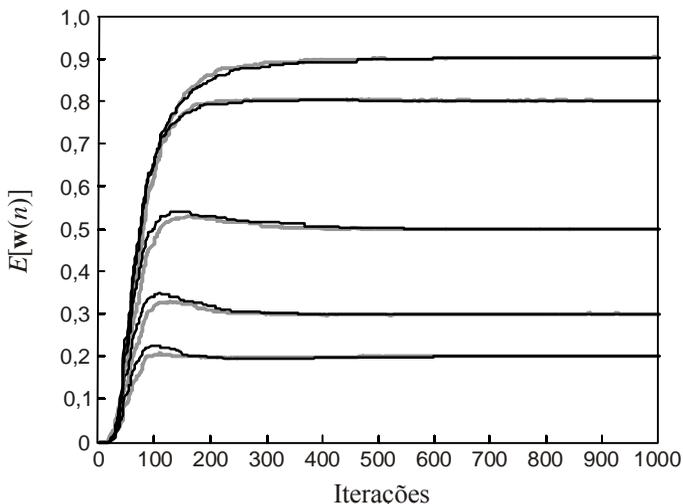
**Figura 4.19.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP1 para sinal de entrada branco.

Um casamento muito bom entre os resultados obtidos a partir da simulação de MC e aqueles obtidos pelo modelo proposto do algoritmo AP1 é observada nas Figuras 4.17 a 4.19, considerando um sinal de entrada branco.

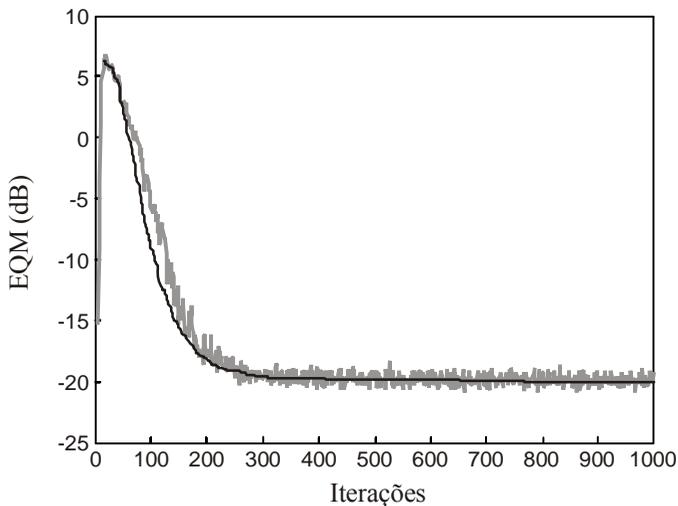
Exemplo 4.7: Aqui, a precisão do modelo desenvolvido para o algoritmo proposto AP1 é verificada considerando um sinal de entrada correlacionado e o mesmo cenário descrito no Exemplo 3.2. As Figuras 4.20, 4.21 e 4.22 mostram, respectivamente, o comportamento do passo de adaptação, vetor de coeficientes e EQM. Com isso, é possível uma comparação entre os resultados obtidos através de simulação de MC (200 realizações independentes) e pelo modelo proposto. A Figura 4.23 verifica por simulação a Aproximação (3.1), comparando as evoluções de  $E[\mu^2(n)]$  e  $E^2[\mu(n)]$ .



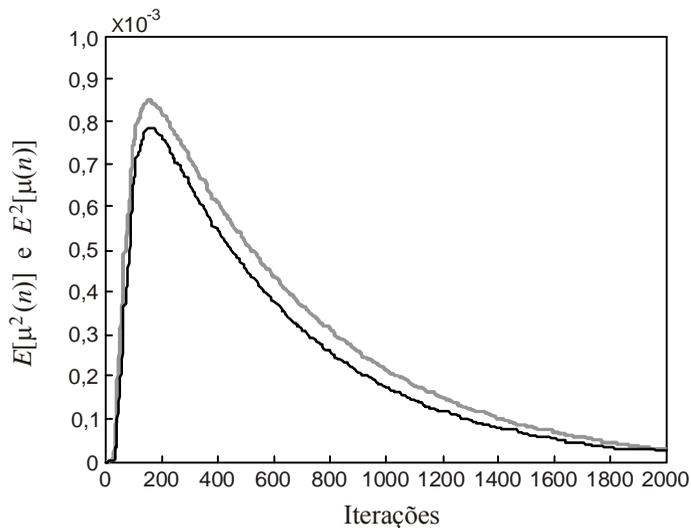
**Figura 4.20.** Comparação entre simulação (cinza) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP1 para sinal de entrada correlacionado.



**Figura 4.21.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $\mathbf{w}(n)$  do algoritmo proposto AP1 para sinal de entrada correlacionado.



**Figura 4.22.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP1 para sinal de entrada correlacionado.



**Figura 4.23.** Evolução de  $E[\mu^2(n)]$  (cinza) e  $E^2[\mu(n)]$  (preto) para o algoritmo proposto AP1.

Pode-se observar a partir das Figuras 4.20 a 4.22 um casamento satisfatório entre as curvas obtidas por simulação e pelo modelo proposto para o algoritmo AP1, considerando um sinal de entrada correlacionado. A utilização de aproximações utilizadas na obtenção do modelo estocástico do algoritmo AP1 é responsável pelos pequenos desvios observados nas Figuras 4.20 a 4.22. A Figura 4.23 verifica por simulação a Aproximação (3.1).

#### 4.4 Algoritmo Proposto #2 (AP2) [30]

O Algoritmo de Aboulnasr [28] foi proposto com o objetivo de obter um melhor desempenho em relação ao algoritmo de Kwong [22], para ambientes de baixa razão sinal-ruído (SNR). No entanto, a exemplo da maioria dos algoritmos VSSLMS, o algoritmo de Aboulnasr [28] apresenta uma forte dependência da flutuação do nível do ruído de medição, obrigando a ajustes de parâmetros sempre que ocorrem mudanças no nível de ruído. Por outro lado, se os parâmetros são mantidos constantes, o seu desempenho é significativamente afetado frente a diferentes níveis de SNR. Além disso, o ajuste de parâmetros é usualmente realizado utilizando procedimentos experimentais (baseados em tentativa e erro) revelando-se uma tarefa não trivial para a maioria das aplicações práticas.

Então, com o objetivo de melhorar a imunidade do algoritmo em relação a variações tanto na potência do ruído de medição quanto na potência do sinal de entrada, sem necessidade de ajuste de parâmetros, um novo algoritmo não-paramétrico VSS-NLMS é aqui proposto. Esse algoritmo [aqui denominado algoritmo proposto #2 (AP2)] também não requer conhecimento prévio da variância do ruído de medição, tal como exigido pelo algoritmo não-paramétrico proposto em [25].

Para tal, consideramos a seguinte lei de ajuste do passo variável:

$$\mu(n+1) = \frac{\alpha(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (4.15)$$

onde  $\alpha(n)$  é a função de controle do passo de adaptação.

Para atingir os objetivos esperados, é necessário determinar uma regra de variação para  $\alpha(n)$  que deve ser independente da variância do ruído de medição. Além disso, como trata-se de um algoritmo NLMS,  $\alpha(n)$  deve variar de 1 a zero conforme o algoritmo se aproxima do

ponto ótimo, sem a necessidade de ajuste de parâmetros. Para tal,  $\alpha(n)$  é definido como

$$\alpha(n) = \frac{p^2(n)}{q^2(n)} \quad (4.16)$$

com  $p(n)$  denotando uma estimativa da autocorrelação entre erros adjacentes obtida como em (3.45), e  $q(n)$  o sinal de erro ao quadrado suavizado, o qual é dado por

$$q(n) = \beta q(n-1) + (1-\beta)e^2(n) \quad (4.17)$$

onde  $\beta$  é uma constante próxima da unidade similar àquela usada para obter  $p(n)$  em (3.45). Observe que aqui  $\beta$  não é um parâmetro que deve ser ajustado para cada condição de operação; ele é uma constante que deve ser próxima, porém menor, do que a unidade, por exemplo,  $\beta = 0,99$ . Por esta razão, o algoritmo proposto é dito ser não-paramétrico.

Com o objetivo de obter estimativas suavizadas de ambos  $p(n)$  e  $q(n)$ , utiliza-se filtros passa-baixas em (3.45) e (4.17). O uso de médias temporais dos sinais  $p(n)$  e  $q(n)$  melhora o desempenho do algoritmo, reduzindo tanto o nível de ruído quanto grandes flutuações em tais estimativas.

A regra de ajuste (4.16) atua da seguinte forma: como no início do processo de convergência  $e(n)$  é aproximadamente igual a  $e(n-1)$ , a correlação entre eles tende a  $E[e^2(n)]$ , levando  $\alpha(n) \rightarrow 1$  e, com isso, acelerando o processo de convergência. À medida que o regime permanente se aproxima,  $e(n)$  tende a ser cada vez mais decorrelacionado, fazendo a correlação entre  $e(n)$  e  $e(n-1)$  tender para zero. Por outro lado, o valor médio final do sinal de erro ao quadrado  $e^2(n)$  tende para a variância do ruído de medição, fazendo  $\alpha(n) \rightarrow 0$  próximo de zero e reduzindo o desajuste final do algoritmo.

Na próxima seção, uma análise estatística desse algoritmo é apresentada confirmando sua eficácia quanto à imunidade ao nível (e variações) do ruído de medição.

#### 4.4.1 Modelo Estocástico para o Algoritmo AP2

O algoritmo proposto AP2, baseado na autocorrelação do sinal de erro, foi idealizado para melhorar a imunidade frente a mudanças na variância do ruído de medição. Com o objetivo de confirmar teoricamente a eficácia de tal algoritmo, uma análise estatística, visando avaliar a dependência do passo de adaptação variável em relação ao ruído de medição, é apresentada.

Para obter o valor esperado de (4.15), o Princípio da Média (AP) é invocado, resultando em

$$E[\mu(n+1)] \cong E[\alpha(n)]E\left[\frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)}\right]. \quad (4.18)$$

As variáveis  $p(n)$  e  $q(n)$  são, respectivamente, estimativas suavizadas da autocorrelação do erro e do erro ao quadrado, obtidas através de filtragem passa-baixas. Assume-se aqui que o processo de adaptação é lento. Com isso, pode-se utilizar a seguinte aproximação para calcular o valor esperado  $E[\alpha(n)]$ :

$$E[\alpha(n)] \cong \frac{E[p^2(n)]}{E[q^2(n)]} \quad (4.19)$$

onde  $E[p^2(n)]$  é obtido por (A.26) e o valor esperado  $E[q^2(n)]$  é determinado no Apêndice C.

Assim, a partir da hipótese (4.19),  $E[\alpha(n)]$  pode ser aproximadamente determinado por

$$\begin{aligned} E[\alpha(n)] &\cong \frac{\sum_{j=0}^{n-1} \beta^{2j} \{\sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j-1) + \mathbf{RK}(n)]\} + \sum_{j=1}^{n-1} \beta^{2j-1} \sigma_{\eta}^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T (n-j+1)]}{\sum_{j=0}^{n-1} \beta^{2j} \{3\sigma_{\eta}^4 + 6\sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j)]\} + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{\sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j) + \mathbf{RK}(n-k)]\}} \\ &+ \frac{\sum_{j=0}^{n-2} \beta^{2j+1} \sigma_{\eta}^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2 (n-j)]}{\sum_{j=0}^{n-1} \beta^{2j} \{3\sigma_{\eta}^4 + 6\sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j)]\} + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{\sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j) + \mathbf{RK}(n-k)]\}}. \end{aligned} \quad (4.20)$$

Assumindo ainda que  $\sigma_{\eta}^4$  é muito menor que os demais termos em (4.20),  $E[\alpha(n)]$  pode ser finalmente dado por

$$E[\alpha(n)] \cong \frac{\sum_{j=0}^{n-1} \beta^{2j} \{ \text{tr}[\mathbf{R}\mathbf{K}(n-j-1) + \mathbf{R}\mathbf{K}(n)] \} + \sum_{j=1}^{n-1} \beta^{2j-1} \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-j+1)] + \sum_{j=0}^{n-2} \beta^{2j+1} \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-j)]}{\sum_{j=0}^{n-1} \beta^{2j} 6 \text{tr}[\mathbf{R}\mathbf{K}(n-j)] + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}[\mathbf{R}\mathbf{K}(n-j) + \mathbf{R}\mathbf{K}(n-k)]}. \quad (4.21)$$

Observa-se desta análise que o algoritmo proposto AP2 apresenta maior imunidade a mudanças na variância do ruído de medição, já que o comportamento médio da função de controle do passo de adaptação independe de  $\sigma_{\eta}^2$ .

Esta proposta resolve dois grandes problemas que surgem no projeto de filtros adaptativos utilizando algoritmos VSSLMS. O primeiro é a difícil tarefa de ajustar parâmetros do algoritmo para uma determinada aplicação prática e o segundo diz respeito à forte dependência da maioria dos algoritmos VSSLMS frente a mudanças na variância do ruído de medição. O algoritmo proposto AP2 não requer qualquer ajuste de parâmetros, mantendo muito bom desempenho mesmo quando ocorrerem grandes mudanças na variância do ruído de medição.

#### 4.4.2 Resultados de Simulação

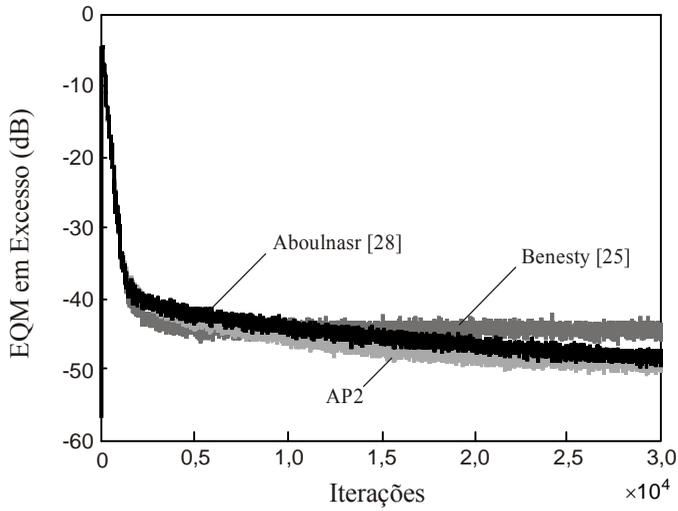
Nesta seção, são apresentados resultados de simulação para verificação de desempenho do algoritmo proposto AP2 nos quatro primeiros exemplos e para avaliação da precisão do modelo estocástico desenvolvido nos dois últimos. Nos Exemplos 4.7, 4.8 e 4.9, as simulações numéricas comparam o desempenho dos algoritmos de Aboulnasr [28] e Benesty [25], com o algoritmo proposto AP2. Para tal, simulações de MC (média de 200 rodadas independentes) são realizadas para um problema de identificação de sistemas, mostrando as curvas de EQM em excesso e do passo de adaptação variável  $\mu(n)$  para todos os algoritmos avaliados. Além disso, para o algoritmo proposto AP2, o comportamento médio da função de controle do passo  $\alpha(n)$  é também mostrado. No Exemplo 4.10, são comparados apenas os algoritmos não-paramétricos, isto é, o dado em [25] e AP2. Nesse exemplo, são mostradas as evoluções do EQM em excesso e  $\mu(n)$ , considerando uma

estimação imperfeita do nível de ruído para o algoritmo não-paramétrico dado em [25].

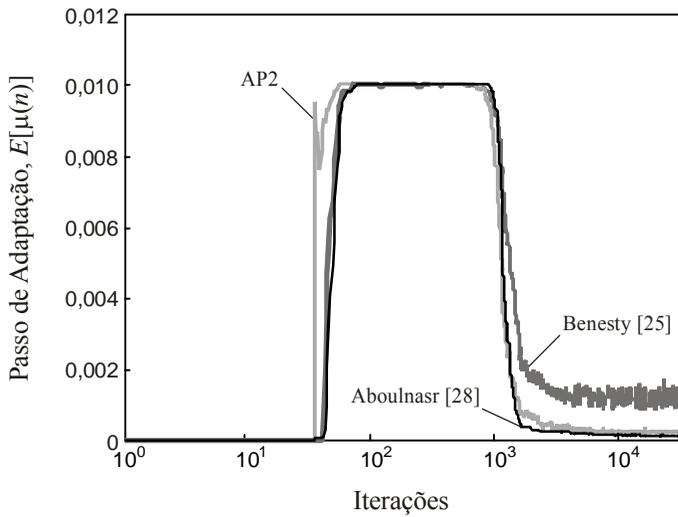
Nos quatro primeiros exemplos desta seção (Exemplos 4.7 – 4.10) utiliza-se uma planta com 32 coeficientes, obtida de uma função janela de Hanning, com  $\|\mathbf{w}_o\|=1$ . O sinal de entrada utilizado é gaussiano, de média zero, obtido de um processo AR(2) dado por  $x(n) = 1,1x(n-1) - 0,85x(n-2) + u(n)$ , onde  $u(n)$  é um ruído branco gaussiano com variância  $\sigma_u^2$ . A variância do sinal de entrada é  $\sigma_x^2 = 1$  e a dispersão dos autovalores da matriz de autocorrelação do sinal de entrada é  $\chi = 404,8$ . O passo de adaptação é limitado a um valor máximo de 0,01 para todos os algoritmos. Já nos Exemplos 4.11 e 4.12, que verificam a precisão do modelo estocástico desenvolvido, utiliza-se o mesmo cenário de simulação dos Exemplos 3.1 e 3.2.

Exemplo 4.7: Neste exemplo utiliza-se  $\sigma_\eta^2 = 0,001$  (SNR = 30dB) e o ajuste de parâmetros para o algoritmo de Aboulnasr [28] é feito manualmente de forma a se obter o mesmo EQM em excesso final do algoritmo proposto igual a -50dB. Para o algoritmo de Benesty [25] é necessário estimar a potência do ruído  $\hat{\sigma}_\eta^2$ , a qual é aqui assumida idêntica ao valor real de  $\sigma_\eta^2$ .

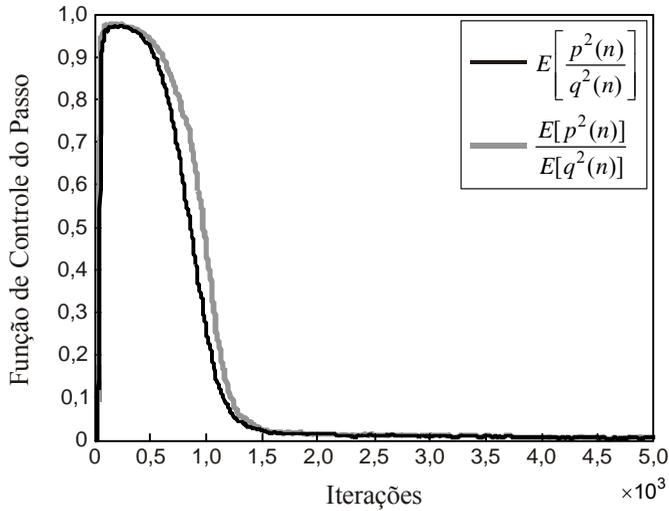
A Figura 4.24 ilustra os resultados obtidos do Exemplo 4.7. A Figura 4.24(a) apresenta a evolução do EQM em excesso comparando o desempenho dos algoritmos de Aboulnasr [28], Benesty [25] com o AP2. Dessa figura pode-se notar um desempenho similar entre os algoritmos avaliados. O algoritmo de Benesty [25] atinge o regime permanente antes, ao custo de um maior erro de desajuste. Na Figura 4.24(b), são apresentadas as curvas dos passos de adaptação dos três algoritmos. Observa-se que em todas as curvas o valor máximo é atingido. Em geral, todos algoritmos avaliados mostram o mesmo comportamento do passo de adaptação variável, exceto pelo fato de que o algoritmo de Benesty [25] exibe um valor final do passo mais elevado, implicando um maior desajuste [veja Figura 4.24(a)]. Já a Figura 4.24(c) ilustra o comportamento médio da função de controle de passo  $\alpha(n)$  e seu valor aproximado por (4.19). Dessa figura observa-se que  $\alpha(n)$  varia de 1 a zero (ou muito próximo de zero) à medida que se aproxima o ponto ótimo de convergência e que (4.19) é uma hipótese razoável para obtenção do modelo.



(a)



(b)

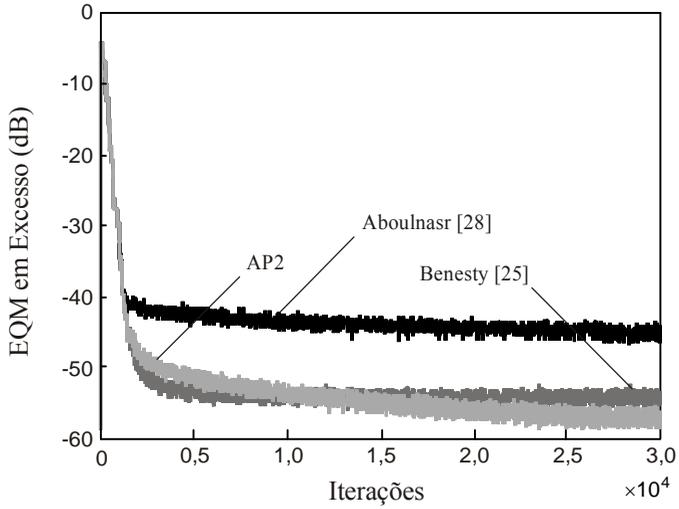


(c)

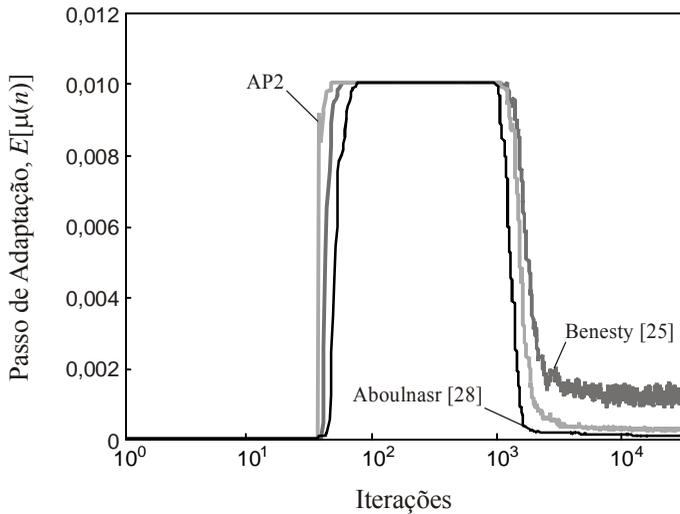
**Figura 4.24.** Resultados de simulação para  $\sigma_{\eta}^2 = 0,001$  (SNR = 30 dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle  $\alpha(n)$  e sua aproximação.

Exemplo 4.8: Neste exemplo, utiliza-se  $\sigma_{\eta}^2 = 0,0001$  (SNR = 40 dB) e os parâmetros do algoritmo de Aboulnasr [28] são mantidos constantes, com os mesmos valores usados no Exemplo 4.8. Para o algoritmo de Benesty [25] é novamente assumido que  $\hat{\sigma}_{\eta}^2 = \sigma_{\eta}^2$ .

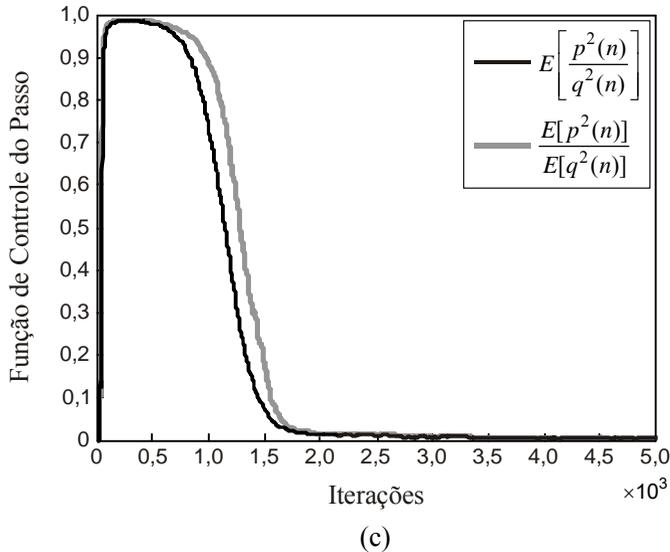
A Figura 4.25 apresenta os resultados obtidos para este exemplo. As Figuras 4.25(a) e (b) ilustram, respectivamente, a evolução do EQM em excesso e do passo de adaptação para os três algoritmos considerados. Dessas figuras, nota-se que o algoritmo de Aboulnasr [28] apresenta um maior EQM em excesso, causado por uma redução prematura do passo de adaptação, quando comparado, aos algoritmos não-paramétricos ([25] e AP2), que apresentam desempenho superior neste cenário. Essa degradação de desempenho do algoritmo de Aboulnasr [28] ocorre pela mudança do nível de ruído sem o correspondente reajuste de parâmetros. Seguindo o mesmo padrão da Figura 4.24(c), a Figura 4.25(c) apresenta a evolução de  $\alpha(n)$  e sua aproximação para o algoritmo proposto.



(a)



(b)

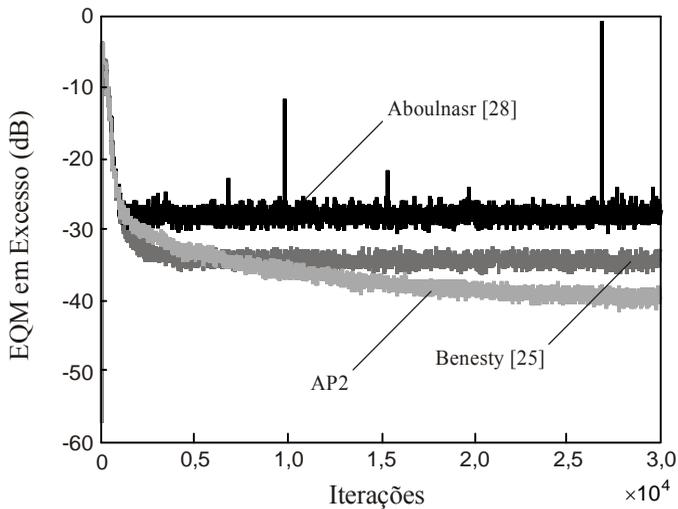


**Figura 4.25.** Resultados de simulação para  $\sigma_{\eta}^2 = 0,0001$  (SNR = 40 dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle  $\alpha(n)$  e sua aproximação.

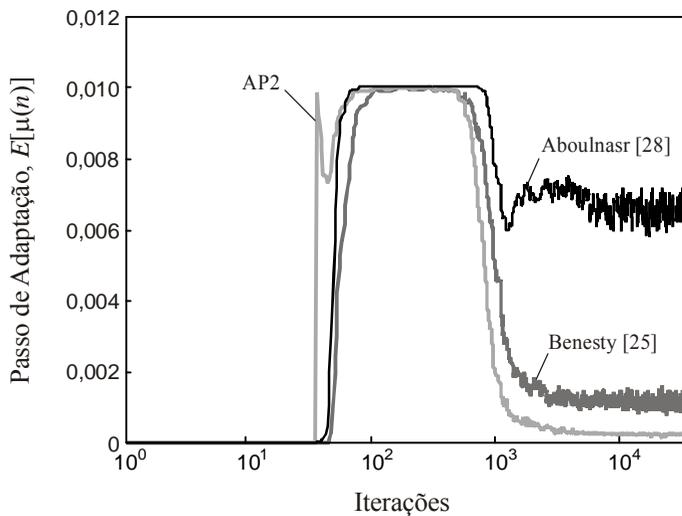
Exemplo 4.9: Para este exemplo, considera-se  $\sigma_{\eta}^2 = 0,01$  (SNR = 20 dB) e os parâmetros do algoritmo de Aboulansr permanecem constantes e iguais aos utilizados nos Exemplos 4.7 e 4.8. Para o algoritmo de Benesty [25] novamente é assumida estimação perfeita do nível de ruído, com  $\hat{\sigma}_{\eta}^2 = \sigma_{\eta}^2$ .

A Figura 4.26 ilustra os resultados obtidos neste exemplo. A evolução do EQM em excesso e o comportamento do passo variável dos três algoritmos são mostrados, respectivamente, nas Figuras 4.26(a) e (b). Observa-se dessas figuras que o algoritmo de Aboulansr [28] apresenta um maior EQM em excesso, causado por valores elevados do passo de adaptação [trabalhando próximo ao limite de estabilidade, veja Figura 4.26(b)], quando comparado aos algoritmos não-paramétricos ([25] e AP2). Esse comportamento do algoritmo de Aboulansr [28] se deve ao aumento do nível de ruído, sem o necessário reajuste de seus parâmetros. Além disso, observa-se ainda das Figuras 4.26(a) e (b) um desempenho superior do algoritmo AP2 nesse cenário. A Figura 4.26(c)

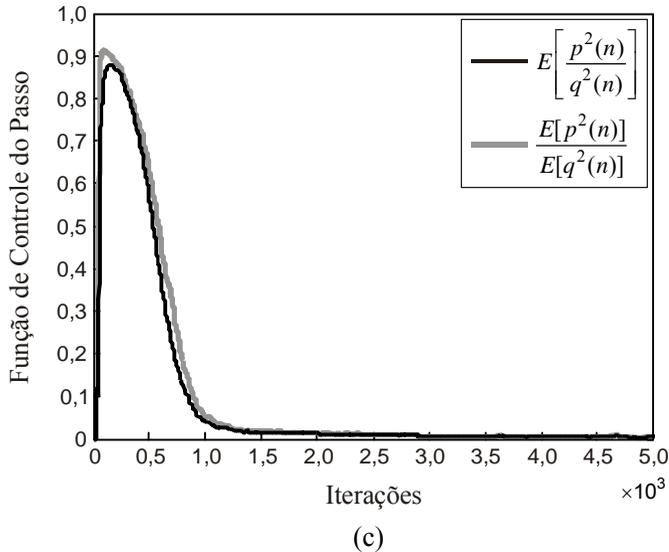
apresenta a evolução de  $\alpha(n)$  e seu valor aproximado por (4.19) para o algoritmo AP2, apresentando o mesmo padrão de curva observado nos exemplos anteriores.



(a)



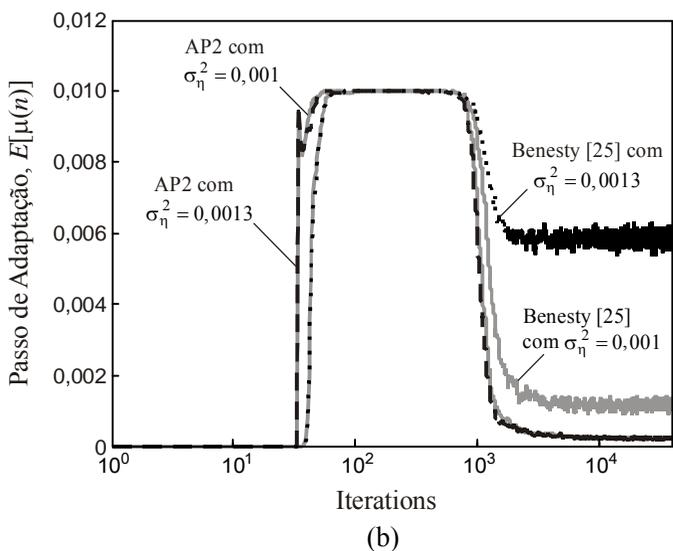
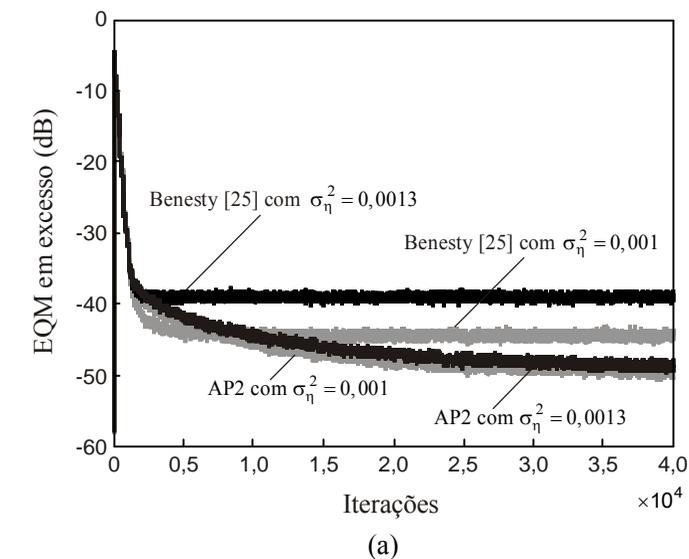
(b)



**Figura 4.26.** Resultados de simulação para  $\sigma_{\eta}^2 = 0,01$  (SNR = 20dB). (a) Curva de EQM em excesso. (b) Evolução do passo variável. (c) Evolução da função de controle  $\alpha(n)$  e sua aproximação.

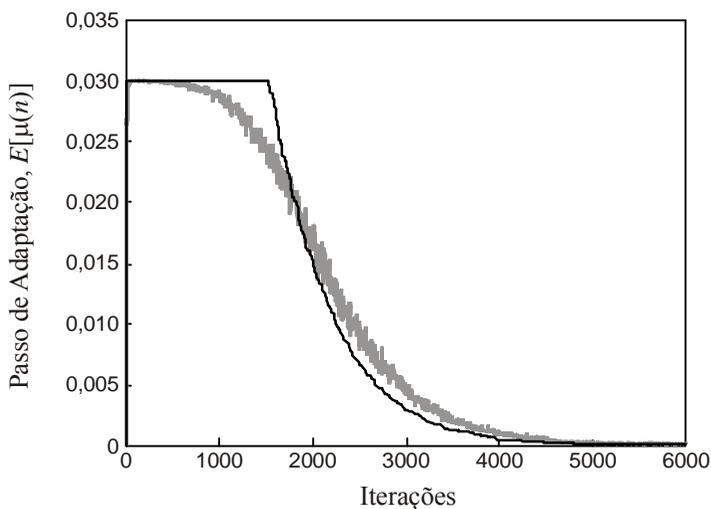
Exemplo 4.10: Este exemplo compara o desempenho dos dois algoritmos não-paramétricos ([25] e AP2), considerando um caso de estimação imperfeita do nível de ruído pelo algoritmo de [25]. Para tal, dois cenários de simulação são considerados levando em conta dois valores diferentes de variâncias do ruído  $\sigma_{\eta}^2 = 0,001$  e  $0,0013$ . Para ambos cenários o algoritmo de Benesty [25] opera com um valor fixo de  $\hat{\sigma}_{\eta}^2 = 0,001$ .

A Figura 4.27 ilustra os resultados obtidos neste exemplo. Nas Figuras 4.27(a) e (b) são mostrados, respectivamente, a evolução do EQM em excesso e o comportamento do passo de adaptação, para ambos algoritmos considerando os dois cenários de simulação. Observa-se dessas figuras uma degradação de desempenho do algoritmo de Benesty [25] quando a variância do ruído é aumentada em 30%, sem a correspondente modificação de  $\hat{\sigma}_{\eta}^2$ . Por outro lado, o algoritmo AP2 apresenta praticamente o mesmo desempenho nos dois cenários, confirmando sua boa imunidade a variações no nível de ruído de medição.

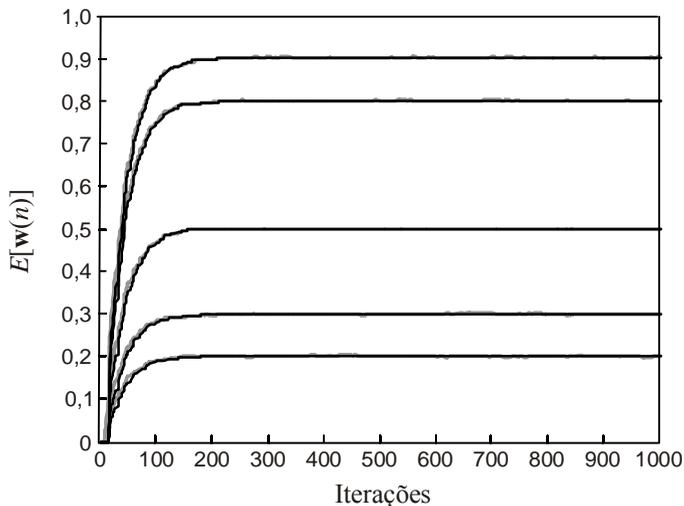


**Figura 4.27.** Resultados de simulação considerando dois níveis de ruído  $\sigma_{\eta}^2 = 0,001$  e  $0,0013$ , e algoritmo de Benesty [25] operando com valor fixo de  $\hat{\sigma}_{\eta}^2 = 0,001$ . (a) Curva de EQM em excesso. (b) Evolução do passo variável.

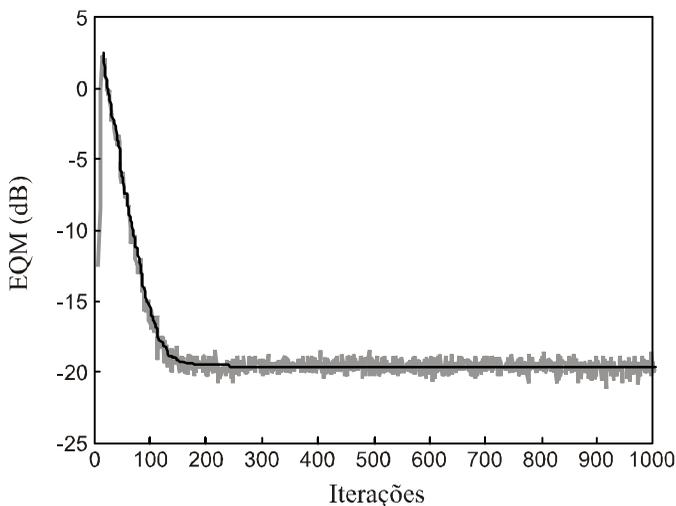
**Exemplo 4.11:** Neste exemplo, a precisão do modelo desenvolvido para o algoritmo proposto AP2 é verificada considerando um sinal de entrada branco e o mesmo cenário descrito no Exemplo 3.1. As Figuras 4.28, 4.29 e 4.30 mostram, respectivamente, o comportamento do passo de adaptação, do vetor de coeficientes e do EQM obtidos através de simulações de MC (200 realizações independentes) e pelo modelo estocástico desenvolvido. Dessa forma, é possível uma comparação das curvas e conseqüente verificação da precisão do modelo.



**Figura 4.28.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP2 para sinal de entrada branco.



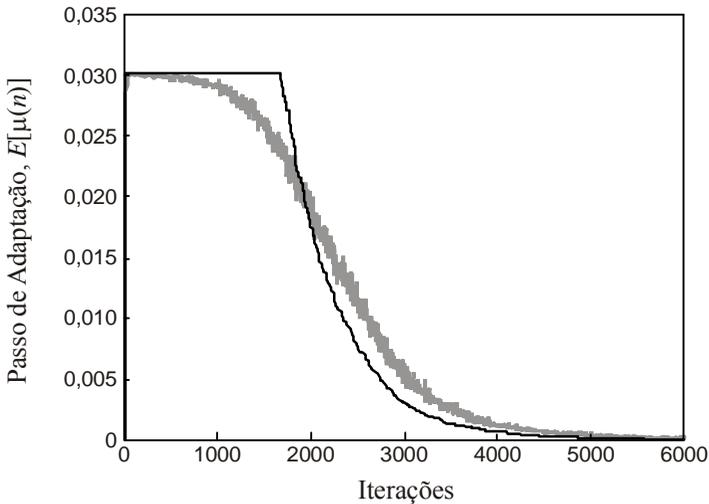
**Figura 4.29.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo proposto AP2 para sinal de entrada branco.



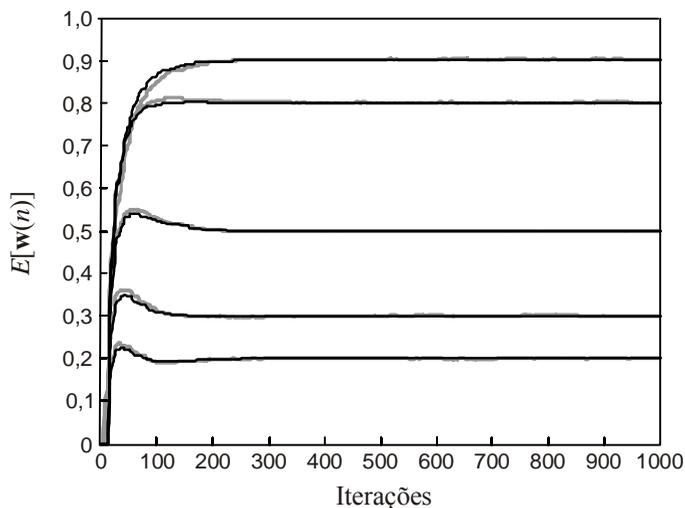
**Figura 4.30.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP2 para sinal de entrada branco.

Observa-se da Figura 4.28 um casamento apenas regular entre simulação e modelo para a evolução do passo de adaptação. Novamente, as aproximações utilizadas justificam os desvios observados nessa curva, especialmente a hipótese de adaptação lenta, uma vez que o algoritmo proposto AP2 opera sempre com alta velocidade de convergência. Já nas Figuras 4.29 e 4.30 pode-se observar um casamento muito bom entre os resultados obtidos a partir da simulação de MC e aqueles obtidos pelo modelo proposto do algoritmo AP2, para um sinal de entrada branco.

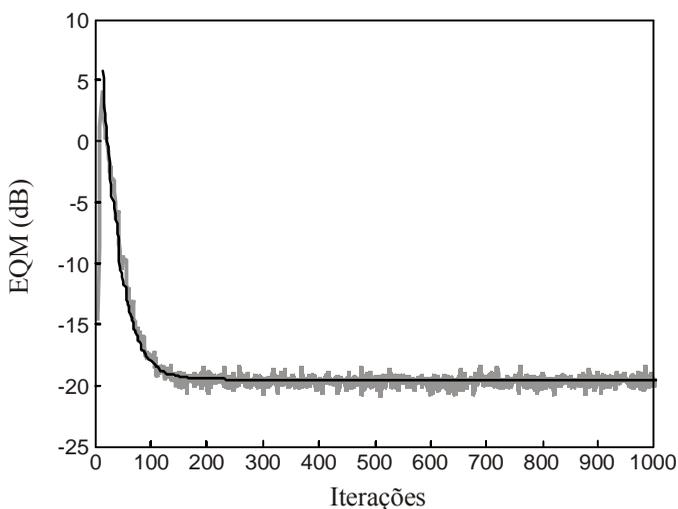
Exemplo 4.12: Neste exemplo, a precisão do modelo desenvolvido para o algoritmo proposto AP2 é verificada considerando um sinal de entrada correlacionado e o mesmo cenário descrito no Exemplo 3.2. As Figuras 4.31, 4.32 e 4.33 mostram, respectivamente, o comportamento do passo de adaptação, vetor de coeficientes e EQM para o algoritmo proposto AP2 (modelo e simulação de MC). Assim, é novamente possível uma comparação entre as curvas obtidas e uma verificação da precisão do modelo. A Figura 4.34 verifica por simulação a Aproximação (3.1), comparando as evoluções de  $E[\mu^2(n)]$  e  $E^2[\mu(n)]$ .



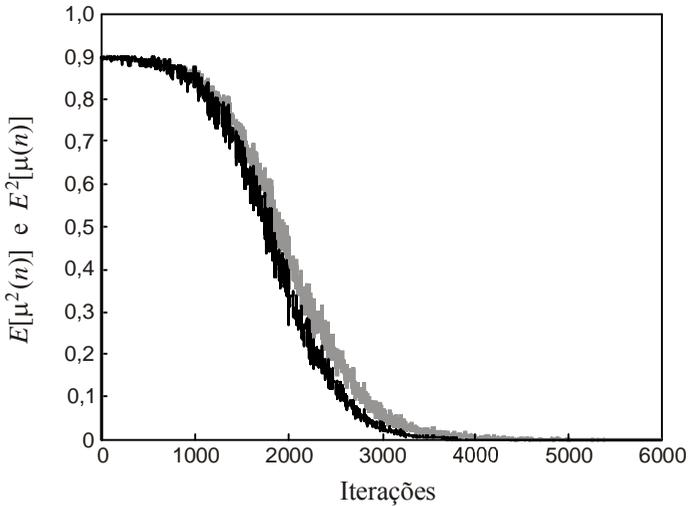
**Figura 4.31.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do passo de adaptação do algoritmo proposto AP2 para sinal de entrada correlacionado.



**Figura 4.32.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do vetor  $w(n)$  do algoritmo proposto AP2 para sinal de entrada correlacionado.



**Figura 4.33.** Comparação entre simulação (cinza irregular) e modelo (preto) da evolução do EQM do algoritmo proposto AP2 para sinal de entrada correlacionado.



**Figura 4.34.** Evolução de  $E[\mu^2(n)]$  (cinza) e  $E^2[\mu(n)]$  (preto) para o algoritmo proposto AP2.

Para o cenário considerado, novamente se observa da Figura 4.31 um casamento apenas regular entre simulação de MC e modelo para a evolução do passo de adaptação, sendo a justificativa para esse desvio a mesma já descrita no exemplo anterior. Em contrapartida, as Figuras 4.32 e 4.33 apresentam um casamento muito bom entre os resultados de simulação e do modelo proposto para as evoluções do vetor de coeficientes e do EQM, respectivamente. Aqui, novamente a Figura 4.34 verifica através de simulação a aproximação (3.1), mostrando um aceitável casamento entre os resultados obtidos.

## 4.5 Conclusões

Este capítulo tratou de novos algoritmos VSSLMS. Foi apresentada uma modificação do algoritmo LMS de passo variável proposto por Shan [17]. Tal modificação visou o aprimoramento desse algoritmo com vistas à sua estabilidade. Com a nova expressão de ajuste do passo de adaptação, fruto da modificação, um modelo para o algoritmo de Shan (modificado) é determinado. Também foram apresentados dois novos algoritmos baseados na autocorrelação do erro (algoritmos AP1 e AP2). O algoritmo AP1 considera uma função de

autocorrelação com  $N$  lags, melhorando o desempenho do algoritmo de Aboulnasr [28] na presença de ruído de medição. Já o algoritmo AP2 apresenta elevada imunidade frente a variações no nível de ruído de medição, além de não requerer o ajuste de qualquer parâmetro para sua operação.

Modelos estocásticos para predição do comportamento médio do vetor de coeficientes e curva de aprendizagem foram apresentados para os três algoritmos desenvolvidos. A partir de simulações numéricas, foi verificado o desempenho dos novos algoritmos e a precisão dos modelos propostos, quando confrontados com resultados obtidos a partir de simulações de Monte Carlo.

## Resultados Comparativos de Algoritmos VSSLMS

### 5.1 Introdução

Este capítulo apresenta uma comparação entre os principais algoritmos VSSLMS publicados na literatura técnica e discutidos no Capítulo 3, juntamente com os novos algoritmos propostos neste trabalho de pesquisa, os quais foram apresentados no Capítulo 4. Tal comparação está baseada na análise da complexidade computacional dos algoritmos, bem como em resultados de simulação considerando um cenário comum de aplicação.

Os resultados de simulação são apresentados por categoria, obedecendo a classificação proposta no Capítulo 3. Além disso, uma comparação entre os algoritmos de melhor desempenho em cada categoria é também realizada.

### 5.2 Complexidade Computacional de Algoritmos VSSLMS

A grande maioria dos algoritmos VSSLMS estudados apresenta apenas um pequeno aumento de complexidade computacional, quando comparada ao LMS convencional. Em todos os algoritmos estudados, a complexidade computacional é da ordem  $O(N)$ .

A Tabela 5.1 apresenta um quadro comparativo da complexidade computacional de todos os algoritmos estudados neste trabalho.

### 5.3 Resultados de Simulação – Comparação dos Algoritmos VSSLMS

Nesta seção, são apresentadas simulações numéricas visando uma comparação de desempenho de todos os algoritmos VSSLMS analisados neste trabalho. Para tanto, um problema de identificação de sistema é novamente considerado.

Exemplo 5.1: A planta utilizada é obtida a partir de uma função janela de Hanning com 10 coeficientes, dada pelo vetor  $\mathbf{w} = [0,0000 \ 0,1170 \ 0,4132 \ 0,7500 \ 0,9698 \ 0,9698 \ 0,7500 \ 0,4132 \ 0,1170 \ 0,0000]^T$ . O sinal de entrada utilizado é correlacionado com variância  $\sigma_x^2 = 1$ , obtido a partir de um processo AR(1), dado por

$x(n) = 0,6x(n-1) + u(n)$ , considerando  $u(n)$  um ruído branco gaussiano com  $\sigma_u^2 = 0,65$ . A dispersão dos autovalores da matriz de autocorrelação do sinal de entrada é  $\chi = 13,2$ . A variância do ruído aditivo é  $\sigma_\eta^2 = 1$ , caracterizando uma SNR de 0 dB. Para avaliar o comportamento dos diversos algoritmos frente a uma mudança abrupta na planta, realiza-se também uma simulação que modifica os parâmetros do sistema a ser identificado durante o processo de adaptação. Na iteração  $n = 10000$ , a planta é mudada para  $-w_0$ , alterando assim o sinal de todos os coeficientes. Em todas as simulações, os algoritmos são ajustados para se obter o mesmo EQM em excesso.

Tabela 5.1. Quadro Comparativo da Complexidade Computacional de Algoritmos VSSLMS

Algoritmo	Número de Multiplicações	Número de Divisões	Número de Somas e Subtrações	Número de Raízes Quadradas
LMS Convencional	$2N+1$	0	$2N$	0
Shan Modificado	$2N+5$	1	$3N+1$	0
Richards	$6N+1$	0	$2N+2$	0
Mathews	$3N+4$	0	$2N+1$	0
Wee-Peng Ang	$5N+3$	0	$2N+2$	0
Farhang	$8N+1$	$N$	$3N$	0
Okello	$5N+3$	0	$4N+1$	0
Shin	$7N+2$	2	$3N+1$	0
Kwong	$2N+4$	0	$2N+1$	0
Nakanishi	$2N+5$	1	$2N+1$	0
Costa	$3N+6$	0	$2N+2$	0
Benesty(Não-Param.)	$3N+5$	2	$3N+1$	1
Aboulnasr	$2N+7$	0	$2N+2$	0
AP1	$3N+6$	0	$3N+3$	0
AP2	$3N+9$	2	$3N+1$	0
Kim	$2N+3$	1	$2N$	0
Gollamudi	$2N+1$	1	$2N+1$	0
Ramadan	$4N+4$	1	$3N+2$	0
Rohani	$3N+1$	0	$3N$	0
Benesty (IPNLMS)	$6N+4$	3	$4N+3$	0
Yan-Bin	$3N+7$	1	$2N+4$	0
Wei	$2N+6$	1	$2N+4$	0

### 5.3.1 Algoritmos Baseados no Gradiente do Erro Quadrático

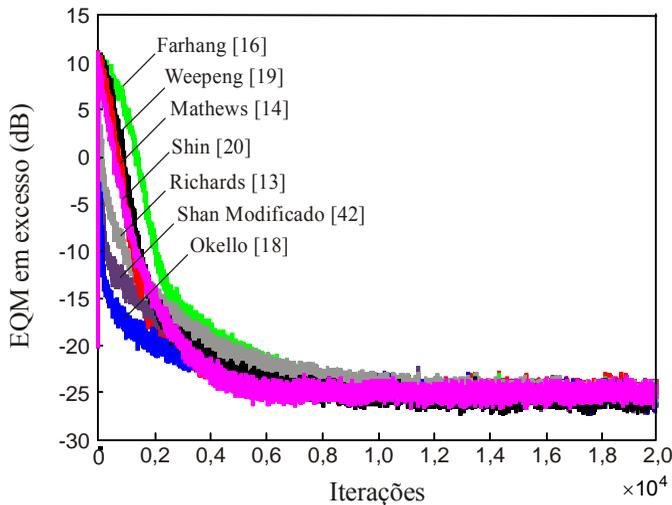
A Figura 5.1 mostra o comportamento do EQM em excesso para os algoritmos da categoria gradiente do erro quadrático. Na Figura 5.2, a evolução do EQM em excesso dos mesmos algoritmos é analisada considerando uma mudança abrupta dos parâmetros do sistema desconhecido através da multiplicação de todos os coeficientes da planta por  $-1$  na iteração 10000.

Observa-se, a partir das Figuras 5.1 e 5.2, o melhor desempenho do algoritmo de Okello [18] para o cenário de simulação considerado.

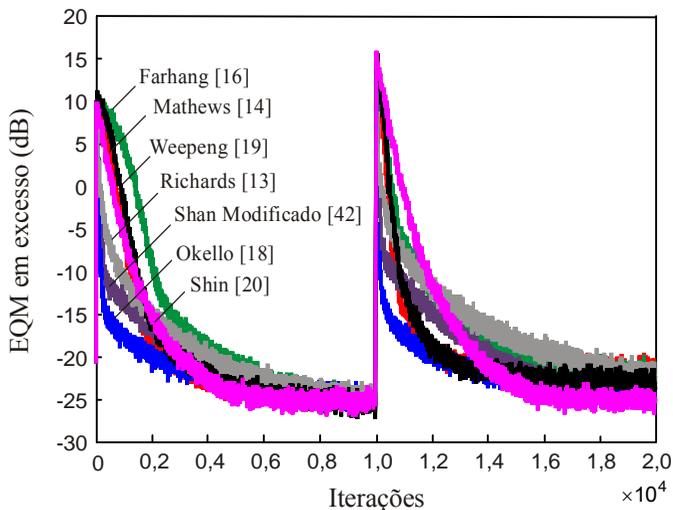
### 5.3.2 Algoritmos Baseados no Erro Quadrático

A Figura 5.3 mostra o comportamento do EQM em excesso para os algoritmos baseados no erro quadrático. Na Figura 5.4, a evolução do EQM em excesso desses algoritmos é analisada considerando uma mudança abrupta dos parâmetros do sistema desconhecido através da multiplicação de todos os seus coeficientes por  $-1$  na iteração 10000.

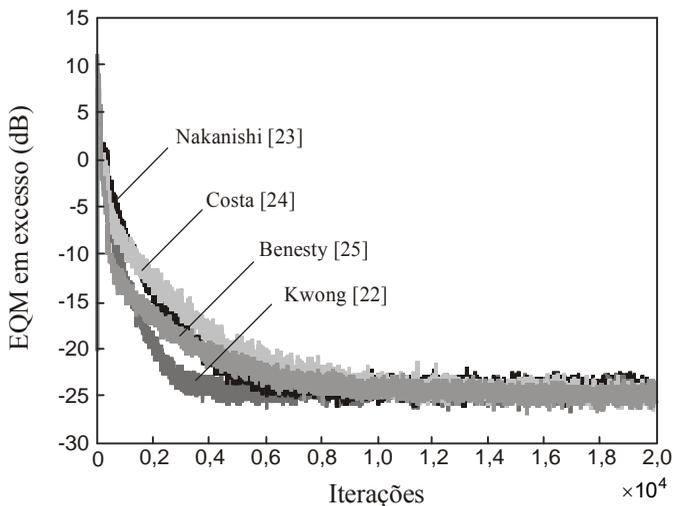
No cenário considerado, o algoritmo de Kwong [22] é que apresenta o melhor desempenho, como mostram as Figuras 5.3 e 5.4.



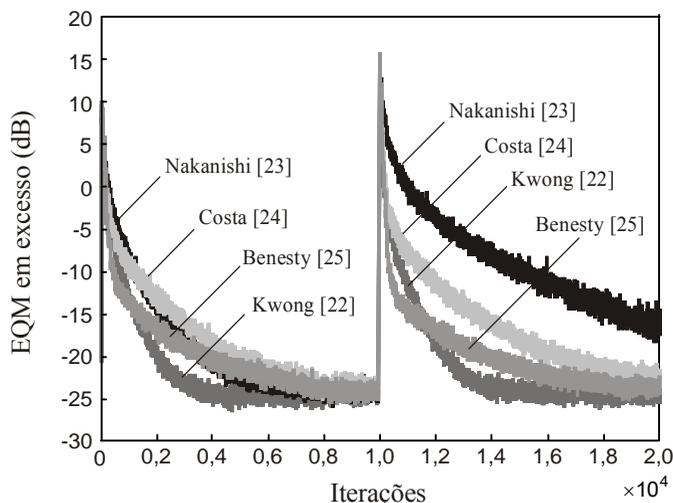
**Figura 5.1.** Comportamento do EQM em excesso dos algoritmos baseados no gradiente do erro quadrático.



**Figura 5.2.** EQM em excesso dos algoritmos baseados no gradiente do erro quadrático considerando uma mudança abrupta no sistema a ser identificado.



**Figura 5.3.** Comportamento do EQM em excesso dos algoritmos baseados no erro quadrático.



**Figura 5.4.** EQM em excesso dos algoritmos baseados no erro quadrático considerando uma mudança abrupta no sistema a ser identificado.

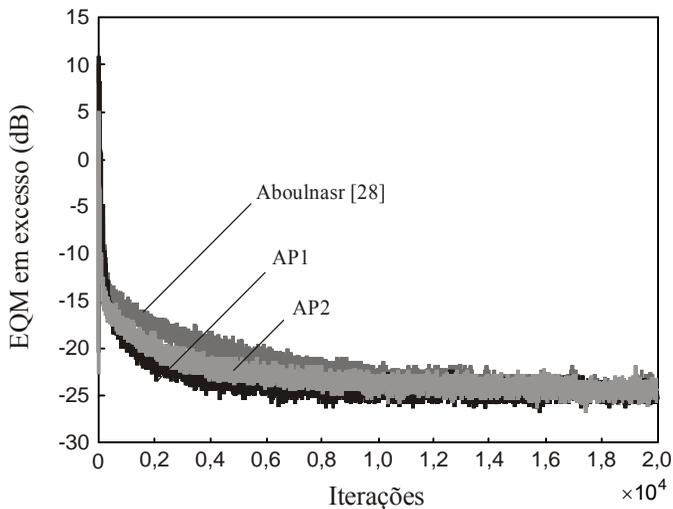
### 5.3.3 Algoritmos Baseados na Autocorrelação do Erro

A Figura 5.5 mostra a evolução do EQM em excesso para os algoritmos baseados na autocorrelação do erro. Já a evolução do EQM considerando uma mudança abrupta dos parâmetros do sistema desconhecido, é visualizada na Figura 5.6.

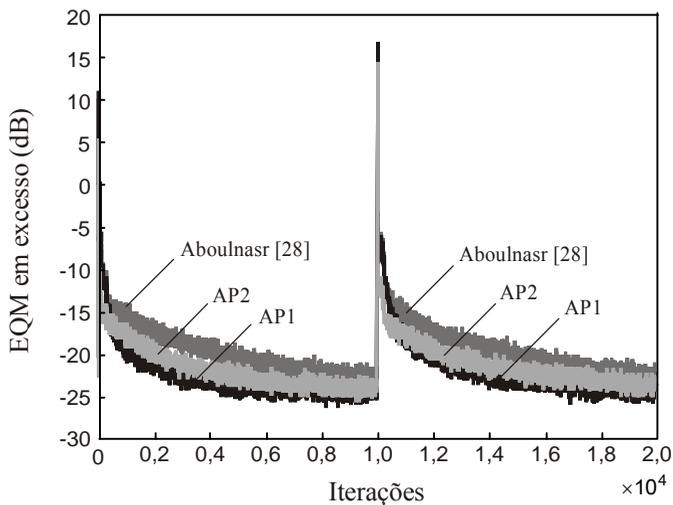
Observa-se das Figuras 5.5 e 5.6 o melhor desempenho do algoritmo AP1 para o cenário considerado.

### 5.3.4 Algoritmos Baseados no Valor Absoluto do Erro

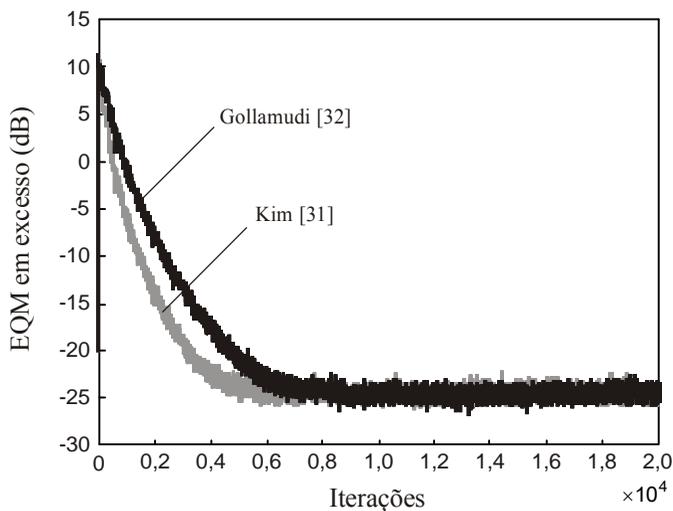
Nesta seção, o comportamento do EQM em excesso para os algoritmos baseados no valor absoluto do erro é analisado. Tal comportamento é ilustrado na Figura 5.7. Quando provocada uma mudança abrupta dos parâmetros do sistema desconhecido, o EQM em excesso comporta-se como mostrado na Figura 5.28. Observa-se nas Figuras 5.7 e 5.8 que o algoritmo de Kim [31] mostra uma convergência mais rápida.



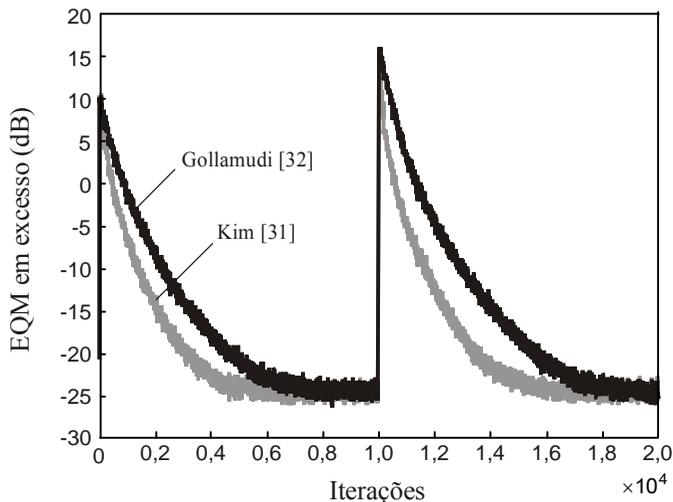
**Figura 5.5.** Comportamento do EQM em excesso dos algoritmos baseados na autocorrelação do erro.



**Figura 5.6.** EQM em excesso dos algoritmos baseados na autocorrelação do erro considerando uma mudança abrupta no sistema a ser identificado.



**Figura 5.7.** Comportamento do EQM em excesso dos algoritmos baseados no valor absoluto do erro.



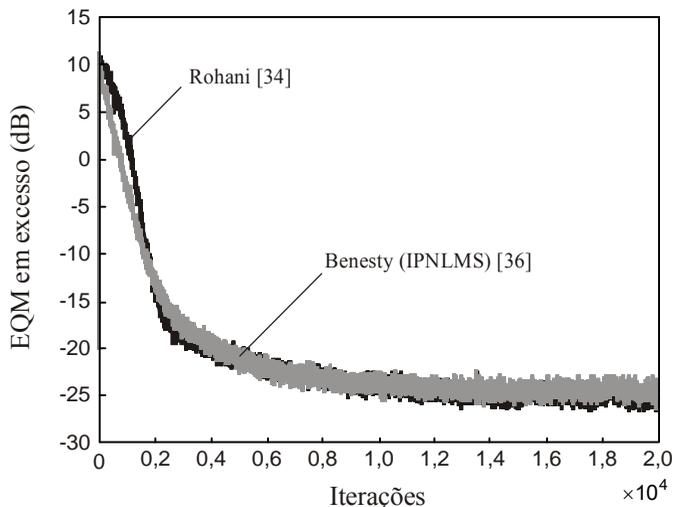
**Figura 5.8.** EQM em excesso dos algoritmos baseados no valor absoluto do erro considerando uma mudança abrupta no sistema a ser identificado.

### 5.3.5 Algoritmos Baseados no Ajuste Proporcional aos Valores Absolutos do Vetor de Coeficientes

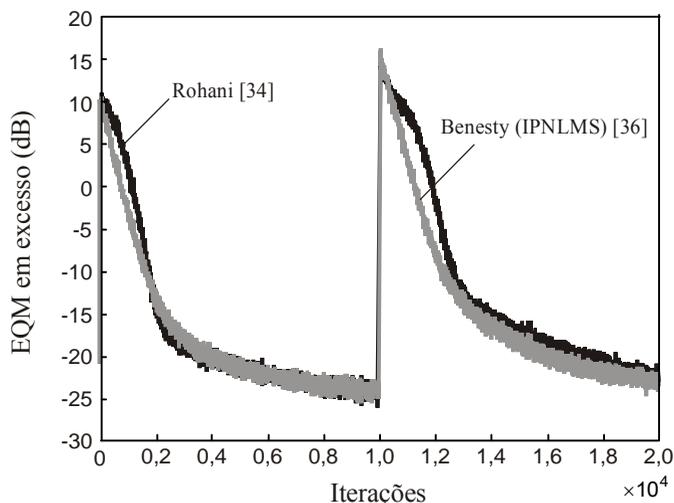
A Figura 5.9 mostra o comportamento do EQM em excesso para os algoritmos baseados no ajuste proporcional aos valores absolutos do vetor de coeficientes. Na Figura 5.10 a evolução do EQM em excesso é mostrada considerando uma mudança abrupta da planta. Assim, no cenário considerado, o algoritmo de Benesty (IPNLMS) [36] apresenta uma convergência pouco mais rápida.

### 5.3.6 Algoritmos Baseados na Normalização do Vetor de Erros, na Teoria do Passo Ótimo e na Otimização com Restrição do Ruído de Medição.

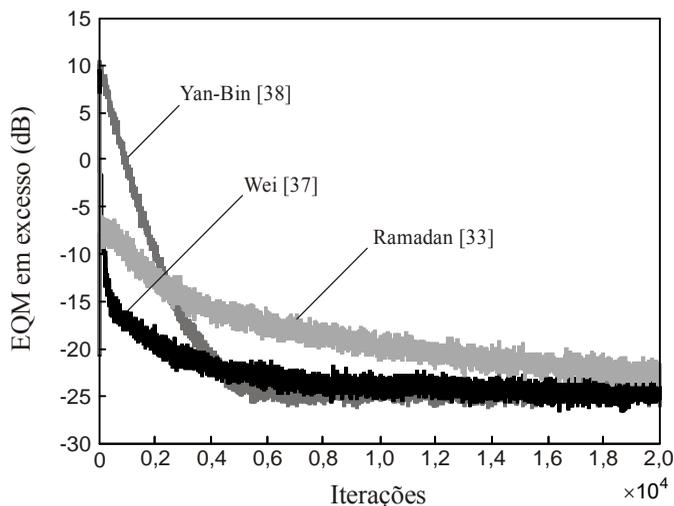
A Figura 5.11 mostra o comportamento do EQM em excesso para os algoritmos baseados na normalização do vetor de erros, na teoria do passo ótimo e na otimização com restrição do ruído de medição. Nessa simulação, o algoritmo de Ramadan [33] converge após a iteração 30000. Na Figura 5.12, a evolução do EQM em excesso é mostrada considerando uma mudança abrupta da planta. Assim, no cenário considerado, o algoritmo de Wei [37] é o que apresenta um melhor desempenho.



**Figura 5.9.** Comportamento do EQM em excesso dos algoritmos baseados no ajuste proporcional ao vetor  $\mathbf{w}(n)$ .



**Figura 5.10.** EQM em excesso dos algoritmos baseados no ajuste proporcional ao vetor  $\mathbf{w}(n)$  considerando uma mudança abrupta na planta.

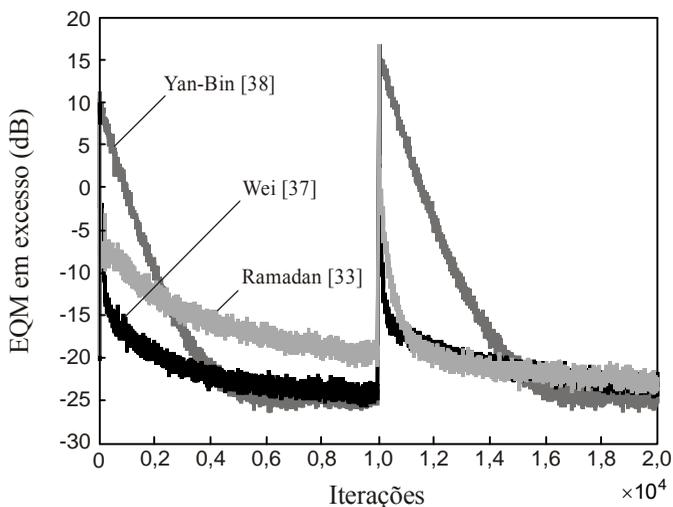


**Figura 5.11.** Comportamento do EQM em excesso dos algoritmos baseados na normalização do vetor de erros, teoria do passo ótimo e otimização com restrição do ruído de medição.

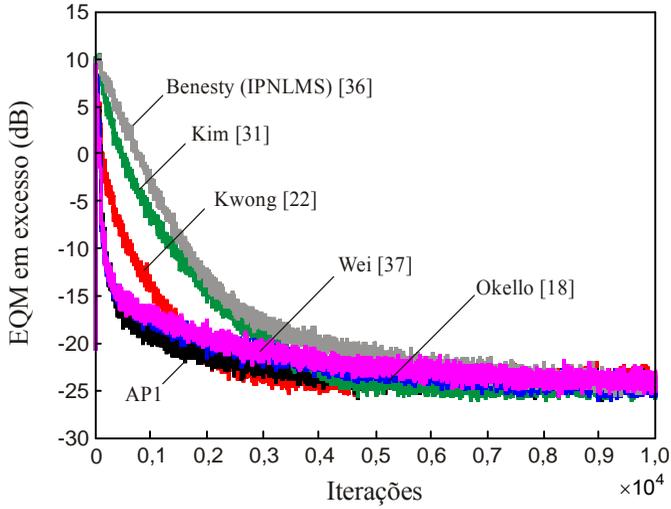
### 5.3.7 Comparação entre os Algoritmos de Melhor Desempenho

A Figura 5.13 apresenta a evolução do EQM em excesso dos algoritmos que apresentaram melhor desempenho nas simulações realizadas nas Seções 5.3.1 a 5.3.6. Na Figura 5.14, a evolução do EQM em excesso desses algoritmos é mostrada considerando uma mudança abrupta da planta.

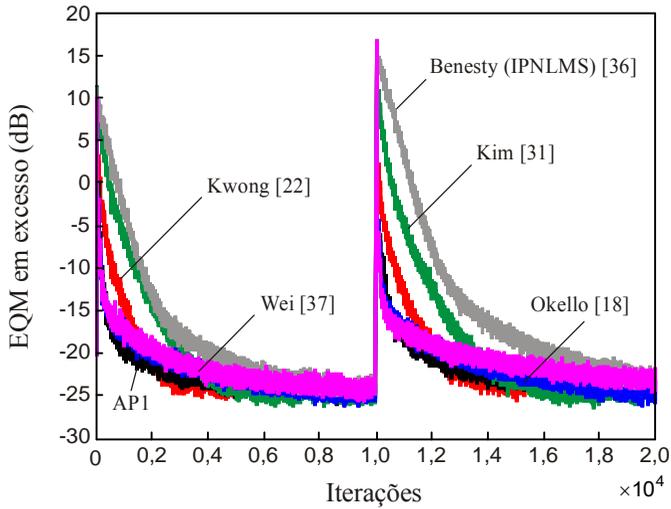
Observa-se das Figuras 5.13 e 5.14 um melhor desempenho dos algoritmos de Okello [18], Wei [37] e do algoritmo API .



**Figura 5.12.** Comportamento do EQM em excesso dos algoritmos baseados na normalização do vetor de erros, teoria do passo ótimo e otimização com restrição do ruído de medição considerando uma mudança abrupta na planta.



**Figura 5.13.** Comparação do EQM em excesso dos algoritmos de melhor desempenho.



**Figura 5.14.** Comparação do EQM em excesso dos algoritmos de melhor desempenho considerando uma mudança abrupta na planta.

## 5.4 Conclusões

Este capítulo apresentou uma breve comparação entre todos os algoritmos VSSLMS considerados neste trabalho de pesquisa. Em relação à complexidade computacional observou-se que os algoritmos propostos AP1 e AP2 apresentaram reduzido número de operações, quando comparados aos seus algoritmos antecessores. Resultados de simulação verificaram o desempenho de todos algoritmos VSSLMS avaliados em um cenário comum de aplicação, utilizando como entrada um sinal correlacionado. Desses resultados destaca-se o desempenho muito bom do algoritmo AP1, embora o algoritmo AP2 tenha a vantagem de não necessitar qualquer ajuste de parâmetros e apresentar melhor imunidade a variações no nível de ruído.

### Conclusões e Comentários Finais

O algoritmo LMS é o mais utilizado em filtragem adaptativa, em virtude da sua simplicidade e robustez. No algoritmo LMS convencional, devido à utilização de um passo de adaptação fixo, não é possível a obtenção simultânea de uma rápida convergência e um reduzido desajuste em regime permanente. Com isso, diversos algoritmos LMS de passo variável (VSSLMS) vêm sendo propostos para contornar tal limitação. Este trabalho trata então do estudo, da modelagem estocástica e de novos algoritmos VSSLMS.

Com o objetivo de fornecer uma base teórica ao estudo dos algoritmos VSSLMS, o algoritmo LMS convencional foi brevemente discutido no Capítulo 2. Sua derivação foi apresentada assim como sua modelagem estocástica desenvolvida à luz da Teoria da Independência. Também, foram abordados aspectos relacionados à estabilidade do referido algoritmo com vistas a escolha do passo de adaptação. Esse capítulo ainda apresentou brevemente o algoritmo LMS normalizado e uma introdução aos algoritmos VSSLMS.

No Capítulo 3, dezoito importantes algoritmos VSSLMS encontrados na literatura técnica foram discutidos, sendo cada um deles brevemente descrito, utilizando para tal uma nomenclatura unificada. Também, foi sugerida uma classificação desses algoritmos segundo o critério utilizado para o ajuste do passo de adaptação, sendo então propostas oito categorias distintas. Tal classificação tem o propósito de facilitar a escolha de um algoritmo VSSLMS para uma determinada aplicação de filtragem adaptativa, assim como apresentar uma compilação dos principais algoritmos da literatura técnica. Dos dezoito algoritmos VSSLMS discutidos no Capítulo 3, somente três possuíam modelos estocásticos recentemente publicados. A modelagem estocástica de algoritmos VSSLMS possibilita a predição do desempenho de tais algoritmos para uma dada condição de operação, permitindo uma melhor compreensão de seu comportamento. O Capítulo 3 apresentou ainda modelos estocásticos de três diferentes algoritmos, considerados de grande importância prática. Resultados de simulação comprovaram a precisão dos modelos desenvolvidos.

Um dos objetivos deste trabalho de pesquisa é o desenvolvimento de novos algoritmos VSSLMS, os quais foram apresentados no

Capítulo 4. Uma modificação em um algoritmo baseado no gradiente do erro quadrático foi proposta, resolvendo com isso um problema elementar de convergência desse algoritmo. Além disso, dois novos algoritmos baseados na autocorrelação do erro foram também apresentados. O primeiro deles, denominado algoritmo AP1, utiliza a função de autocorrelação com  $N$  lags e foi concebido com o objetivo de aumentar a velocidade de convergência na presença de níveis elevados de ruído de medição. Já o segundo algoritmo, denominado AP2, utiliza a autocorrelação do erro e o erro quadrático no cálculo de uma função de controle de passo aplicada a um algoritmo NLMS. Esse esquema manteve a velocidade de convergência do algoritmo agregando ainda uma elevada imunidade a variações no ruído de medição como também robustez a variações na potência no sinal de entrada, além de não ser necessário qualquer ajuste de parâmetros. Com isso, o algoritmo AP2 contorna dois principais problemas envolvidos no projeto de filtros adaptativos, utilizando algoritmos VSSLMS, que são a baixa imunidade a mudanças no nível de ruído além da necessidade freqüente de ajuste de parâmetros. O Capítulo 4 apresentou também uma análise estatística de cada novo algoritmo desenvolvido neste trabalho, originando três novos modelos estocásticos. Considerando as estratégias de ajuste dos passos de adaptação dos algoritmos AP1 e AP2, baseadas na autocorrelação do erro, o desenvolvimento de seus modelos estocásticos exigiram um trabalho matemático intenso, sendo que algumas aproximações foram utilizadas para viabilizar seus desenvolvimentos. Tais aproximações, quando utilizadas, foram devidamente justificadas. Resultados de simulação numérica foram ainda apresentados no Capítulo 4, comprovando tanto a eficácia dos novos algoritmos propostos quando comparados a algoritmos de referência, quanto a precisão dos modelos estocásticos desenvolvidos.

No Capítulo 5, foram apresentados alguns resultados comparativos entre todos os algoritmos VSSLMS estudados e discutidos no Capítulo 3, juntamente com os novos algoritmos propostos no Capítulo 4. Uma análise de complexidade computacional demonstrou que os novos algoritmos propostos apresentam um número menor de operações aritméticas, quando comparados à maioria dos algoritmos avaliados. O Capítulo 5 traz ainda resultados de simulação numérica comparando o desempenho de todos os algoritmos discutidos neste trabalho, considerando um cenário comum de aplicação. Desses resultados, destaca-se o bom desempenho do algoritmo proposto AP1

que apresentou a maior velocidade de convergência dentre todos os algoritmos avaliados.

Como proposta para trabalhos futuros, sugere-se o desenvolvimento de novos algoritmos VSSLMS não-paramétricos para utilização em filtros adaptativos aplicados a sistemas com resposta ao impulso esparsa. A obtenção de novos modelos estocásticos de algoritmos VSSLMS, sem o uso da suposição de independência do passo de adaptação, pode também constituir uma continuação deste trabalho de pesquisa. Trabalhos futuros podem ainda considerar ferramentas de análise não-linear aplicadas aos algoritmos VSSLMS para avaliação de suas características, tais como estabilidade e convergência. Novos trabalhos utilizando projeções em espaços vetoriais podem favorecer a compreensão e o estudo da velocidade de convergência e análise de mínimos locais e/ou globais. Sugere-se ainda a utilização de ferramentas de análise espectral dos algoritmos propostos. Por fim, é sugerida a aplicação prática dos algoritmos aqui desenvolvidos em ambientes reais, com a correspondente análise de comportamento de tais algoritmos frente a problemas, tais como quantização, operação em ponto flutuante ou ponto fixo, questões de implementação em DSP, dentre outros.



## Apêndice A

Determinação de  $E[p^2(n)]$

Escrevendo (3.45) em uma forma fechada, resulta em

$$p(n) = (1 - \beta) \sum_{k=0}^{n-1} \beta^k e(n-k)e(n-k-1). \quad (\text{A.1})$$

Então, elevando ao quadrado ambos os lados de (A.1), obtém-se

$$p^2(n) = (1 - \beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} e(n-k)e(n-k-1)e(n-j)e(n-j-1). \quad (\text{A.2})$$

Usando (2.22), pode-se escrever  $e(n)$  em função de  $\eta(n)$ . Assim,

$$e(n) = \eta(n) - \mathbf{v}^T(n)\mathbf{x}(n) \quad (\text{A.3})$$

e substituindo (A.3) em (A.2), chega-se a

$$\begin{aligned} & p^2(n) \\ &= (1 - \beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ [\eta(n-k) - \mathbf{v}^T(n-k)\mathbf{x}(n-k)] \\ & \quad \times [\eta(n-k-1) - \mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)] \\ & \quad \times [\eta(n-j) - \mathbf{v}^T(n-j)\mathbf{x}(n-j)] [\eta(n-j-1) - \mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \}. \end{aligned} \quad (\text{A.4})$$

Agora, expandindo (A.4), obtém-se

$$\begin{aligned}
p^2(n) &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \\
&\times \{ [\eta(n-k)\eta(n-k-1)\eta(n-j)\eta(n-j-1) \\
&- \eta(n-k)\eta(n-k-1)\eta(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1) \\
&- \eta(n-k)\eta(n-k-1)\eta(n-j-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
&+ \eta(n-k)\eta(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1) \\
&- \eta(n-k)\eta(n-j)\eta(n-j-1)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1) \\
&+ \eta(n-k)\eta(n-j)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1) \\
&+ \eta(n-k)\eta(n-j-1)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
&- \eta(n-k)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1) \\
&\times \mathbf{x}(n-j-1) - \eta(n-k-1)\eta(n-j)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
&+ \eta(n-k-1)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1) \\
&+ \eta(n-k-1)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
&- \eta(n-k-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1) \\
&\times \mathbf{x}(n-j-1) + \eta(n-j)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1) \\
&\times \mathbf{x}(n-k-1) - \eta(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
&\times \mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1) - \eta(n-j-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-k) \\
&\times \mathbf{x}(n-k)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1) + \mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1) \\
&\times \mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1) \}].
\end{aligned} \tag{A.5}$$

Assumindo que  $\eta(n)$ ,  $\mathbf{v}(n)$  e  $\mathbf{x}(n)$  sejam estatisticamente independentes (II) e tomando o valor esperado de ambos os lados de (A.5), tem-se

$$\begin{aligned}
E[p^2(n)] &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \\
&\times \{ [E[\eta(n-k)\eta(n-k-1)\eta(n-j)\eta(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-j-1)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&+ E[\eta(n-k-1)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k-1)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&+ E[\eta(n-j)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)] \\
&+ E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1) \\
&\quad \times \mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \}.
\end{aligned} \tag{A.6}$$

Considerando ainda  $\eta(n)$  ruído branco, (A.6) é reduzida a

$$\begin{aligned}
&E[p^2(n)] \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ [E[\eta(n-k)\eta(n-k-1)\eta(n-j)\eta(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-j-1)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&+ E[\eta(n-k-1)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k-1)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&+ E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1) \\
&\quad \times \mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \}.
\end{aligned} \tag{A.7}$$

Então, fazendo  $E[p^2(n)] = E[p_1(n)] + E[p_2(n)]$ , obtém-se

$$\begin{aligned}
E[p_1(n)] &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \\
&\times \{ [E[\eta(n-k)\eta(n-k-1)\eta(n-j)\eta(n-j-1)]] \\
&+ E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k)\eta(n-j-1)\mathbf{v}^T(n-k-1)\mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&+ E[\eta(n-k-1)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-k-1)\eta(n-j-1)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}
\end{aligned} \tag{A.8}$$

e

$$\begin{aligned}
E[p_2(n)] &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-1) \\
&\times \mathbf{x}(n-k-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)].
\end{aligned} \tag{A.9}$$

Agora, primeiramente calculando  $E[p_1(n)]$  e considerando  $k=j$  em (A.8), tem-se

$$\begin{aligned}
&\{E[p_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{i=0}^{n-1} \beta^{2j} \{ [E[\eta(n-j)\eta(n-j-1)\eta(n-j)\eta(n-j-1)]] \\
&+ E[\eta(n-j)\eta(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
&+ E[\eta(n-j-1)\eta(n-j-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}.
\end{aligned} \tag{A.10}$$

Então, considerando  $\eta(n)$  gaussiano e usando o teorema de fatoração de momentos gaussianos [1], (A.10) pode ser reescrita como

$$\begin{aligned}
&\{E[p_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{i=0}^{n-1} \beta^{2j} \{ [\sigma_\eta^4 + \sigma_\eta^2 E[\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-j-1) \\
&\times \mathbf{x}(n-j-1)] + \sigma_\eta^2 E[\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}.
\end{aligned} \tag{A.11}$$

Como os valores esperados em (A.11) são escalares, pode-se ainda escrever

$$\begin{aligned} \{E[p_1(n)]\}_{k=j} &= (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} \\ &\times \left( \sigma_{\eta}^4 + \sigma_{\eta}^2 E\{\text{tr}[\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)]\} \right. \\ &\left. + \sigma_{\eta}^2 E\{\text{tr}[\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)]\} \right). \end{aligned} \quad (\text{A.12})$$

Então, usando a propriedade  $\text{tr}[\mathbf{AB}] = \text{tr}[\mathbf{BA}]$ , tem-se

$$\begin{aligned} \{E[p_1(n)]\}_{k=j} &= (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} \\ &\times \left( \sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}\{E[\mathbf{x}(n-j-1)\mathbf{x}^T(n-j-1)]E[\mathbf{v}(n-j-1)\mathbf{v}^T(n-j-1)]\} \right. \\ &\left. + \sigma_{\eta}^2 \text{tr}\{E[\mathbf{x}(n-j)\mathbf{x}^T(n-j)]E[\mathbf{v}(n-j)\mathbf{v}^T(n-j)]\} \right). \end{aligned} \quad (\text{A.13})$$

Substituindo  $\mathbf{K}(n) = E[\mathbf{v}(n)\mathbf{v}^T(n)]$  e  $\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$  em (A.13), obtém-se

$$\{E[p_1(n)]\}_{k=j} = (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} \{(\sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}[\mathbf{RK}(n-j-1) + \mathbf{RK}(n)])\}. \quad (\text{A.14})$$

Considerando, então,  $k = j-1$  em (A.8), tem-se

$$\begin{aligned} E[p_1(n)]_{k=j-1} &= (1-\beta)^2 \sum_{j=1}^{n-1} \beta^{2j-1} \\ &\times \{ [E[\eta(n-j+1)\eta(n-j)\eta(n-j)\eta(n-j-1)] \\ &+ E[\eta(n-j+1)\eta(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\ &+ E[\eta(n-j)\eta(n-j)\mathbf{v}^T(n-j+1)\mathbf{x}(n-j+1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\ &+ E[\eta(n-j)\eta(n-j-1)\mathbf{v}^T(n-j+1)\mathbf{x}(n-j+1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}. \end{aligned} \quad (\text{A.15})$$

Aplicando um procedimento similar ao utilizado para obter (A.13), pode-se reescrever (A.15) como

$$\begin{aligned}
& \{E[p_1(n)]\}_{k=j-1} \\
&= (1-\beta)^2 \sum_{j=1}^{n-1} \beta^{2j-1} \left( \sigma_\eta^2 \text{tr}\{E[\mathbf{x}(n-j+1)\mathbf{x}^T(n-j-1)]\} \right. \\
& \quad \left. \times E[\mathbf{v}(n-j-1)\mathbf{v}^T(n-j+1)] \right\}. \quad (\text{A.16})
\end{aligned}$$

Substituindo  $\mathbf{K}_2(n) = E[\mathbf{v}(n)\mathbf{v}^T(n-2)]$  e  $\mathbf{R}_2 = E[\mathbf{x}(n)\mathbf{x}^T(n-2)]$  em (A.16), obtém-se

$$\{E[p_1(n)]\}_{k=j-1} = (1-\beta)^2 \sum_{j=1}^{n-1} \beta^{2j-1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-j+1)]. \quad (\text{A.17})$$

Note que  $\mathbf{R}_2$  é obtido sem invocar a TI, já que conforme tal hipótese  $E[\mathbf{x}(n)\mathbf{x}^T(n-2)] = 0$ .

Finalmente, considerando  $k = j+1$  em (A.8), tem-se

$$\begin{aligned}
& \{E[p_1(n)]\}_{k=j+1} = (1-\beta)^2 \sum_{i=0}^{n-2} \beta^{2j+1} \\
& \times \{ [E[\eta(n-j-1)\eta(n-j-2)\eta(n-j)\eta(n-j-1)]] \\
& + E[\eta(n-j-1)\eta(n-j)\mathbf{v}^T(n-j-2)\mathbf{x}(n-j-2)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
& + E[\eta(n-j-1)\eta(n-j-1)\mathbf{v}^T(n-j-2)\mathbf{x}(n-j-2)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
& + E[\eta(n-j-2)\eta(n-j)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)] \\
& + E[\eta(n-j-2)\eta(n-j-1)\mathbf{v}^T(n-j-1)\mathbf{x}(n-j-1)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}. \quad (\text{A.18})
\end{aligned}$$

Novamente, aplicando um procedimento similar ao utilizado para obter (A.13), pode-se reescrever (A.18) como

$$\begin{aligned}
& \{E[p_1(n)]\}_{k=j+1} \\
&= (1-\beta)^2 \sum_{j=0}^{n-2} \beta^{2j+1} \left( \sigma_\eta^2 \{ \text{tr} E[\mathbf{x}(n-j-2)\mathbf{x}^T(n-j)\mathbf{v}(n-j)\mathbf{v}^T(n-j-2)] \} \right) \\
& \quad (\text{A.19})
\end{aligned}$$

e substituindo  $\mathbf{K}_2(n)$  e  $\mathbf{R}_2$  em (A.19), chega-se a

$$\{E[p_1(n)]\}_{k=j+1} = (1-\beta)^2 \sum_{j=0}^{n-2} \beta^{2j+1} \sigma_{\eta}^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-j)] \quad (\text{A.20})$$

Para as demais combinações de  $j$  e  $k$ , todos os termos de (A.8) são nulos. Então  $E[p_1(n)]$  pode ser obtido de

$$E[p_1(n)] = (1-\beta)^2 \left( \sum_{j=0}^{n-1} \beta^{2j} \{ \sigma_{\eta}^4 + \sigma_{\eta}^2 \text{tr}[\mathbf{R}\mathbf{K}(n-j-1) + \mathbf{R}\mathbf{K}(n)] \} \right. \\ \left. + \sum_{j=1}^{n-1} \beta^{2j-1} \sigma_{\eta}^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-j+1)] + \sum_{j=0}^{n-2} \beta^{2j+1} \sigma_{\eta}^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-j)] \right). \quad (\text{A.21})$$

Pode-se desde já observar a forte dependência de  $E[p_1(n)]$  em relação a variância do ruído aditivo  $\sigma_{\eta}^2$ . Determina-se a seguir  $E[p_2(n)]$ . Aplicando o operador traço de matriz em (A.9) e utilizando a propriedade  $\text{tr}[\mathbf{A}\mathbf{B}] = \text{tr}[\mathbf{B}\mathbf{A}]$ , obtém-se

$$E[p_2(n)] \\ = (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} E\{ \text{tr}[\mathbf{x}(n-j)\mathbf{x}^T(n-j-1)\mathbf{v}(n-j-1)\mathbf{v}^T(n-k) \\ \times \mathbf{x}(n-k)\mathbf{x}^T(n-k-1)\mathbf{v}(n-k-1)\mathbf{v}^T(n-j)] \}. \quad (\text{A.22})$$

Fazendo agora a seguinte aproximação

$$E[p_2(n)] \\ \cong (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{ E[\mathbf{x}(n-j)\mathbf{x}^T(n-j-1)] E[\mathbf{v}(n-j-1)\mathbf{v}^T(n-k)] \\ \times E[\mathbf{x}(n-k)\mathbf{x}^T(n-k-1)] E[\mathbf{v}(n-k-1)\mathbf{v}^T(n-j)] \} \quad (\text{A.23})$$

e substituindo  $\mathbf{R}_1 = E[\mathbf{x}(n)\mathbf{x}^T(n-1)]$  e  $\mathbf{K}_l(n) = E[\mathbf{v}(n)\mathbf{v}^T(n-l)]$  em (A.23), chega-se a

$$\begin{aligned}
& E[p_2(n)] \\
& \cong (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{\mathbf{R}_1 \mathbf{K}_{k-j-1}(n-j-1) \mathbf{R}_1 \mathbf{K}_{j-k-1}(n-k-1)\}. \quad (\text{A.24})
\end{aligned}$$

Então,  $E[p^2(n)]$  pode ser completamente determinado pela seguinte expressão:

$$\begin{aligned}
& E[p^2(n)] = E[p_1(n)] + E[p_2(n)] \\
& = (1-\beta)^2 \left( \sum_{j=0}^{n-1} \beta^{2j} \{\sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R} \mathbf{K}(n-j-1) + \mathbf{R} \mathbf{K}(n)]\} \right. \\
& \quad + \sum_{j=1}^{n-1} \beta^{2j-1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-j+1)] + \sum_{j=0}^{n-2} \beta^{2j+1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-j)] \quad (\text{A.25}) \\
& \quad \left. + \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{\mathbf{R}_1 \mathbf{K}_{k-j-1}(n-j-1) \mathbf{R}_1 \mathbf{K}_{j-k-1}(n-k-1)\} \right).
\end{aligned}$$

Como os elementos de  $\mathbf{K}_{k-j-1}$  e  $\mathbf{K}_{j-k-1}$  se tornam muito pequenos à medida que  $|j-k|$  aumenta, assume-se que  $E[p_1(n)] \gg E[p_2(n)]$  e  $E[p^2(n)] \cong E[p_1(n)]$ . Com isso a expressão para  $E[p^2(n)]$  pode ser aproximada por

$$\begin{aligned}
& E[p^2(n)] \\
& \cong (1-\beta)^2 \left( \sum_{j=0}^{n-1} \beta^{2j} \{\sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R} \mathbf{K}(n-j-1) + \mathbf{R} \mathbf{K}(n)]\} \right. \\
& \quad \left. + \sum_{j=1}^{n-1} \beta^{2j-1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2 \mathbf{K}_2^T(n-j+1)] + \sum_{j=0}^{n-2} \beta^{2j+1} \sigma_\eta^2 \text{tr}[\mathbf{R}_2^T \mathbf{K}_2(n-j)] \right). \quad (\text{A.26})
\end{aligned}$$

## Apêndice B

### Determinação de $E[p(n)]$

Escrevendo (4.10) em uma forma fechada, resulta em

$$p(n) = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N [e(n-k)e(n-k-i)]^2 \quad (\text{B.1})$$

ou ainda

$$p(n) = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N [e^2(n-k)e^2(n-k-i)]. \quad (\text{B.2})$$

Agora, substituindo (A.3) em (B.2), obtém-se

$$p(n) = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N \{[\eta(n-k) - \mathbf{v}^T(n-k)\mathbf{x}(n-k)]^2 \times [\eta(n-k-i) - \mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)]^2\}. \quad (\text{B.3})$$

Expandindo então (B.3), resulta em

$$\begin{aligned} p(n) = & (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N [\eta(n-k)\eta(n-k)\eta(n-k-i)\eta(n-k-i) \\ & - \eta(n-k)\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & - \eta(n-k)\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & + \eta(n-k)\eta(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & - \eta(n-k)\eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\ & + \eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & + \eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & - \eta(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i) \\ & \times \mathbf{x}(n-k-i) - \eta(n-k)\eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\ & + \eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & + \eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i) \\ & - \eta(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i) \end{aligned}$$

$$\begin{aligned}
& \times \mathbf{x}(n-k-i) + \eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k) \\
& \times \mathbf{x}(n-k) - \eta(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
& \times \mathbf{v}^T(n-k)\mathbf{x}(n-k) - \eta(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k) \\
& \times \mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) + \mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
& \times \mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)].
\end{aligned} \tag{B.4}$$

Assumindo novamente  $\eta(n)$ ,  $\mathbf{v}(n)$  e  $\mathbf{x}(n)$  estatisticamente independentes e tomando o valor esperado de (B.4), chega-se a

$$\begin{aligned}
E[p(n)] &= (1-\beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N \{E[\eta(n-k)\eta(n-k)\eta(n-k-i)\eta(n-k-i)] \\
& + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + 2E[\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + 2E[\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + E[\eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)] \\
& + E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \times \\
& \quad \mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)]\}.
\end{aligned} \tag{B.5}$$

Tomando agora  $E[p(n)] = E[p_1(n)] + E[p_2(n)]$ , obtém-se

$$\begin{aligned}
E[p_1(n)] &= (1-\beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N \{E[\eta(n-k)\eta(n-k)\eta(n-k-i)\eta(n-k-i)] \\
& + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + 2E[\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + 2E[\eta(n-k)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
& + E[\eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)]\}
\end{aligned} \tag{B.6}$$

e

$$E[p_2(n)] = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \times \mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)]. \quad (\text{B.7})$$

Primeiramente determina-se  $E[p_1(n)]$ . Assumindo  $\eta(n)$  ruído branco em (B.6), resulta em

$$E[p_1(n)] = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N \{ [E[\eta(n-k)\eta(n-k)\eta(n-k-i)\eta(n-k-i)] + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] + E[\eta(n-k-i)\eta(n-k-i)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)] \}. \quad (\text{B.8})$$

Agora, aplicando o teorema de fatoração de momentos gaussianos[1] e levando em conta que os valores esperados em (B.8) são escalares, tem-se

$$E[p_1(n)] = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N (\sigma_\eta^4 + \sigma_\eta^2 \{ E[\text{tr}\{\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i) \times \mathbf{x}(n-k-i)\}] + E[\text{tr}\{\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\}] \}). \quad (\text{B.9})$$

Aplicando agora a propriedade  $\text{tr}[\mathbf{AB}] = \text{tr}[\mathbf{BA}]$ ,

$$E[p_1(n)] = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N (\sigma_\eta^4 + \sigma_\eta^2 \{ E[\text{tr}\{\mathbf{x}(n-k-i)\mathbf{x}^T(n-k-i)\mathbf{v}(n-k-i)\mathbf{v}^T(n-k-i)\}] + E[\text{tr}\{\mathbf{x}(n-k)\mathbf{x}^T(n-k)\mathbf{v}(n-k)\mathbf{v}^T(n-k)\}] \}) \quad (\text{B.10})$$

e substituindo  $\mathbf{K}(n) = E[\mathbf{v}(n)\mathbf{v}^T(n)]$  e  $\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$  em (B.10), chega-se a

$$E[p_1(n)] = (1 - \beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N (\sigma_\eta^4 + \sigma_\eta^2 \{ \text{tr}[\mathbf{RK}(n-k-i) + \mathbf{RK}(n-k)] \}). \quad (\text{B.11})$$

Determina-se agora  $E[p_2(n)]$ , utilizando o mesmo procedimento usado para o cálculo de  $E[p_1(n)]$ . Assim,

$$\begin{aligned}
& E[p_2(n)] \\
&= \sum_{k=0}^{n-1} \beta^k \sum_{i=0}^N E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
&\quad \times \mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)\mathbf{v}^T(n-k-i)\mathbf{x}(n-k-i)] \\
&= \sum_{k=0}^{n-1} \beta^k \sum_{i=0}^N E\{\text{tr}[\mathbf{x}(n-k-i)\mathbf{x}^T(n-k-i)\mathbf{v}(n-k-i)\mathbf{v}^T(n-k) \\
&\quad \times \mathbf{x}(n-k)\mathbf{x}^T(n-k)\mathbf{v}(n-k)\mathbf{v}^T(n-k-i)]\} \\
&\cong \sum_{k=0}^{n-1} \beta^k \sum_{i=0}^N \text{tr}\{E[\mathbf{x}(n-k-i)\mathbf{x}^T(n-k-i)]E[\mathbf{v}(n-k-i)\mathbf{v}^T(n-k)] \\
&\quad \times E[\mathbf{x}(n-k)\mathbf{x}^T(n-k)]E[\mathbf{v}(n-k)\mathbf{v}^T(n-k-i)]\} \\
&= \sum_{k=0}^{n-1} \beta^k \sum_{i=0}^N \text{tr}\{\mathbf{R}E[\mathbf{v}(n-k-i)\mathbf{v}^T(n-k)]\mathbf{R}E[\mathbf{v}(n-k)\mathbf{v}^T(n-k-i)]\} \\
&= \sum_{k=0}^{n-1} \beta^k \sum_{i=0}^N \text{tr}\{\mathbf{R}\mathbf{K}_i^T(n-k)\mathbf{R}\mathbf{K}_i(n-k)\}.
\end{aligned} \tag{B.12}$$

Então,  $E[p(n)]$  pode ser determinado por

$$\begin{aligned}
E[p(n)] &= E[p_1(n)] + E[p_2(n)] \\
&= (1-\beta) \sum_{k=0}^{n-1} \beta^k \sum_{i=1}^N (\sigma_\eta^4 + \sigma_\eta^2 \{\text{tr}[\mathbf{R}\mathbf{K}(n-k-i) + \mathbf{R}\mathbf{K}(n-k)]\} \\
&\quad + \text{tr}\{\mathbf{R}\mathbf{K}_i^T(n-k)\mathbf{R}\mathbf{K}_i(n-k)\}).
\end{aligned} \tag{B.13}$$

## Apêndice C

### Determinação de $E[q^2(n)]$

Escrevendo (4.30) como segue:

$$q(n) = (1 - \beta) \sum_{k=0}^{n-1} \beta^k e^2(n-k) \quad (\text{C.1})$$

e elevando ao quadrado ambos os lados de (C.1), tem-se

$$q^2(n) = (1 - \beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} e^2(n-k) e^2(n-j). \quad (\text{C.2})$$

Agora, substituindo (A.3) em (C.2), resulta em

$$\begin{aligned} q^2(n) = (1 - \beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ & [\eta(n-k) - \mathbf{v}^T(n-k)\mathbf{x}(n-k)]^2 [\eta(n-j) \\ & - \mathbf{v}^T(n-j)\mathbf{x}(n-j)]^2 \}. \end{aligned} \quad (\text{C.3})$$

Então, expandindo (C.3), obtém-se

$$\begin{aligned} q^2(n) = (1 - \beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ & [\eta(n-k)\eta(n-k)\eta(n-j)\eta(n-j) \\ & - \eta(n-k)\eta(n-k)\eta(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j) - \eta(n-k)\eta(n-k)\eta(n-j) \\ & \times \mathbf{v}^T(n-j)\mathbf{x}(n-j) + \eta(n-k)\eta(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j) \\ & \times \mathbf{x}(n-j) - \eta(n-k)\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\ & + \eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\ & + \eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\ & - \eta(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\ & - \eta(n-k)\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \end{aligned}$$

$$\begin{aligned}
& +\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
& +\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
& -\eta(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
& +\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
& -\eta(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
& -\eta(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
& +\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j) \\
& \times \mathbf{x}(n-j)\}.
\end{aligned} \tag{C.4}$$

Assumindo, novamente,  $\eta(n)$ ,  $\mathbf{v}(n)$  e  $\mathbf{x}(n)$  estatisticamente independentes e tomando o valor esperado em ambos lados de (C.4), tem-se

$$\begin{aligned}
E[q^2(n)] &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \\
& \times \{ [E[\eta(n-k)\eta(n-k)\eta(n-j)\eta(n-j)]] \\
& + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
& + 2E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \quad (C.5) \\
& + 2E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
& + E[\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)] \\
& + E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j) \\
& \times \mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}.
\end{aligned}$$

Fazendo agora  $E[q^2(n)] = E[q_1(n)] + E[q_2(n)]$ , obtêm-se

$$\begin{aligned}
& E[q_1(n)] \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ [E[\eta(n-k)\eta(n-k)\eta(n-j)\eta(n-j)] \\
&\quad + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \quad (C.6) \\
&\quad + 2E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&\quad + 2E[\eta(n-k)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
&\quad + E[\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)] \}
\end{aligned}$$

e

$$\begin{aligned}
E[q_2(n)] &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} E[\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k) \\
&\quad \times \mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)]. \quad (C.7)
\end{aligned}$$

Agora, primeiramente calculando  $E[q_1(n)]$  e considerando  $k = j$  em (C.6), tem-se

$$\begin{aligned}
& \{E[q_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} \{ E[\eta(n-j)^4] \quad (C.8) \\
&\quad + 6E[\eta(n-j)^2 \mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \}.
\end{aligned}$$

Considerando agora o teorema de fatoração de momentos gaussianos [1], (C.8) pode ser reescrita como

$$\begin{aligned}
& \{E[q_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{i=0}^{n-1} \beta^{2j} \{ [3\sigma_\eta^4 + 6\sigma_\eta^2 E[\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)]] \}. \quad (C.9)
\end{aligned}$$

Como os valores esperados em (C.9) são escalares, pode-se escrever

$$\begin{aligned}
& \{E[q_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} (3\sigma_\eta^4 + 6\sigma_\eta^2 E\{\text{tr}[\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)]\}).
\end{aligned} \tag{C.10}$$

Utilizando novamente a propriedade  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ , tem-se

$$\begin{aligned}
& \{E[q_1(n)]\}_{k=j} \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} ([3\sigma_\eta^4 + 6\sigma_\eta^2 E\{\text{tr}[\mathbf{x}(n-j)\mathbf{x}^T(n-j)]\} \\
&\quad \times E\{\text{tr}[\mathbf{v}(n-j)\mathbf{v}^T(n-j)]\}).
\end{aligned} \tag{C.11}$$

Agora, calculando o valor esperado de (C.11), obtém-se

$$\{E[q_1(n)]\}_{k=j} = (1-\beta)^2 \sum_{j=0}^{n-1} \beta^{2j} \{3\sigma_\eta^4 + 6\sigma_\eta^2 \text{tr}[\mathbf{RK}(n-j)]\} \tag{C.12}$$

Considerando agora  $j \neq k$  em (C.6), tem-se

$$\begin{aligned}
& \{E[q_1(n)]\}_{j \neq k} = (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \\
& \times \{E[\eta(n-k)\eta(n-k)\eta(n-j)\eta(n-j)] \\
& + E[\eta(n-k)\eta(n-k)\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j)\mathbf{x}(n-j)] \\
& + E[\eta(n-j)\eta(n-j)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)]\}
\end{aligned} \tag{C.13}$$

Aplicando novamente o teorema de fatoração de momentos gaussianos [1] e usando o operador traço, (C.13) pode ser reescrita como

$$\begin{aligned}
& \{E[q_1(n)]\}_{j \neq k} \\
&= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} (\sigma_\eta^4 + \sigma_\eta^2 \{E[\text{tr}\{\mathbf{v}^T(n-j)\mathbf{x}(n-j)\mathbf{v}^T(n-j) \\
&\quad \times \mathbf{x}(n-j)\}] + E[\text{tr}\{\mathbf{v}^T(n-k)\mathbf{x}(n-k)\mathbf{v}^T(n-k)\mathbf{x}(n-k)\}]\}).
\end{aligned} \tag{C.14}$$

Aplicando, novamente, a propriedade  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ , obtém-se

$$\begin{aligned}
 & \{E[q_1(n)]\}_{j \neq k} \\
 &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} (\sigma_\eta^4 + \sigma_\eta^2 \{E[\text{tr}\{\mathbf{x}(n-j)\mathbf{x}^\text{T}(n-j)\mathbf{v}(n-j) \\
 & \quad \times \mathbf{v}^\text{T}(n-j)\}\} + E[\text{tr}\{\mathbf{x}(n-k)\mathbf{x}^\text{T}(n-k)\mathbf{v}(n-k)\mathbf{v}^\text{T}(n-k)\}\}])
 \end{aligned} \tag{C.15}$$

resultando em

$$\begin{aligned}
 & \{E[q_1(n)]\}_{j \neq k} \\
 &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} (\sigma_\eta^4 + \sigma_\eta^2 \{\text{tr}[\mathbf{R}\mathbf{K}(n-j) + \mathbf{R}\mathbf{K}(n-k)]\}).
 \end{aligned} \tag{C.16}$$

Assim,  $E[q_1(n)]$  pode ser obtido de

$$\begin{aligned}
 E[q_1(n)] &= (1-\beta)^2 \left( \sum_{j=0}^{n-1} \beta^{2j} \{3\sigma_\eta^4 + 6\sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-j)]\} \right. \\
 & \quad \left. + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ \sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-j) + \mathbf{R}\mathbf{K}(n-k)] \} \right).
 \end{aligned} \tag{C.17}$$

Determina-se a seguir  $E[q_2(n)]$ . Aplicando o operador traço de matriz em (C.7) e usando a propriedade  $\text{tr}(\mathbf{A}\mathbf{B}) = \text{tr}(\mathbf{B}\mathbf{A})$ , tem-se

$$\begin{aligned}
 & E[q_2(n)] \\
 &= (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} E\{\text{tr}[\mathbf{x}(n-j)\mathbf{x}^\text{T}(n-j)\mathbf{v}(n-j)\mathbf{v}^\text{T}(n-k) \\
 & \quad \times \mathbf{x}(n-k)\mathbf{x}^\text{T}(n-k)\mathbf{v}(n-k)\mathbf{v}^\text{T}(n-j)]\}.
 \end{aligned} \tag{C.18}$$

Considerando agora a seguinte aproximação:

$$\begin{aligned}
 & E[q_2(n)] \\
 &\cong (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{E[\mathbf{x}(n-j)\mathbf{x}^\text{T}(n-j)]E[\mathbf{v}(n-j)\mathbf{v}^\text{T}(n-k)] \\
 & \quad \times E[\mathbf{x}(n-k)\mathbf{x}^\text{T}(n-k)]E[\mathbf{v}(n-k)\mathbf{v}^\text{T}(n-j)]\}
 \end{aligned} \tag{C.19}$$

e substituindo  $\mathbf{R}$  e  $\mathbf{K}_l(n)$  em (C.19), tem-se

$$E[q_2(n)] \cong (1-\beta)^2 \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{\mathbf{R}\mathbf{K}_{k-j}(n-j)]\mathbf{R}\mathbf{K}_{j-k}(n-k)\}. \quad (\text{C.20})$$

Então,  $E[q^2(n)]$  pode ser obtido por

$$\begin{aligned} E[q^2(n)] &= E[q_1(n)] + E[q_2(n)] \\ &= (1-\beta)^2 \left( \sum_{i=0}^{n-1} \beta^{2i} \{3\sigma_\eta^4 + 6\sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-i)]\} \right. \\ &\quad + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ \sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-j) + \mathbf{R}\mathbf{K}(n-k)] \} \quad (\text{C.21}) \\ &\quad \left. + \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \text{tr}\{\mathbf{R}\mathbf{K}_{k-j}(n-j)]\mathbf{R}\mathbf{K}_{j-k}(n-k)\} \right) \end{aligned}$$

Como os elementos de  $\mathbf{K}_{k-j}$  e  $\mathbf{K}_{j-k}$  se tornam muito pequenos à medida que  $|j-k|$  aumenta, assume-se que  $E[q_1(n)] \gg E[q_2(n)]$  e  $E[q^2(n)] \cong E[q_1(n)]$ . Com isso a expressão para  $E[q^2(n)]$  pode ser aproximada por

$$\begin{aligned} E[q^2(n)] &\approx (1-\beta)^2 \left( \sum_{i=0}^{n-1} \beta^{2i} \{3\sigma_\eta^4 + 6\sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-i)]\} \right. \\ &\quad \left. + \sum_{\substack{j=0 \\ j \neq k}}^{n-1} \sum_{k=0}^{n-1} \beta^{j+k} \{ \sigma_\eta^4 + \sigma_\eta^2 \text{tr}[\mathbf{R}\mathbf{K}(n-j) + \mathbf{R}\mathbf{K}(n-k)] \} \right). \quad (\text{C.22}) \end{aligned}$$

## Referências Bibliográficas

- [1] S. Haykin, *Adaptive Filter Theory*, 4<sup>th</sup> ed. Upper Saddle River, NJ: Prentice Hall, 2002.
- [2] J. V. Lopez, J. C. Sanchez, H. P. Meana, "Adaptive echo canceller using a modified LMS algorithm," in *Proc. 2nd Int. Conf. on Electrical and Electronics Engineering (ICEEE)*, Mexico City, Mexico, Sep. 2005, pp. 93-96.
- [3] B. Farhang-Boroujeny, "Channel equalization via channel identification: algorithms and simulation for rapidly fading HF channels," *IEEE Trans. Communications*, vol. 44, no. 11, pp. 1409-1412, Nov. 1996.
- [4] S. M. Kuo and D. R. Morgan, "Active noise control: a tutorial review," *Proc. of the IEEE*, vol. 87, no. 6, pp. 943-973, Jun. 1999.
- [5] M. Mboup, M. Bonnet, and N. Bershada, "LMS coupled adaptive prediction and system identification: a statistical model and transient mean analysis," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2607-2615, Oct. 1994.
- [6] B. Widrow and M. Hoff, "Adaptive switching circuits," in *Proc. IRE Western Electronic Show and Convention*, New York, USA, Part.4, Aug. 1960, pp. 96-104.
- [7] AL. M. Nelson, P. Axelrad, and D. M. Etter, "Adaptive detection of code delay and multipath in a simplified gps signal model," *Proc. 10th Int. Tec. Meet. Sat. Div. of the Instit. of Navigat.(ION GPS)*, Kansas City, USA, vol. 1, Sep. 1997, pp. 569-580.
- [8] H. Yang and Wen-Rong Wu, "A novel adaptive code acquisition using antenna array for DS/CDMA systems," in *Proc. of IEEE Int. Work. Small Antennas and Novel Metamaterials (IWAT)*, Singapore, Mar. 2005, pp. 458-461.
- [9] M. Shafiq, "Internal model control structure using adaptive inverse control strategy," *ISA Trans.*, vol. 44, no. 3, pp. 353-362, Jul. 2005.

- [10] J. Fei, G. Song, “Adaptive feedforward model reference control scheme for vibration suppression,” *Proc. IEEE Int. Symp. Intelligent Control*, Taipei, Taiwan, Sep. 2004, pp. 357-362.
- [11] A. H. Sayed, *Fundamentals of Adaptive Filtering*, Hoboken, NJ: J. Wiley & Sons: IEEE Press, 2003.
- [12] R. W. Harris, D. M. Chabries, and F. A. Bishop, “A variable step (VS) adaptive filter,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 2, pp. 309-316, Apr. 1986.
- [13] J. C. Richards, M. A. Webster, and J. C. Principe, “A gradient-based variable step-size LMS algorithm,” in *Proc. IEEE Southeastcon*, Williamsburg, USA, vol. 2, Apr. 1991, pp. 1083-1087.
- [14] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step size,” *IEEE Trans. Signal Process.*, vol. 41, no. 6, pp. 2075-2087, Jun. 1993.
- [15] A. I. Sulyman and A. Zerguine, “Convergence and steady state analysis of a variable step-size normalized LMS algorithm,” *Proc. IEEE Int. Symp. Signal Processing and Its Applications (ISSPA)*, Paris, France, vol. 2, Jul. 2003, pp. 591-594.
- [16] B. Farhang-Boroujeny, “Variable step size LMS algorithm – New developments and experiments,” *IEE Proceedings – Vision, Image, Signal Process.*, vol. 141, no. 5, pp. 311-317, Oct. 1994.
- [17] T. J. Shan and T. Kailath, “Adaptive algorithms with an automatic gain control feature,” *IEEE Trans. Circuits Syst.*, vol. CAS-35, no. 1, pp. 122-127, Jan. 1988.
- [18] J. Okello, Y. Itoh, Y. Fukui, I. Nakanishi, and M. Kobayashi, “A new modified variable step size for the LMS algorithm,” in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, Monterey, USA, vol. 5, Jun. 1998, pp. 170-173.
- [19] W. P. Ang and B. Farhang-Boroujeny, “A new class of gradient adaptive step-size LMS algorithms,” *IEEE Trans. Signal Process.*, vol. 49, no. 4, pp. 805-810, Apr. 2001.
- [20] H. C. Shin, A. H. Sayed and W. J. Song, “Variable step-size NLMS and affine projection algorithms,” *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 132-135, Feb. 2004.

- [21] Yun-Long Bi, Feng-Chang Lai, Yi-Zheng Ye, “Gradient vector driven variable step size least mean square algorithm”, in *Proc. 2nd International Asia Conference on Informatics in Control*, Mar. 2010.
- [22] R. H. Kwong and E. W. Johnston, “A variable step size LMS algorithm,” *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1633-1642, Jul. 1992.
- [23] I. Nakanishi and Y. Fukui, “A new adaptive convergence factor algorithm with the constant damping parameter,” *IEICE Trans. Fundamentals*, vol. E78-A, no. 6, pp. 649-655, Jun. 1995.
- [24] M. H. Costa and J. C. M. Bermudez, “A robust variable step size algorithm for LMS adaptive filters,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toulouse, France, vol. 3, May 2006, pp. 93-96.
- [25] J. Benesty, H. Rey, L. R. Vega, and S. Tressens, “A nonparametric VSS NLMS algorithm,” *IEEE Signal Processing Letters*, vol. 13, no. 10, pp. 581-584, Oct. 2006.
- [26] Jeng-Kuang Hwang and Yuan-Ping Li. “Variable step-size LMS algorithm with a gradient-based weighted average,” *IEEE Signal Processing Letters*, vol. 16, no. 12, pp. 1043-1046, Dec. 2009.
- [27] H. Takekawa, T. Shimamura, and S. A. Jimaa, “An efficient and effective variable step-size NLMS algorithm”, in *Proc. 42nd ASILOMAR Conference on Signals, Systems and Computers*, Pacific Grove, USA, Oct. 2008.
- [28] T. Aboulnasr and K. Mayyas, “A robust variable step-size LMS-type algorithm: analysis and simulations,” *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 631-639, Mar. 1997.
- [29] J. G. F. Zipf, O. J. Tobias, R. Seara, “A new VSSLMS algorithm based on error autocorrelation,” in *Proc. Europ. Signal Process. Conf. (EUSIPCO)*, Lausanne, Switzerland, Aug. 2008, pp. 1-5.
- [30] J. G. F. Zipf, O. J. Tobias, R. Seara, “Non-parametric VSS-NLMS algorithm with control parameter based on the error correlation,” in *Proc. 7th International Telecommunications Symposium (ITS 2010)*, Manaus, AM, Brazil, Sep. 2010, pp. 1-6.

- [31] D. W. Kim, J. H. Hoi, Y. S. Choi, C. H. Jeon, and H. Y. Ko, "A VS-LMS algorithm using normalized absolute estimation error," in *Proc. IEEE Digital Signal Processing Applications (TENCON)*, Perth, Australia, vol. 2, Nov. 1996, pp. 692-697.
- [32] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Process. Lett.*, vol. 5, no. 5, pp. 111-114, May 1998.
- [33] Z. Ramadan and A. Poularikas, "A robust variable step-size LMS algorithm using error-data normalization," in *Proc. IEEE Southeastcon*, Huntsville, USA, Apr. 2005, pp. 219-224.
- [34] B. Rohani and K. S. Chung, "A modified LMS algorithm with improved convergence," in *Proc. IEEE Singapore Int. Conf. Communication Systems*, Singapore, Nov. 1994, pp. 845-849.
- [35] D. L. Duttweiler, "Proportionate normalized LMS adaptation in echo cancellers," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508-518, Sep. 2000.
- [36] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Orlando, USA, May 2002, pp. 1881-1884.
- [37] Y. Wei and S. B. Gelfand, "Noise-constrained least mean squares algorithm," *IEEE Trans. Signal Process.*, vol. 49, no. 9, pp. 1961-1970, Sep. 2001.
- [38] Q. Yan-bin, M. Fan-gang, and G. Lei, "A new variable step size LMS adaptive filtering algorithm," in *Proc. IEEE Int. Symp. Industrial Electronics*, Vigo, Spain, Jun. 2007, pp. 1601-1605.
- [39] L. R. Vega, H. Rey and J. Benesty, "A New Robust Variable Step-Size NLMS Algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1878-1893, May 2008.
- [40] Li Xiao-bo, Fan Yang-yu and Peng Ke, "A variable step-size LMS adaptive filtering algorithm," in *Proc. 5th Int. Conf. Wireless Communications, Networking and Mobile Computing*, Beijing, China, Sep. 2009, pp. 2283-2286.

- [41] J. G. F. Zipf, O. J. Tobias, R. Seara, “Estudo comparativo e classificação de algoritmos LMS de passo variável,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- [42] J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo estocástico para o algoritmo de Shan de passo variável modificado,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- [43] J. G. F. Zipf, O. J. Tobias, R. Seara, “New insights on the noise constrained LMS algorithm,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Hawaii, USA, Apr. 2007, pp. 1365-1368.
- [44] J. G. F. Zipf, O. J. Tobias, R. Seara, “Estudo e modelo estocástico do algoritmo LMS restrito ao ruído de medição,” in *Anais do XXV Simpósio Brasileiro de Telecomunicações (SBrT)*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- [45] J. G. F. Zipf, O. J. Tobias, R. Seara, “Modelo estocástico do algoritmo LMS de passo variável baseado no gradiente do sinal de erro quadrático,” in *Anais do XXVI Simpósio Brasileiro de Telecomunicações (SBrT)*, Rio de Janeiro, RJ, Brasil, Set. 2008, pp. 1-6.
- [46] J. E. Kolodziej, O. J. Tobias, R. Seara, “An improved stochastic model of the NLMS algorithm for correlated input data,” in *Proc. Europ. Signal Process. Conf. (EUSIPCO)*, Poznan, Poland, Sep. 2007, pp. 360-364.
- [47] S. B. Gelfand, Y. Wei, and J. V. Krogmeier, “The stability of variable step-size LMS algorithms,” *IEEE Trans. Signal Process.*, vol. 47, no. 12, pp. 3277-3288, Dec. 1999.
- [48] H. Robbins and S. Monro, “A stochastic approximation method,” *Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400-407, Sep. 1951.