

JAVIER ERNESTO KOLODZIEJ

**ALGORITMOS ADAPTATIVOS COM SINAL DE ENTRADA
NORMALIZADO: MODELAGEM ESTATÍSTICA E
APRIMORAMENTOS**

**FLORIANÓPOLIS
2010**

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

ALGORITMOS ADAPTATIVOS COM SINAL DE ENTRADA
NORMALIZADO: MODELAGEM ESTATÍSTICA E
APRIMORAMENTOS

Tese submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a obtenção do Grau de Doutor
em Engenharia Elétrica.

JAVIER ERNESTO KOLODZIEJ

Florianópolis, Agosto de 2010

**ALGORITMOS ADAPTATIVOS COM SINAL DE ENTRADA
NORMALIZADO: MODELAGEM ESTATÍSTICA E
APRIMORAMENTOS**

Javier Ernesto Kolodziej

‘Esta Tese foi julgada adequada para obtenção do Título de Doutor em Engenharia Elétrica, 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. Roberto de Souza Salgado, Dr.
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Prof. Rui Seara, Dr. - Presidente

Prof. João Marcos Travassos Romano, Dr.

Prof. Luís Geraldo Meloni, Dr.

Prof^a. Elen Macedo Lobato, Dra.

Prof. Hans Helmut Zürn, Ph.D.

Prof. Walter Pereira Carpes Junior., Dr.

Florianópolis, Agosto de 2010

*A Orlando José Tobias e Pedro Ricardo Santander, in
memoriam. Este trabalho é parte de seus legados.*

AGRADECIMENTOS

A Deus, pelo dom da vida, da sabedoria e do discernimento.

Aos meus pais, Carlos e Laura, por me darem a vida, pelo amor incondicional e pelos ensinamentos.

A meu grande amor, Nancy, pela sua carinhosa presença nesse e todos os momentos importantes da minha vida.

Ao Prof. Dr. Rui Seara pela disposição em compartilhar sua sabedoria e principalmente pela paciente orientação nestes anos de trabalho.

A meu querido co-orientador e amigo Orlando José Tobias. Sua competência, perseverança e entusiasmo continuarão sendo meus exemplos.

Aos participantes da banca examinadora: Prof. Dr. João Marcos Travassos; Prof. Dr. Luís Geraldo Meloni; Profa. Dra. Elen Macedo Lobato; Prof. Dr. Hans Helmut Zürn e Prof. Dr. Walter Pereira Carpes Junior, pelas pertinentes contribuições sugeridas ao trabalho.

Aos meus colegas e amigos da Facultad de Ingeniería da Universidad Nacional de Misiones, em especial a Pedro Ricardo Santander, pelo incentivo e motivação.

Ao Programa de Estudante-Convênio de Pós-Graduação (PEC-PG) e à Universidad Nacional de Misiones, pelo apoio financeiro, fundamental na realização deste trabalho.

A todos os colegas do LINSE, em especial ao Elton pela presteza na diagramação dos artigos e da presente tese.

A todos os meus amigos que transformaram as adversidades, próprias da fase de doutorado, em momentos saudosamente prazerosos da minha ótima estada em Florianópolis.

Resumo de Tese apresentada à UFSC como parte dos requisitos para a obtenção do grau de Doutor em Engenharia Elétrica.

ALGORITMOS ADAPTATIVOS COM SINAL DE ENTRADA NORMALIZADO: MODELAGEM ESTATÍSTICA E APRIMORAMENTOS

Javier Ernesto Kolodziej

Agosto/2010

Orientador: Rui Seara, Dr.

Co-orientador: Orlando J. Tobias, Dr.

Área de Concentração: Comunicações e Processamento de Sinais.

Palavras-chave: Algoritmo CSG, algoritmo CSG melhorado, algoritmo LMS no domínio transformado, algoritmo NLMS, modelagem estatística.

Número de páginas: 155

Este trabalho apresenta uma análise estatística de três importantes algoritmos adaptativos baseados no gradiente estocástico utilizando sinais de entrada normalizados. Os algoritmos com sinal de entrada normalizado são utilizados como uma alternativa ao algoritmo LMS (*least-mean-square*) convencional, visando melhorar a velocidade de convergência (especialmente para sinais de entrada correlacionados), tornando o desempenho do filtro adaptativo mais robusto frente a variações de potência do sinal de entrada. São apresentados modelos estatísticos mais precisos para os algoritmos considerados, a saber: LMS normalizado (NLMS), LMS no domínio transformado (LMS-DT) e gradiente estocástico com restrições (*Constrained Stochastic Gradient - CSG*). Em particular, o algoritmo CSG, aqui discutido, é utilizado em controle de arranjos de antenas para sistemas celulares. Através do modelo do algoritmo CSG, é verificado um comportamento anômalo e é proposta uma versão melhorada para esse algoritmo. Para os outros algoritmos adaptativos estudados, os modelos obtidos apresentam maior precisão quando comparados com outros modelos disponíveis na literatura, permitindo um melhor domínio desses algoritmos para diferentes condições de operação.

Abstract of Theses presented to UFSC as a partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

NORMALIZED ADAPTIVE ALGORITHMS: STATISTICAL MODELING AND IMPROVEMENTS

Javier Ernesto Kolodziej

August/2010

Advisor: Rui Seara, Dr.

Co-advisor: Orlando J. Tobias, Dr.

Area of Concentration: Communications and Signal Processing.

Keywords: CSG algorithm, improved CSG algorithm, transform domain algorithm, NLMS algorithm, statistical modeling.

Number of pages: 155

This research work presents a statistical analysis for three important adaptive algorithms based on the stochastic gradient using normalized input signal. Algorithms with normalized input signal are used as an alternative to the standard least-mean-square (LMS) algorithm aiming to improve the convergence speed (especially for correlated input signal), increasing the adaptive filter robustness under input signal power variations. More accurate statistical models for the normalized LMS (NLMS), transform domain LMS (TDLMS), and constrained stochastic gradient (CSG) algorithms are presented. In particular, the CSG algorithm here considered is used for controlling antenna arrays in cellular systems. Through the CSG algorithm model, an anomalous behavior in its standard version is verified and an improved algorithm is also proposed. For the other algorithms, the obtained models are more accurate than the ones available in the literature, allowing a better and deeper understanding of these algorithms under different operating conditions.

SUMÁRIO

1 INTRODUÇÃO	27
1.1 Definição do Problema de Modelagem de Sistemas Adaptativos	28
1.1.1 Filtro de Wiener	28
1.1.2 Algoritmo LMS	29
1.2. Modelagem Estatística de Algoritmos Normalizados	30
1.2.1. Modelagem do algoritmo NLMS	30
1.2.2. Modelagem do algoritmo LMS-DT	33
1.2.3. Modelagem do algoritmo CSG	34
1.3. Organização do Trabalho	35
2 UM MODELO PRECISO PARA O ALGORITMO LMS NORMALIZADO.....	37
2.1. Introdução	37
2.2. Equação de Adaptação	37
2.3. Comportamento Médio do Vetor de Pesos	38
2.4. Curva de Aprendizagem.....	45
2.5. Resultados Experimentais	48
2.6. Conclusões e Comentários Finais	55
3 ANÁLISE ESTOCÁSTICA DO ALGORITMO LMS NO DOMÍNIO TRANSFORMADO PARA AMBIENTE NÃO-ESTACIONÁRIO	57
3.1. Introdução	57
3.2. Ambiente Não-estacionário.....	58
3.3. Modelo Proposto	59
3.3.1. Considerações de análise	61
3.3.2. Momento de primeira ordem de $\mathbf{v}_T(n)$	61

3.3.3. Momento de segunda ordem de $v_T(n)$	63
3.3.4. Curva de aprendizagem	67
3.4. Análise	68
3.4.1. Erro em excesso de regime permanente	68
3.4.2. Desajuste	69
3.4.3. Passo de adaptação para o mínimo erro em excesso	72
3.4.4. Grau de não-estacionaridade	75
3.4.5. Relação entre o desajuste e grau de não-estacionaridade ...	76
3.5. Modelos Simplificados	78
3.5.1. Momento de primeira ordem utilizando o PM	78
3.5.2. Momento de segunda ordem e curva de aprendizagem.....	79
3.5.3. Erro em excesso.....	79
3.6. Resultados de Simulação	80
3.6.1. Verificação do modelo proposto	81
3.6.2. Verificação das aproximações da seção 3.3.3	92
3.6.3. Resultados de comparação para as matrizes P e S.....	97
3.7. Conclusões.....	98

4 MODELAGEM ESTATÍSTICA DO ALGORITMO CSG APLICADA PARA UM ARRANJO ADAPTATIVO DE ANTENAS 99

4.1. Introdução.....	99
4.2. Algoritmo CSG.....	101
4.3. Modelagem Estatística do Algoritmo CSG.....	103
4.3.1. Momento de primeira ordem do vetor de pesos	103
4.3.2. Razão sinal/ruído-mais-interferência (SINR).....	105
4.4. Resultados de Simulação	106
4.4.1. Função de distribuição de probabilidades da SINR.....	107
4.4.2. Convergência da SINR.....	107
4.4.3. Convergência dos pesos	108
4.5. Conclusões.....	114

5 CONSIDERAÇÕES SOBRE O COMPORTAMENTO DO ALGORITMO CSG	115
5.1. Introdução	115
5.2. Equações de Atualização	115
5.3. Comportamento Não-Balanceado	116
5.3.1. Caso I: θ_s , θ_{u_1} e θ_{u_2} distantes	116
5.3.2. Caso II: $\theta_s = \theta_{u_1}$	117
5.4. Análise	119
5.4.1. Comportamento médio dos pesos	119
5.4.2. Discussão sobre o comportamento não-balanceado.....	120
5.4.3. Algoritmo CSG modificado.....	122
5.5. Desempenho do Algoritmo ICSG	124
5.5.1. Exemplo 1	124
5.5.2. Exemplo 2.....	126
5.5.3. Exemplo 3.....	128
5.6. Conclusões	130
6 CONCLUSÕES FINAIS E TRABALHOS FUTUROS	131
6.1. Conclusões	131
6.2. Trabalhos Futuros	134
REFERÊNCIAS BIBLIOGRÁFICAS	135
APÊNDICES.....	139
Apêndice 1 Cálculo da Matriz P	141
Apêndice 2 Determinação da Matriz S	147
Apêndice 3 Cálculo Exato das Matrizes de Covariância Normalizadas	151
Apêndice 4 Verificação do Autovalor Dominante.....	153

LISTA DE FIGURAS

Figura 1.1. Representação em diagrama de blocos do problema de filtragem estatística.	29
Figura 1.2. Diagrama de blocos de um filtro adaptativo.	30
Figura 2.1. Diagrama de blocos do algoritmo NLMS.	38
Figura 2.2. Comportamento médio dos pesos para sinal de entrada gaussiano e correlacionado $\chi = 74$, para uma planta de comprimento $N = 8$	49
Figura 2.3. Curva de aprendizagem (EQM) para sinal de entrada gaussiano e correlacionado.....	52
Figura 2.4. Curva de aprendizagem (EQM) para sinal de entrada SIRP correlacionado com função densidade de probabilidade marginal: (a) K_0 . (b) Função Gama generalizada. (c) Laplaciana.	54
Figura 3.1. Desajuste (3.46) como uma função de μ e a	71
Figura 3.2. Desajuste como uma função de μ	71
Figura 3.3. Curvas do passo de adaptação para o mínimo erro em excesso	74
Figura 3.4. Exemplo 1. Curvas do comportamento médio do vetor de erro.....	83
Figura 3.5. Exemplo 1. Curvas do EQM para $M = 8$	84
Figura 3.6. Exemplo 2. Curvas do EQM para $M = 16$	86
Figura 3.7. Exemplo 3. Curvas do EQM para $M = 64$	87

Figura 3.8. Exemplo 4. Curvas do EQM para o caso estacionário.	89
Figura 3.9. Exemplo 5. Resultados de comparação para o erro em excesso.....	91
Figura 3.10. Teste da Aproximação (3.24)	93
Figura 3.11. Teste da Aproximação (3.25)	94
Figura 3.12. Teste da Aproximação (3.27)	95
Figura 3.13. Teste da Aproximação (3.29)	96
Figura 3.14. Teste de aproximação da matriz P	97
Figura 3.15. Teste de aproximação da matriz S	98
Figura 4.1. Configurações do sistema celular	99
Figura 4.2. Diagrama de blocos de um arranjo adaptativo de antenas.	102
Figura 4.3. Comportamento da função distribuição de probabilidade da SINR.....	110
Figura 4.4 Comportamento de convergência de $\gamma(n)$	111
Figura 4.5. Comportamento dos pesos.....	113
Figura 5.1. Evolução da SINR para os casos considerados.	117
Figura 5.2. Evolução da P_{IC} para o Caso II.....	118
Figura 5.3. Caso II. Diagrama de radiação	118
Figura 5.4. Exemplo 1. Curva da SINR.	124
Figura 5.5. Exemplo 1. Diagrama de radiação.....	125
Figura 5.6. Exemplo 2. Curvas da SINR.....	126

Figura 5.7. Exemplo 2. Diagramas de radiação	127
Figura 5.8. Exemplo 3. Curvas da SINR	128
Figura 5.9. Exemplo 3. Diagramas de radiação	129

LISTA DE TABELAS

Tabela 2.1. Dispersão dos autovalores de E_2	45
---	----

GLOSSÁRIO

AIH	— Aproximação da integral hiperelíptica
CSG	— <i>Constrained stochastic gradient</i>
EDE	— Equações diferenciais estocásticas
EQM	— Erro quadrático médio
EQME	— Erro quadrático médio em excesso
ERB	— Estação de rádio base
FDD	— <i>Frequency division duplex</i>
FDP	— Função densidade de probabilidade
HI	— Hipótese de independência
ICSG	— <i>Improved CSG</i>
INR	— Razão interferência-ruído
JDCF	— Janela deslizante de comprimento fixo
LMS	— <i>Least mean square</i>
LMS-DT	— LMS no domínio transformado
MC	— Monte Carlo
NLMS	— LMS normalizado
PM	— Princípio da media
SG	— <i>Stochastic gradient</i>
SINR	— <i>Signal-to-interference-plus-noise ratio</i>
SIRP	— <i>Spherically invariant random process</i>
SNR	— <i>Signal-to-noise ratio</i>
TCD	— Transformada do cosseno discreta
TLK	— Transformada de Karhunen-Loève
TM	— Terminal móvel

As pesquisas referentes a sistemas adaptativos compreendem diferentes tópicos, dentre os quais se podem destacar: projeto de novos algoritmos de adaptação e estruturas de filtros adaptativos, determinação de parâmetros ótimos e modelagem estatística do comportamento do sistema adaptativo. A presente tese tem seu foco, principalmente, neste último tópico, desenvolvendo modelos estatísticos mais realísticos de alguns importantes algoritmos adaptativos. Adicionalmente, utilizam-se as expressões dos modelos obtidos para uma melhor compreensão do funcionamento dos algoritmos adaptativos selecionados para estudo, visando projetá-los corretamente e aplicá-los com sucesso em situações práticas.

Em geral, a modelagem estatística dos sistemas adaptativos¹ é uma tarefa complexa e, apesar dos esforços até então realizados, não existe ainda uma teoria unificada de análise para todos os algoritmos encontrados na literatura [1], [2]. Assim, a modelagem estatística deve encaminhar-se seguindo as particularidades associadas a cada tipo de algoritmo de adaptação considerado. É por isso que o presente trabalho se restringe à modelagem de algumas variantes dos algoritmos adaptativos baseados no gradiente estocástico. O membro mais importante dessa família é o algoritmo LMS (*least-mean-square*). Particularmente, são apresentados aqui novos e mais precisos modelos para três destacados algoritmos dessa classe, considerando o sinal de entrada normalizado, a saber: algoritmo LMS normalizado (NLMS), algoritmo LMS no domínio transformado (LMS-DT) e algoritmo do gradiente estocástico com restrições (CSG). Além do mais, uma versão melhorada para o algoritmo CSG é proposta como também parâmetros de funcionamento ótimo para o algoritmo LMS-DT são apresentados.

A área de filtragem adaptativa é abrangente, interligando diferentes conceitos, tais como cálculo diferencial e integral, processos estocásticos e teoria da informação. Visando uma clara apresentação desses conceitos, este capítulo é dividido em duas partes. Primeiramente, são apresentados os fundamentos da filtragem adaptativa e o interesse da modelagem dos sistemas adaptativos; em seguida, são

¹ Cabe ressaltar que na literatura da área os termos sistemas adaptativos, filtros adaptativos e algoritmos adaptativos são utilizados indistintamente.

detalhados o estado-da-arte da modelagem dos sistemas adaptativos aqui considerados e as contribuições específicas deste trabalho de pesquisa.

1.1 DEFINIÇÃO DO PROBLEMA DE MODELAGEM DE SISTEMAS ADAPTATIVOS

Para compreender a operação de um sistema adaptativo, primeiramente, é apresentada uma breve revisão dos fundamentos da teoria de estimação linear.

1.1.1 FILTRO DE WIENER

Os primeiros trabalhos na área de estimação linear foram realizados, de forma independente, por Kolmogorov [3] e Wiener [4], respectivamente, no final da década de 1930 e início da década de 1940. Kolmogorov desenvolveu um abrangente trabalho sobre previsão linear para processos estocásticos em tempo discreto. Por outro lado, Wiener formulou o problema de previsão linear em tempo contínuo, derivando uma formulação explícita para o estimador ótimo, conhecida como equação de Wiener-Hopf [5]. Para tal, Wiener considerou o problema de estimação de um processo corrompido por ruído aditivo, que é basicamente um problema de filtragem.

Em 1947, Levinson [6] formulou o problema de filtragem de Wiener em tempo discreto (veja Figura 1.1), assumindo a seguinte forma matricial:

$$\mathbf{R}\mathbf{w}_o = \mathbf{p} \quad (1.1)$$

onde \mathbf{w}_o é o vetor de pesos ótimos do filtro de Wiener que minimiza o erro de estimação $e(n)$ no sentido quadrático médio, \mathbf{R} é a matriz de autocorrelação do sinal de entrada $x(n)$, \mathbf{p} é o vetor de correlação cruzada entre o sinal de entrada $x(n)$ e a resposta desejada $d(n)$. A solução da equação (1.1) nem sempre é viável do ponto de vista prático, visto que se requer o conhecimento *a priori* das estatísticas dos sinais envolvidos no processo. Por outro lado, através de algoritmos adaptativos, podemos tornar tal solução viável do ponto de vista prático.

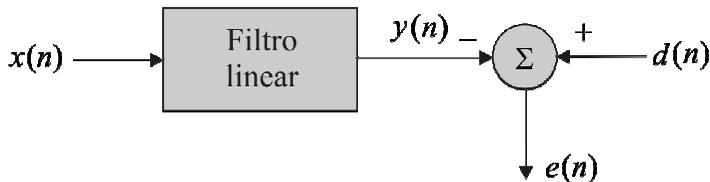


Figura 1.1. Representação em diagrama de blocos do problema de filtragem estatística.

Dentre os principais algoritmos de adaptação conhecidos na literatura, o algoritmo LMS é um dos mais utilizados, fornecendo um mecanismo iterativo capaz de atingir a solução dada em (1.1), considerando estimativas instantâneas ao invés das estatísticas dos sinais (veja Figura 1.2). Tal caso é discutido na próxima seção.

1.1.2 ALGORITMO LMS

O sistema adaptativo mais utilizado por sua simplicidade e robustez é o que emprega o LMS como algoritmo de adaptação associado a um filtro com resposta ao impulso finita. Tal estrutura foi concebida por Widrow e Hoff em 1965 no contexto de um estudo sobre reconhecimento de padrões (comumente referido na literatura como *Adaline*) [7]. O algoritmo LMS atualiza cada peso do filtro na direção oposta ao gradiente da função custo, projetado sobre o hiperplano dos pesos. A função custo considerada é definida como o sinal de erro ao quadrado. A direção de atualização dos pesos é estimada diretamente utilizando o produto do vetor de sinal de entrada do filtro e o sinal de erro de estimação. Esse procedimento faz com que o algoritmo LMS apresente duas desvantagens. Em primeiro lugar, a velocidade de convergência é afetada pelas variações de amplitude do sinal de entrada. Isto é, quando o nível de potência do sinal de entrada é alto, o algoritmo LMS amplifica o ruído do gradiente podendo acarretar uma instabilidade no sistema. Por outro lado, se o nível de potência do sinal de entrada é baixo, a convergência torna-se muito lenta [8]. Outra desvantagem é que a convergência também é afetada (torna-se mais lenta) quando aumenta a dispersão dos autovalores da matriz de autocorrelação do sinal de entrada [8], [9]. Assim, diversas técnicas vêm sendo propostas na literatura visando contornar tais problemas, levando a novos e aprimorados algoritmos.

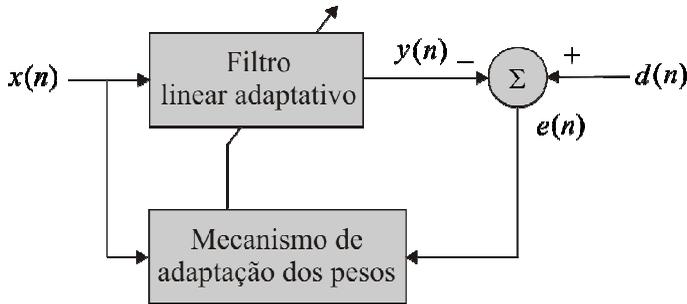


Figura 1.2. Diagrama de blocos de um filtro adaptativo.

Uma maneira de se entender melhor o funcionamento dos algoritmos adaptativos e quais parâmetros dominam seu comportamento é através de sua modelagem estatística. Em particular, é de interesse obter equações descrevendo o comportamento médio do vetor de pesos e da curva de aprendizagem. Por exemplo, os modelos estatísticos permitem inferir valores de parâmetros do algoritmo que garantam a estabilidade e determinar expressões para diversas figuras de mérito, tais como o erro em excesso, desajuste, velocidade de convergência e passo de adaptação ótimo.

1.2. MODELAGEM ESTATÍSTICA DE ALGORITMOS NORMALIZADOS

A modelagem de sistemas adaptativos é uma tarefa não trivial, requerendo um vasto conhecimento de diversas ferramentas matemáticas e de análise de sinais, apresentando grandes desafios teóricos que persistem até o presente momento [1]. Assim, o presente trabalho visa contribuir com um melhor entendimento de uma classe particular de algoritmos adaptativos através da proposta de novos modelos estatísticos. Os algoritmos abordados aqui são os seguintes: LMS normalizado (*normalized least-mean-square* - NLMS), LMS no domínio transformado (LMS-DT) e algoritmo do gradiente estocástico com restrições (*constrained stochastic gradient* - CSG).

1.2.1. MODELAGEM DO ALGORITMO NLMS

O algoritmo NLMS foi apresentado na literatura de forma independente por Nagumo e Noda [10] e Albert e Gardner [11] em 1967, porém, a terminologia atual de LMS normalizado foi proposta por

Bitmead e Anderson [12] em 1980. O algoritmo NLMS normaliza a estimação do gradiente, segundo a norma euclidiana ao quadrado do vetor de sinal de entrada, fazendo com que a velocidade de convergência não seja afetada pelas variações da potência do sinal de entrada. O algoritmo NLMS é o mais popular dentre os algoritmos com sinal de entrada normalizado.

Na bibliografia disponível sobre modelagem de algoritmos normalizados se destaca o trabalho de Rupp [13], o qual propõe a modelagem do algoritmo NLMS considerando apenas independência estatística entre o vetor de pesos e os dados de entrada. O uso de uma única hipótese simplificativa fornece um modelo preciso; porém, leva a um obstáculo matemático no cálculo dos valores esperados do tipo

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^T(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] \quad (1.2)$$

sendo aqui $\mathbf{x}(n)$ o vetor de sinal de entrada. O cálculo exato desses valores esperados para o sinal de entrada sendo um processo aleatório esfericamente invariante² e correlacionado envolve a solução de uma integral hiperelíptica de alta ordem.

Diversos autores vêm evitando o cálculo destas integrais usando muitas vezes aproximações mais ou menos grosseiras ou não válidas [8], [14]-[18]. Assim, em [8] é utilizada a seguinte aproximação:

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^T(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] \cong \frac{E[\mathbf{x}(n)\mathbf{x}^T(n)]}{E[\mathbf{x}^T(n)\mathbf{x}(n)]} = \frac{1}{N\sigma_x^2} \mathbf{R} \quad (1.3)$$

onde σ_x^2 é a variância do sinal de entrada e N é o número de pesos do filtro, levando a resultados de modelagem satisfatórios para filtros adaptativos tendo um número grande de pesos e se o sinal de entrada é não-correlacionado. Em [14] e [15], são associados o Princípio da Média (PM) [19] e a hipótese de independência do sinal de entrada [8] para obter a aproximação

² Do Inglês *spherically invariant random process* – SIRP; processo estocástico estacionário no qual as funções de densidade conjunta podem ser calculadas a partir de uma função de densidade unidimensional (o processo gaussiano é um caso particular do SIRP) [13].

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^T(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] \cong E \left[\frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] E[\mathbf{x}(n)\mathbf{x}^T(n)] \cong \frac{1}{(N-2)\sigma_x^2} \mathbf{R}. \quad (1.4)$$

Apesar de esta estratégia simplificar notavelmente a matemática envolvida, a hipótese de independência não é apropriada para sinais de entrada correlacionados nem o PM é válido para filtros de curto comprimento, resultando em um modelo bastante limitado.

Posteriormente, em [16], apenas o PM é considerado para obter uma solução aproximada de integrais hiperelípticas, sendo os resultados satisfatórios para sinais de entrada correlacionados, entretanto, ainda restrita às condições de aplicação do PM.

Uma outra estratégia, proposta em [17], consiste na utilização de uma abordagem baseada em equações diferenciais estocásticas (EDE) (*stochastic differential equation* – SDE). No entanto, tal abordagem é restrita a passos de adaptação infinitesimais, sendo necessária também a determinação do cálculo de valores esperados com termos normalizados.

Recentemente, em [18], é apresentada uma nova estratégia de análise do vetor de erro nos pesos do algoritmo NLMS. Esse procedimento não invoca o princípio da independência, não requer que o passo de adaptação seja pequeno e evita também o cálculo de valores esperados com termos normalizados. No entanto, não são fornecidas expressões que possam ser úteis para o projeto do algoritmo adaptativo. A busca de um equacionamento para a curva de aprendizagem e outras estatísticas de primeira e segunda ordens, seguindo tal estratégia, leva ao cálculo de valores esperados com termos normalizados, recaindo assim no problema antes mencionado que é determinar valores esperados dependentes do cálculo de integrais hiperelípticas.

Um dos objetivos deste trabalho é obter um modelo para o algoritmo NLMS com a maior precisão possível. Para tal, existe a necessidade de se considerar um mínimo de suposições simplificativas. Particularmente, procura-se obter uma formulação fechada da proposta de modelagem discutida em [13], propondo-se um método aproximado para calcular os valores esperados envolvidos sem utilizar o PM. Tal procedimento leva a um modelo mais preciso dos até agora encontrados na literatura sem restringir a ordem do filtro adaptativo, especialmente quando considerados sinais de entrada correlacionados.

1.2.2. MODELAGEM DO ALGORITMO LMS-DT

O algoritmo LMS-DT foi proposto por Narayan et al. [20] objetivando melhorar as características de convergência do algoritmo LMS padrão. Especificamente, o algoritmo LMS-DT reduz o grau de correlação do sinal de entrada do filtro, dessa forma, aumentando a velocidade de convergência do algoritmo. Essa descorrelação é obtida por meio da combinação de dois processos. Primeiro, uma transformação do sinal separa o sinal de entrada em sub-bandas, obtendo uma matriz de autocorrelação quase diagonal; em seguida, o conteúdo energético em cada sub-banda é normalizado, reduzindo a dispersão dos autovalores da mencionada matriz.

Para normalizar o conteúdo energético em cada sub-banda, é necessário estimar a variância da sub-banda. O procedimento de modelagem deve considerar tal condição para obter um modelo preciso. Contudo, quando considerado o impacto do estimador na modelagem, surge uma importante dificuldade, especificamente, valores esperados do tipo

$$E[\hat{\mathbf{D}}^{-1}(n)\hat{\mathbf{R}}(n)] \quad (1.5)$$

devem ser determinados, com $\hat{\mathbf{D}}(n)$ e $\hat{\mathbf{R}}(n)$ sendo matrizes que dependem do sinal de entrada (o acento circunflexo indica que tais matrizes surgem de um processo de estimação). Para resolver (1.5), várias aproximações vêm sendo utilizadas [8], [21]-[24]. A que produz resultados mais precisos é a apresentada em [24]. Nessa referência, o PM é utilizado para separar valores esperados do tipo (1.5), como

$$E[\hat{\mathbf{D}}^{-1}(n)\hat{\mathbf{R}}(n)] \cong E[\hat{\mathbf{D}}^{-1}(n)]E[\hat{\mathbf{R}}(n)] \quad (1.6)$$

onde $E[\hat{\mathbf{R}}(n)]$ é diretamente determinado a partir das características estatísticas do sinal de entrada; entretanto, $E[\hat{\mathbf{D}}^{-1}(n)]$ é obtido calculando de forma aproximada as integrais que surgem do operador valor esperado [24]. A aplicação do PM resulta em expressões simples, porém existem casos de interesse prático que não são bem modelados. Assim, buscando um modelo preciso em todas as condições de operação do algoritmo, o PM não é o mais recomendado.

Neste trabalho, é proposto um modelo em que o valor esperado (1.5) é calculado sem invocar o PM, objetivando uma maior generalidade. O modelo proposto é derivado considerando ambientes não-estacionários (planta variante no tempo). Essa situação tem grande importância prática, sendo o caso estacionário obtido diretamente a partir das expressões propostas. Expressões para o erro em excesso, desajuste e passo de adaptação para o mínimo erro em excesso são também derivadas. Além do mais, uma expressão que relaciona o grau de não-estacionaridade com o desajuste do algoritmo é fornecida. Comparações entre resultados obtidos a partir de simulações de Monte Carlo (MC) e das expressões teóricas confirmam uma muito boa precisão do modelo proposto. Todas as simulações apresentadas no neste trabalho foram obtidas utilizando o software Matlab®.

1.2.3. MODELAGEM DO ALGORITMO CSG

O algoritmo CSG foi proposto por Morgan em 2003 [25], [26], visando resolver um problema de autovalores generalizados. Para tal, é maximizada uma função objetivo envolvendo uma razão de polinômios, conhecida como razão de Rayleigh [25]. Na literatura técnica, existem vários algoritmos utilizados para resolver o problema de autovalores generalizados, como, por exemplo, o algoritmo quase-Newton [27] e o algoritmo do gradiente estocástico (SG), também discutidos em [25] e [26]. Porém, o algoritmo CSG se destaca por apresentar um menor custo computacional (da ordem do número de pesos) com uma ligeira redução na velocidade de convergência.

Até o momento, na literatura, não existem análises estatísticas do comportamento do algoritmo CSG. Portanto, neste trabalho é proposto um modelo para tal algoritmo, considerando a condição de passo de adaptação pequeno. O algoritmo CSG, aqui discutido, é utilizado no contexto de arranjos de antenas em sistemas celulares para obter os pesos ótimos, visando maximizar a razão sinal/ruído-mais-interferência (*signal-to-interference-plus-noise ratio* - SINR) do sistema [25]. Nessa aplicação, os sinais considerados são representados por números complexos, situação que permite obter expressões fechadas para os valores esperados envolvidos.

Uma das contribuições originais deste trabalho é que, através do modelo proposto, é identificado um comportamento anômalo (denominado aqui comportamento não-balanceado) quando o algoritmo GSC é utilizado no controle de arranjos de antenas em sistemas celulares sob certas condições de trabalho. Utilizando as expressões do

modelo estatístico do algoritmo CSG, é obtida uma explicação teórica para a existência de tal comportamento, bem como é proposta uma versão melhorada desse algoritmo visando contornar tal problema.

1.3. ORGANIZAÇÃO DO TRABALHO

O presente trabalho de tese está organizado como segue. No Capítulo 2, é proposto um modelo estocástico preciso para descrever o comportamento do algoritmo LMS normalizado (NLMS), considerando sinais de entrada correlacionados e um ambiente estacionário. No Capítulo 3, um modelo estocástico para o algoritmo LMS-DT, levando em conta um ambiente não-estacionário é proposto, derivando-se também expressões para o erro em excesso, desajuste e passo de adaptação para o mínimo erro em excesso. No Capítulo 4, é apresentado um modelo que descreve o comportamento médio dos pesos e a razão sinal/ruído-mais-interferência do algoritmo CSG, considerando uma aplicação de arranjo de antenas. No Capítulo 5, uma versão melhorada para o algoritmo CSG é proposta. Finalmente, o Capítulo 6 apresenta as conclusões desta tese de doutorado e sugestões para trabalhos futuros.

2.1. INTRODUÇÃO

Neste capítulo, é proposto um modelo estocástico preciso para descrever o comportamento do algoritmo LMS normalizado (NLMS), considerando sinais de entrada correlacionados obtidos através de processos aleatórios esfericamente invariantes (SIRP). Tais processos podem descrever tanto sinais gaussianos quanto uma ampla gama de sinais não-gaussianos, incluindo aqueles com função densidade de probabilidade marginal laplaciana, K_0 e Gama. Esses processos são de grande utilidade para modelar uma classe importante de sinais reais usados em aplicações de filtragem adaptativa. O modelo obtido apresenta melhores resultados do que outros concorrentes existentes na literatura, com sua precisão confirmada através de simulações numéricas.

2.2. EQUAÇÃO DE ADAPTAÇÃO

A Figura 2.1 ilustra o diagrama de blocos do algoritmo NLMS. Nessa figura, $x(n)$ representa o sinal de entrada; $d(n)$, o sinal desejado; $y(n)$, o sinal de saída; $e(n)$, o sinal erro de estimação; $w_i(n)$, os pesos do filtro adaptativo e $z(n)$, o ruído de medição.

A equação de atualização dos pesos do algoritmo NLMS é dada por [8]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{e(n)\mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (2.1)$$

onde

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

é o erro de estimação $\mathbf{w}(n) = [w(n) \ w(n-1) \cdots \ w(n-N+1)]^T$ representa o vetor contendo os N pesos do filtro adaptativo, $z(n)$, um ruído de medição, i.i.d., com média zero, variância σ_z^2 e não correlacionado com qualquer outro sinal do sistema. O parâmetro μ controla a velocidade de convergência do algoritmo. O vetor de entrada é $\mathbf{x}(n) = [x(n) \ x(n-1) \cdots \ x(n-N+1)]^T$, sendo σ_x^2 a variância do processo $\{x(n)\}$.

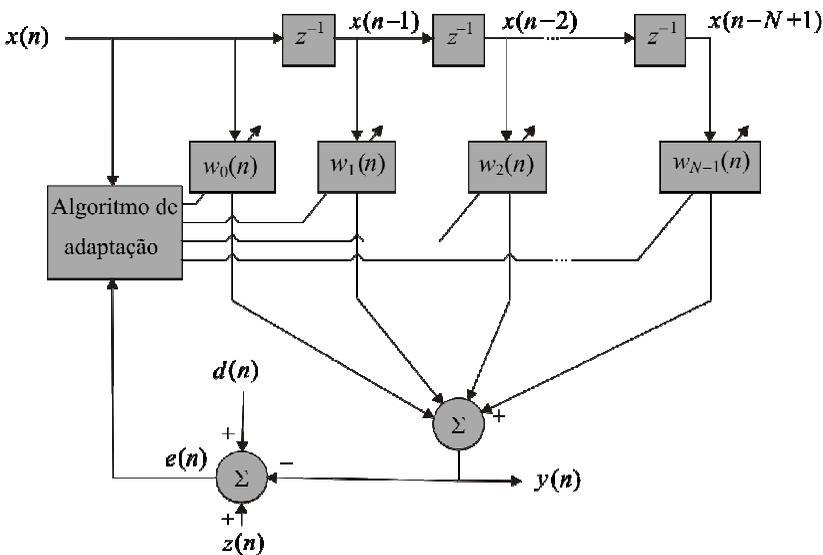


Figura 2.1. Diagrama de blocos do algoritmo NLMS.

2.3. COMPORTAMENTO MÉDIO DO VETOR DE PESOS

Substituindo (2.2) em (2.1) e tomando o valor esperado de ambos os lados da expressão resultante, obtém-se

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu E \left[\frac{\mathbf{x}(n)d(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] - \mu E \left[\frac{\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] + \mu E \left[\frac{\mathbf{x}(n)z(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right]. \quad (2.3)$$

Para determinar os valores esperados em (2.3), consideramos que $\mathbf{w}(n)$ e $\mathbf{x}(n)$ são estatisticamente independentes (assumindo condição de adaptação lenta) [8]. O quarto termo do lado direito de (2.3) é zero, decorrente do fato de $z(n)$ apresentar média nula [8]. Assim, podemos reescrever (2.3) como

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \mu \underbrace{E\left[\frac{\mathbf{x}(n)d(n)}{\mathbf{x}^T(n)\mathbf{x}(n)}\right]}_{E_1} - \mu \underbrace{E\left[\frac{\mathbf{x}(n)\mathbf{x}^T(n)}{\mathbf{x}^T(n)\mathbf{x}(n)}\right]}_{E_2} E[\mathbf{w}(n)]. \quad (2.4)$$

O ponto principal agora é determinar os valores esperados E_1 e E_2 de (2.4). O procedimento para determinar tais valores esperados é notavelmente complexo, visto que, para tal, é requerido o cálculo de uma integral hiperelíptica de alta ordem. Para a solução dessa integral não se dispõe de uma expressão fechada, como pode ser verificado na literatura [28]. Em [13], E_1 e E_2 são resolvidas considerando filtros adaptativos de até quarta ordem. Recentemente em [14] e [15], E_1 e E_2 têm sido calculados associando o PM e a hipótese de independência (HI) do sinal de entrada. Essa estratégia, denominada aqui PM/HI, simplifica notavelmente a matemática envolvida. No entanto, a hipótese de independência não é verdadeira para sinais de entrada autocorrelacionados. Lembrando que o propósito do algoritmo NLMS é particularmente focado no tratamento de sinais correlacionados, ignorar tal correlação (com o objetivo de simplificar a matemática) pode levar a um modelo impreciso. Por exemplo, utilizando o procedimento apresentado em [14] e [15], os valores esperados E_1 e E_2 são, respectivamente, dados por

$$E\left[\frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)}\right] E[d(n)\mathbf{x}(n)] = \frac{1}{(N-2)\sigma_x^2} \mathbf{p} \quad (2.5)$$

e

$$E\left[\frac{1}{\mathbf{x}^T(n)\mathbf{x}(n)}\right] E[\mathbf{x}(n)\mathbf{x}^T(n)] = \frac{1}{(N-2)\sigma_x^2} \mathbf{R} \quad (2.6)$$

onde $\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ e $\mathbf{p} = E[d(n)\mathbf{x}(n)]$ são, respectivamente, a matriz de autocorrelação do sinal de entrada e o vetor de correlação cruzada entre o sinal desejado e o sinal de entrada. Os valores esperados (2.5) e (2.6) são verdadeiros se o processo $\{x(n)\}$ é gaussiano, independente e de média zero [29]. No caso de dados correlacionados, (2.5) e (2.6) tornam-se aproximações imprecisas.

Recentemente, Lobato et al. [16] invocaram o PM associado a uma solução aproximada de integrais hiperelípticas, estratégia chamada aqui PM/AIH (aproximação da integral hiperelíptica), obtendo uma solução satisfatória para sinais de entrada correlacionados, porém restritas às condições de aplicação do PM.

As estratégias baseadas no PM supõem a ergodicidade do processo de entrada, produzindo nesse caso resultados aceitáveis se a dimensão do vetor $\mathbf{x}(n)$ for suficientemente grande. Além do mais, visto que um processo gaussiano estacionário é o único SIRP ergódico [13], não se pode usar o PM para um processo SIRP não-gaussiano.

Barrault et al. [17] usaram uma abordagem baseada em equações diferenciais estocásticas (EDE) (*stochastic differential equation* – SDE). Tal abordagem fornece previsões satisfatórias unicamente para valores de passo de adaptação próximos da unidade.

Procurando então considerar um mínimo de suposições simplificativas, este trabalho visa a generalização da proposta de modelagem do algoritmo NLMS discutida em [13], apresentando agora um caminho aproximado (porém bastante preciso) para calcular as integrais hiperelípticas de alta ordem, levando a um modelo preciso para sinais de entrada correlacionados.

Consideremos agora o termo E_2 de (2.4) e definindo a matriz

$$\mathbf{X}(n) = \frac{\mathbf{x}(n)\mathbf{x}^T(n)}{\mathbf{x}^T(n)\mathbf{x}(n)} \quad (2.7)$$

o valor esperado de cada elemento de $\mathbf{X}(n)$ é (por definição) expresso como

$$E \left[\frac{x(n-i+1)x(n-j+1)}{\mathbf{x}^T(n)\mathbf{x}(n)} \right] = \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{N \text{ integrais}} \frac{x(n-i+1)x(n-j+1)}{\mathbf{x}^T(n)\mathbf{x}(n)} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \quad (2.8)$$

onde $f_{\mathbf{x}}(\mathbf{x})$ representa a função densidade de probabilidade multivariada do vetor de entrada $\mathbf{x}(n)$. A função densidade de probabilidade de um sinal $\mathbf{x}(n)$ SIRP, correlacionado de ordem N , pode ser representada por

$$f_{\mathbf{x}}(\mathbf{x}) = \frac{1}{\sqrt{\pi^N \det(\mathbf{R})}} f_{\mathbf{u}}(\mathbf{x}^T \mathbf{R}^{-1} \mathbf{x}, N) \quad (2.9)$$

com $\det(\cdot)$ denotando o determinante da uma matriz e $f_{\mathbf{u}}(\cdot, N)$ é uma função que descreve um processo SIRP não-correlacionado de ordem N .

Agora, definindo uma função auxiliar (como em [13]), denotada por $f_{i,j}(\omega)$, dada por

$$f_{i,j}(\omega) = \frac{1}{\sqrt{\pi^N \det(\mathbf{R})}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{N \text{ integrais}} \frac{x(n-i+1)x(n-j+1)}{\mathbf{x}^T \mathbf{x}} f_{\mathbf{u}}(\mathbf{x}^T \mathbf{L}^{-1} \mathbf{x}, N) d\mathbf{x} \quad (2.10)$$

para $i=1, \dots, N$ e $j=1, \dots, N$, onde $\mathbf{L} = (\mathbf{R}^{-1} + 2\omega \mathbf{I})^{-1}$, o valor esperado (2.8) pode ser alternativamente escrito como

$$\{E[\mathbf{X}(n)]\}_{i,j} = f_{i,j}(0). \quad (2.11)$$

Diferenciando (2.10) em relação a ω , utilizando a definição de correlação cruzada e integrando a expressão resultante, obtém-se

$$f_{i,j}(0) = \int_0^{\infty} \frac{[\mathbf{R}(\mathbf{I} + 2\omega \mathbf{R})^{-1}]_{i,j}}{\sqrt{\det(\mathbf{I} + 2\omega \mathbf{R})}} d\omega \quad (2.12)$$

onde \mathbf{I} representa a matriz identidade.

Na literatura técnica, não existe uma expressão fechada para o cálculo da integral (2.12), sendo que o principal obstáculo é a presença do operador raiz quadrada em seu denominador. Assim, utiliza-se um

procedimento aproximado para resolver tal integral, conforme detalhado a seguir.

Visto que a matriz de autocorrelação \mathbf{R} é, em geral, definida positiva, podemos utilizar o seguinte procedimento para calcular a integral em questão:

- i) Decompor \mathbf{R} de acordo com $\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$, onde \mathbf{Q} é a matriz de autovetores de \mathbf{R} e $\mathbf{\Lambda}$ é uma matriz diagonal na qual seus elementos são os autovalores λ_i de \mathbf{R} . Assim,

$$E[\mathbf{X}(n)] = \mathbf{Q}\mathbf{H}\mathbf{Q}^T \quad (2.13)$$

em que \mathbf{H} é uma matriz diagonal, com elementos dados por

$$h_k = \int_0^{\infty} \frac{1}{\sqrt{\det(\mathbf{I} + 2\omega\mathbf{\Lambda})}} \frac{\lambda_k}{(1 + 2\omega\lambda_k)} d\omega. \quad (2.14)$$

- ii) Expressar o determinante do denominador de (2.12) como um polinômio $P(\omega)$ de ordem N , tal que

$$\sqrt{P(\omega)} = \sqrt{\det(\mathbf{I} + 2\omega\mathbf{\Lambda})} = \sqrt{\prod_{i=1}^N (1 + 2\omega\lambda_i)} = \sqrt{2^N a_N \prod_{i=1}^N (\omega - \omega_i)}. \quad (2.15)$$

Em (2.15), o coeficiente a_N é dado por $a_N = \prod_{i=1}^N \lambda_i$ e as raízes do polinômio são $\omega_i = -1/(2\lambda_i)$, para $i = 1, 2, \dots, N$.

- iii) Substituir pares de autovalores λ_i de \mathbf{R} adjacentes por sua média geométrica com multiplicidade dois [16], dada por

$$\lambda'_q = \sqrt{\lambda_{2q-1} \lambda_{2q}}, \quad q = 1, 2, \dots, N/2 \text{ com } N \text{ par.} \quad (2.16)$$

Nesta linha, um polinômio aproximado é obtido, tendo raízes com multiplicidade dois. Logo, (2.14) é reescrito como segue:

$$h_i \cong \int_0^{\infty} \frac{\lambda_i d\omega}{(1 + 2\omega\lambda_i) \sqrt{\prod_{q=1}^{N/2} (1 + 2\omega\lambda'_q)^2}}. \quad (2.17)$$

Visto que estamos interessados apenas nas soluções positivas de (2.17), o operador raiz quadrada junto com os termos quadráticos podem ser substituídos por um operador valor absoluto, resultando em

$$h_i \cong \int_0^{\infty} \frac{\lambda_i d\omega}{(1 + 2\omega\lambda_i) \left| \prod_{q=1}^{N/2} (1 + 2\omega\lambda'_q) \right|}. \quad (2.18)$$

O intervalo de integração de (2.18) é $\omega \geq 0$, implicando que o valor absoluto pode ser desconsiderado, dando lugar a

$$h_i \cong \int_0^{\infty} \frac{\lambda_i d\omega}{(1 + 2\omega\lambda_i) \prod_{q=1}^{N/2} (1 + 2\omega\lambda'_q)}. \quad (2.19)$$

Logo, considerando que em (2.19) há somente raízes reais e distintas (o caso mais recorrente para sinais de entrada correlacionados), a expansão em frações parciais do integrando de (2.19) nos permite obter uma solução fechada para essa integral. Assim,

$$h_i \cong \frac{1}{2\sqrt{a_N}} \left[\sum_{q=1}^{N/2} A_{i,q} \ln(\lambda'_q) + B_i \ln(\lambda_i) \right] \quad (2.20)$$

com

$$A_{i,q} = \frac{\lambda'_q \lambda_i}{\lambda'_q - \lambda_i} \prod_{\substack{j=1 \\ j \neq q}}^{N/2} \frac{\lambda'_q \lambda'_j}{\lambda'_q - \lambda'_j} \quad (2.21)$$

e

$$B_i = \prod_{q=1}^{N/2} \frac{\lambda_i \lambda'_q}{\lambda_i - \lambda'_q}. \quad (2.22)$$

O valor esperado E_1 em (2.4) é obtido utilizando um vetor auxiliar através de um procedimento similar ao considerado em (2.8)-(2.12). Assim,

$$E \left\{ \left[\begin{array}{c} \mathbf{x}(n)d(n) \\ \mathbf{x}^T(n)\mathbf{x}(n) \end{array} \right]_i \right\} = \int_0^\infty \frac{\{[\mathbf{I} + 2\omega\mathbf{R}]^{-1}\mathbf{p}\}_i}{\sqrt{\det(\mathbf{I} + 2\omega\mathbf{\Lambda})}} d\omega \quad (2.23)$$

e, utilizando a decomposição $\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$, obtemos

$$E \left[\begin{array}{c} \mathbf{x}(n)d(n) \\ \mathbf{x}^T(n)\mathbf{x}(n) \end{array} \right] = \mathbf{Q}\mathbf{\Lambda}^{-1}\mathbf{H}\mathbf{Q}^T\mathbf{p}. \quad (2.24)$$

Finalmente, substituindo (2.13) e (2.24) em (2.4), a expressão do vetor dos pesos médios (modelo proposto) é dada por

$$E[\mathbf{w}(n+1)] = [\mathbf{I} - \mu\mathbf{Q}\mathbf{H}\mathbf{Q}^T] E[\mathbf{w}(n)] + \mu\mathbf{Q}\mathbf{\Lambda}^{-1}\mathbf{H}\mathbf{Q}^T\mathbf{p}. \quad (2.25)$$

De acordo com os resultados obtidos nas simulações de Monte Carlo (MC) (200 realizações), a matriz representada por E_2 tem menor dispersão dos autovalores do que a matriz \mathbf{R} (evidenciando a vantagem do algoritmo NLMS sobre o LMS). O método aqui proposto para calcular E_2 preserva tal característica. Em contraste, quando E_2 é obtido usando o PM, a dispersão dos seus autovalores difere da obtida por simulação. A Tabela 2.1 apresenta uma comparação entre os valores de dispersão considerando os parâmetros utilizados no primeiro exemplo da Seção 2.5. Nesse exemplo é verificada a diferença existente entre os modelos utilizando PM e o aqui proposto, indicando que o novo modelo prediz muito bem o referido parâmetro.

Tabela 2.1. Dispersão dos autovalores de E_2 .

Método utilizado	Simulações de MC	PM	Método proposto
Dispersão dos autovalores	16,47	35,11	16,67

2.4. CURVA DE APRENDIZAGEM

Nesta seção, uma expressão que descreve o comportamento do erro quadrático médio (EQM) do algoritmo NLMS é determinada. Para tal, é definido o vetor de erro nos pesos como $\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_o$, onde \mathbf{w}_o é o vetor de pesos ótimo. Assim, o erro de estimação pode ser expresso como

$$e(n) = d(n) - \mathbf{x}^T(n)\mathbf{v}(n) - \mathbf{x}^T(n)\mathbf{w}_o + z(n) = e_o(n) - \mathbf{x}^T(n)\mathbf{v}(n) \quad (2.26)$$

com

$$e_o(n) = d(n) - \mathbf{x}^T(n)\mathbf{w}_o + z(n). \quad (2.27)$$

Elevando ao quadrado ambos os lados de (2.26), determinando o valor esperado da expressão resultante e usando o Princípio da Ortogonalidade [9], obtém-se

$$\begin{aligned} E[e^2(n)] &= e_{\min} + E[\mathbf{x}^T(n)\mathbf{v}(n)\mathbf{x}^T(n)\mathbf{v}(n)] = \\ &= e_{\min} + \text{tr}[\mathbf{R} \mathbf{K}(n)] = e_{\min} + \boldsymbol{\lambda}^T \mathbf{k}(n) \end{aligned} \quad (2.28)$$

onde $e_{\min} = E[e_o^2(n)]$ denota o erro mínimo, $\mathbf{K}(n) = E[\mathbf{v}(n)\mathbf{v}^T(n)]$ é a matriz de covariância do vetor de erro nos pesos (momento de segunda ordem), $\boldsymbol{\lambda}$ denota o vetor contendo os autovalores λ_i de \mathbf{R} e $\mathbf{k}(n)$ é o vetor com os elementos da diagonal principal da matriz $\mathbf{K}'(n) = \mathbf{Q}^T \mathbf{K}(n) \mathbf{Q}$. Note que (2.28) é completamente determinada se o

vetor $\mathbf{k}(n)$ é conhecido. Para tal, subtraindo \mathbf{w}_0 de ambos os lados de (2.1), determinando o produto externo $\mathbf{v}(n)\mathbf{v}^T(n)$, tomando o valor esperado de ambos os lados da expressão resultante e pré- e pós-multiplicando por \mathbf{Q}^T e \mathbf{Q} , respectivamente, obtém-se a seguinte recursão para $\mathbf{k}(n)$ [13]:

$$\mathbf{k}(n+1) = \mathbf{B}\mathbf{k}(n) + \mu^2(N-2)m_u e_{\min} \mathbf{s} \quad (2.29)$$

onde \mathbf{B} é uma matriz cujos elementos são dados por

$$b_{i,j} = \begin{cases} \mu^2 m_{ij}, & i \neq j \\ 1 - 2\mu h_i + 3\mu^2 \ell_i, & i = j \end{cases} \quad (2.30)$$

com

$$\ell_i = \int_0^\infty \int_0^\infty \frac{\lambda_i^2 d\omega_1 d\omega_2}{(1 + 2\omega_1 \lambda_i)^2 \sqrt{\prod_{k=1}^N (1 + 2\omega_1 \lambda_k)}}, \quad (2.31)$$

e

$$m_{i,j} = \int_0^\infty \int_0^\infty \frac{\lambda_i \lambda_j d\omega_1 d\omega_2}{(1 + 2\omega_1 \lambda_i)(1 + 2\omega_1 \lambda_j) \sqrt{\prod_{k=1}^N (1 + 2\omega_1 \lambda_k)}}. \quad (2.32)$$

O i -ésimo elemento do vetor \mathbf{s} é

$$s_i = \int_0^\infty \int_0^\infty \frac{\lambda_i}{\sqrt{\prod_{k=1}^N (1 + 2\omega_1 \lambda_k)} (1 + 2\omega_1 \lambda_i)} d\omega_1 d\omega_2 \quad (2.33)$$

e

$$m_u = E \left[\frac{1}{\mathbf{u}^T(n)\mathbf{u}(n)} \right] \quad (2.34)$$

onde $\mathbf{x}(n) = \mathbf{J}\mathbf{u}(n)$ [13]. A matriz \mathbf{J} é tal que $\mathbf{R} = \mathbf{J}\mathbf{J}^T$, obtida por fatoração de Cholesky. Para determinar (2.29), é requerido o cômputo de várias integrais hiperelípticas duplas [veja (2.31), (2.32) e (2.33)]. Dessa forma, aplica-se o procedimento anteriormente proposto [veja (i)-(iii) da Seção 2.4]. Assim, considerando que todas as raízes são reais e distintas, a expansão em frações parciais dos integrandos de (2.31), (2.32) e (2.33) nos permite obter uma solução fechada para tais integrais. Então, nós obtemos, respectivamente,

$$\ell_i = \frac{1}{4\sqrt{a_N}} \left\{ \sum_{q=1}^{N/2} A_{i,q} \left\{ \frac{\lambda'_q}{\lambda_i - \lambda'_q} [1 + \ln(\lambda_i)] + \frac{\lambda_i}{\lambda_i - \lambda'_q} [1 + \ln(\lambda'_q)] \right\} + b_i \ln(\lambda_i) \right\}, \quad (2.35)$$

$$m_{i,j} = \frac{1}{4\sqrt{a_N}} \left\{ \sum_{q=1}^{N/2} C_{i,j,q} [1 + \ln(\lambda'_q)] + c_{i,j} [1 + \ln(\lambda_i)] + c_{j,i} [1 + \ln(\lambda_j)] \right\} \quad (2.36)$$

e

$$s_i = \frac{-1}{4\sqrt{a_N}} \left\{ \sum_{q=1}^{N/2} A_{i,q} \left[\frac{1}{\lambda'_q} + \frac{1}{\lambda'_q} \ln(\lambda'_q) \right] + B_i \left[\frac{1}{\lambda_i} + \frac{1}{\lambda_i} \ln(\lambda_i) \right] \right\} \quad (2.37)$$

para

$$C_{i,j,q} = \frac{\lambda'_q \lambda_j \lambda_i}{(\lambda_i - \lambda'_q)(\lambda'_q - \lambda_j)} \prod_{\substack{k=1 \\ k \neq q}}^{N/2} \frac{\lambda'_q \lambda'_k}{\lambda'_q - \lambda'_k} \quad (2.38)$$

e

$$c_{i,j} = \frac{\lambda_j}{\lambda_j - \lambda_i} \prod_{q=1}^{N/2} \frac{\lambda'_q \lambda_i}{\lambda_i - \lambda'_q} \quad (2.39)$$

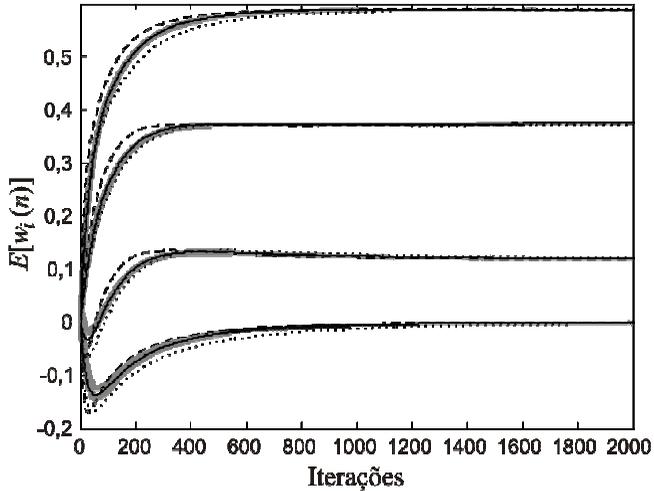
2.5. RESULTADOS EXPERIMENTAIS

Considerando um problema de identificação de sistemas, alguns exemplos são mostrados para confirmar a precisão do modelo proposto. A planta \mathbf{w}_p usada é obtida a partir de uma janela de Hanning normalizada, tal que $\|\mathbf{w}_p\|=1$. O sinal de entrada é gaussiano e correlacionado, obtido de um processo AR(2) dado por

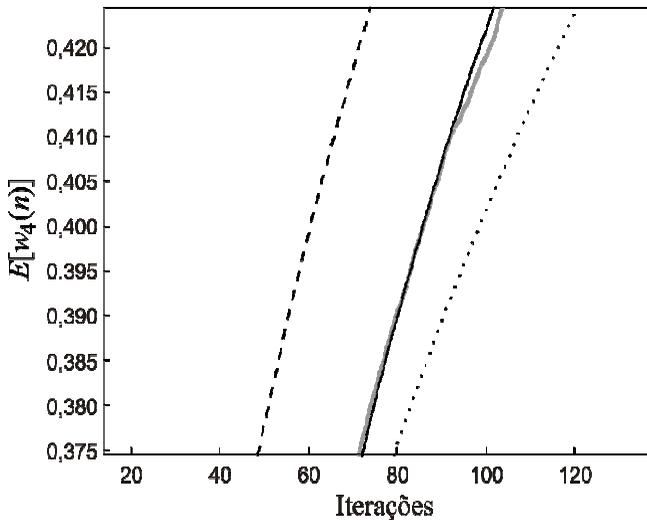
$$x(n) = a_1x(n-1) + a_2x(n-2) + v(n), \quad (2.40)$$

onde $v(n)$ é um ruído branco com variância σ_v^2 . Os pesos do processo AR são $a_1 = 0,61$ e $a_2 = -0,85$, resultando em um sinal de entrada correlacionado com $\chi = 74$ (χ denota a dispersão dos autovalores da matriz de autocorrelação do sinal de entrada). O ruído de medição $z(n)$ tem variância $\sigma_z^2 = 10^{-4}$ (SNR = 40 dB). Para todos os exemplos, o parâmetro de controle do passo de adaptação é $0,1\mu_{\max}$, sendo que a máxima velocidade de convergência ocorre quando $\mu_{\max} = 1$.

A Figura 2.2 mostra o comportamento médio do vetor de pesos obtido por simulações de MC como também através dos seguintes modelos: PM/HI [14], [15], PM/AIH [16] e o proposto neste trabalho. A planta utilizada é um vetor com 8 elementos (janela de Hanning normalizada) dado por $\mathbf{w}_p = [0,000 \ 0,1162 \ 0,3773 \ 0,5867 \ 0,5867 \ 0,3773 \ 0,1162 \ 0,000]^T$. Através dos resultados apresentados, verifica-se a precisão obtida usando o modelo proposto, particularmente durante o regime transiente. Para evidenciar melhor a precisão do modelo proposto, a Figura 2.2(b) apresenta, em detalhes, o regime transiente para o peso $w_4(n)$, cujo valor em regime permanente é 0,5867.



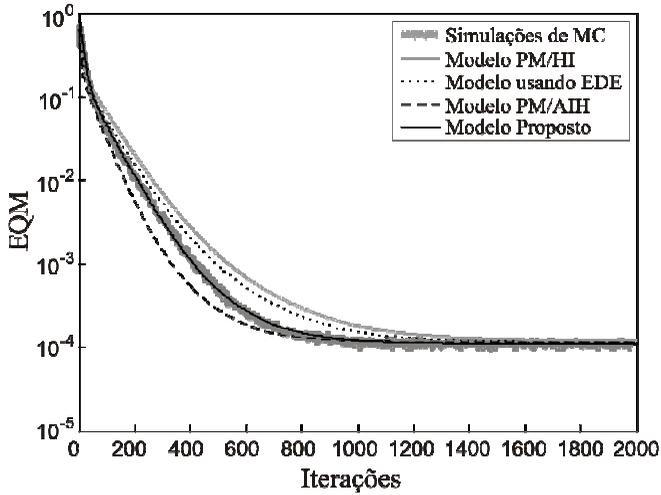
(a)



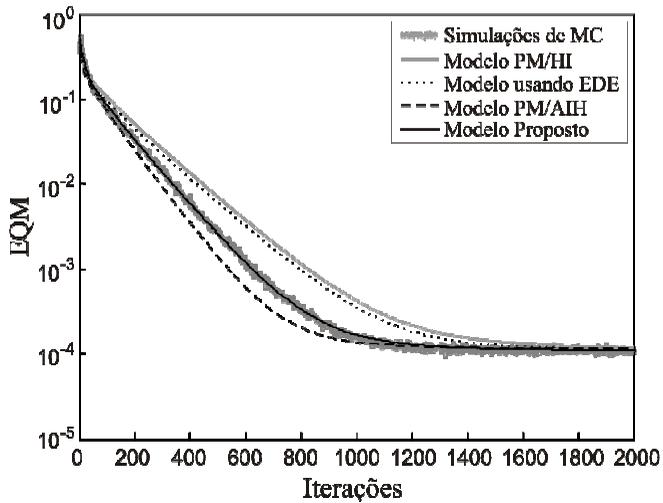
(b)

Figura 2.2. Comportamento médio dos pesos para sinal de entrada gaussiano e correlacionado $\chi = 74$, para uma planta de comprimento $N = 8$. (Linha cinza) Simulações de MC. (Linha tracejada) modelo PM/HI [14], [15]. (Linha pontilhada) modelo PM/AIH [16]. (Linha escura sólida) modelo proposto. (a) Curvas dos pesos. (b) Detalhes do regime transiente do peso $w_4(n)$.

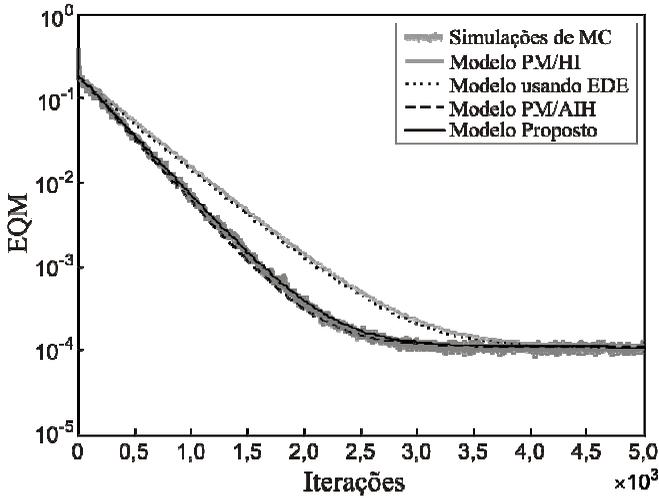
Para avaliar a dependência do modelo em relação ao número de pesos, a Figura 2.3 mostra diversas curvas de aprendizagem utilizando plantas e filtros adaptativos com 6, 8, 16 e 64 pesos, considerando sinais de entrada correlacionados com dispersão de autovalores χ iguais a 57, 74, 135 e 254, respectivamente. Em todos os casos, são apresentados os resultados obtidos por simulações de MC (média de 500 rodadas independentes), considerando o modelo PM/HI [14], [15], modelo baseado em EDE [17], modelo PM/AIH [16] e o modelo proposto. A partir dessas curvas, é novamente verificada uma precisão muito boa do modelo proposto para todos os comprimentos de filtro e dispersões de autovalores considerados. Note que quanto maior for o número de pesos, mais o modelo PM/AIH [17] aproxima-se do modelo proposto para sinais de entrada gaussianos [veja Figura 2.3(d)].



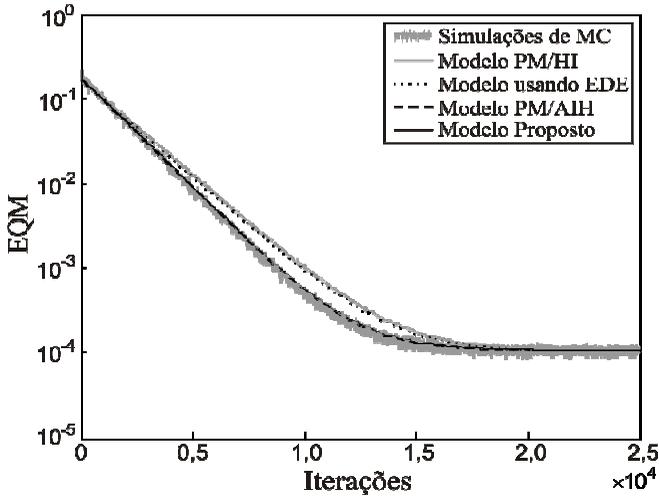
(a)



(b)



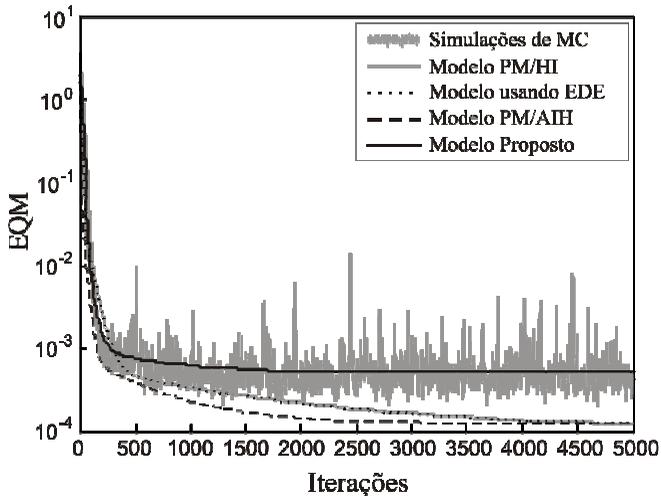
(c)



(d)

Figura 2.3. Curva de aprendizagem (EQM) para sinal de entrada gaussiano e correlacionado: resultados de simulações e modelos. (a) $N = 6$ e $\chi = 57$. (b) $N = 8$ e $\chi = 74$. (c) $N = 16$ e $\chi = 135$. (d) $N = 64$ e $\chi = 254$.

Na Figura 2.4, são apresentadas comparações entre simulações de MC, modelo PM/HI [14], [15], modelo baseado em EDE [17], modelo PM/AIH [16] e modelo proposto, considerando três sinais SIRP correlacionados com funções de densidade de probabilidade marginal laplaciana, K_0 e Gama generalizada. Os processos SIRP são gerados como descrito em [30], tendo média zero e variância unitária. Os sinais correlacionados são obtidos através de um processo AR(2) com coeficientes $a_1=1,67$ e $a_2=-0,85$. A dispersão dos autovalores é $\chi=883$. A variável m_u é obtida conforme discutida em [13]. Para os três casos apresentados, é utilizada a planta \mathbf{w}_p , obtendo novamente uma precisão muito boa quando utilizado o modelo proposto tanto para o regime transiente quanto para o regime permanente. Note que os modelos PM/HI [14], [15], baseado em EDE [17] e PM/AIH [16] não consideram a não-gaussianidade do processo de entrada, falhando assim na previsão do comportamento do algoritmo.



(a)

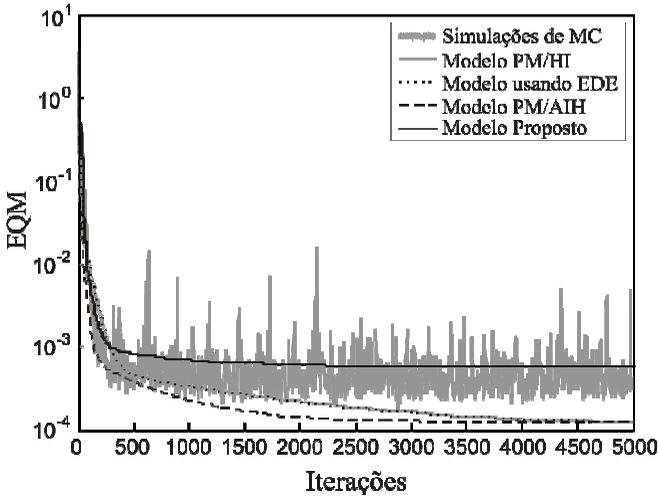
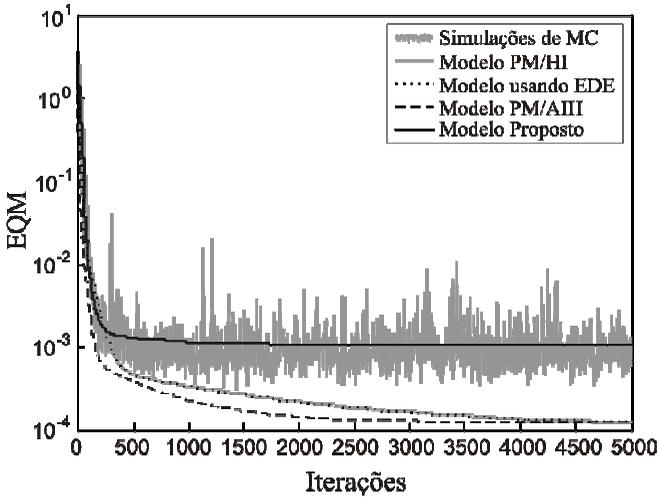


Figura 2.4. Curva de aprendizagem (EQM) para sinal de entrada SIRP correlacionado com função densidade de probabilidade marginal: (a) K_0 . (b) Função Gama generalizada. (c) Laplaciana.

2.6. CONCLUSÕES E COMENTÁRIOS FINAIS

Neste capítulo, foi apresentado um modelo estocástico para o algoritmo LMS normalizado assumindo apenas uma hipótese simplificativa: independência entre os pesos do filtro adaptativo e as amostras do sinal de entrada. Em função do procedimento de análise, o modelo obtido é mais complexo do que outros apresentados na literatura, porém muito mais preciso. Essa precisão pode ser atribuída ao fato que o modelo proposto acompanha a redução na dispersão dos autovalores das matrizes envolvidas, característica que não pode ser reproduzida se utilizado o PM.

A determinação das expressões do modelo proposto requer o cálculo de N integrais hiperelípticas simples, para o momento de primeira ordem, e $N^2 + N$ integrais hiperelípticas duplas para o momento de segunda ordem. Essas integrais podem ser calculadas a partir das expressões fechadas propostas.

O presente trabalho fornece, adicionalmente, novas ferramentas de análise para serem utilizadas na modelagem de outros algoritmos tipo NLMS, como também serve para melhorar a precisão de modelos existentes obtidos através de outras estratégias, tais como os baseados em EDE [17] e na conservação de energia [1], visto que tais modelos também requerem o cálculo de valores esperados similares.

3.1. INTRODUÇÃO

O algoritmo LMS no domínio transformado (LMS-DT) foi proposto por Narayan et al. [20] objetivando melhorar as características de convergência do algoritmo LMS padrão. O algoritmo LMS-DT é similar ao LMS, porém tendo seu sinal de entrada pré-processado por uma transformação ortogonal com N sub-bandas, seguido de um processo de normalização que equaliza o conteúdo energético de cada sub-banda. Na prática, para implementar o processo de normalização, são necessárias estimativas da potência do sinal de entrada em cada sub-banda. Para tal, é utilizada uma medida da variância do sinal de entrada através de uma janela deslizante de comprimento fixo (JDCF).

Com vistas à modelagem, devido ao processo de normalização, uma importante dificuldade encontrada é o cálculo de valores esperados tais como

$$E[\hat{\mathbf{D}}^{-1}(n)\hat{\mathbf{R}}(n)] \quad (3.1)$$

onde $\hat{\mathbf{D}}(n)$ e $\hat{\mathbf{R}}(n)$ são matrizes que dependem dos dados observados. Para calcular (3.1), várias aproximações simplificativas são geralmente consideradas [21]-[24]. Em particular, os resultados mais precisos são obtidos aplicando o método apresentado em [24], no qual o Princípio da Média (PM) é utilizado para separar (3.1) em dois valores esperados. Assim,

$$E[\hat{\mathbf{D}}^{-1}(n)\hat{\mathbf{R}}(n)] \cong E[\hat{\mathbf{D}}^{-1}(n)]E[\hat{\mathbf{R}}(n)] \quad (3.2)$$

onde $E[\hat{\mathbf{R}}(n)]$ é facilmente determinado enquanto $E[\hat{\mathbf{D}}^{-1}(n)]$ é obtido calculando de forma aproximada as integrais que surgem da operação para obter o valor esperado. O PM resulta em expressões simples, apresentando precisão satisfatória se grandes janelas de observação são utilizadas. No entanto, tal condição não é comumente encontrada em situações práticas, pois implica aumento de custo computacional. Assim,

procurando por um modelo mais preciso, independentemente do comprimento da janela de observação, o uso do PM não é o mais recomendado.

Neste capítulo, é proposto um modelo que considera os seguintes pontos:

- i) Valores esperados semelhantes a (3.1) são calculados sem invocar o PM, fazendo com que o modelo proposto seja muito mais preciso do que outros encontrados na literatura.
- ii) O modelo proposto é derivado considerando ambientes não-estacionários (planta variante no tempo) por duas razões: tem maior importância prática; e o caso estacionário pode ser obtido diretamente a partir das expressões propostas.
- iii) Expressões para o erro em excesso, desajuste e passo de adaptação para o mínimo erro em excesso são também derivadas.
- iv) Um modelo simplificado considerando um ambiente não estacionário é proposto e seus limites de aplicabilidade são mostrados e discutidos.

Este capítulo é organizado com segue. A Seção 3.2 descreve o modelo utilizado para caracterizar a planta variante no tempo. A Seção 3.3 é dedicada à derivação do modelo proposto para os momentos de primeira e segunda ordens do vetor de pesos e a curva de aprendizagem. Na Seção 3.4 são derivadas expressões para o erro em excesso, passo de adaptação para o mínimo erro em excesso, desajuste e também são relacionados o grau de não-estacionaridade e o desajuste. A Seção 3.5 apresenta modelos simplificados que descrevem o comportamento do algoritmo LMS-DT. Na Seção 3.6, resultados de simulações numéricas são mostrados. Finalmente, a Seção 3.7 apresenta as conclusões do capítulo.

3.2. AMBIENTE NÃO-ESTACIONÁRIO

Nesta seção, a formulação básica do algoritmo LMS-DT para sua operação em um ambiente não-estacionário é introduzida. Para realizar uma análise sob tal condição, é usual assumir que os sinais envolvidos variam de acordo com um dado modelo matemático. Dessa forma, o sinal desejado $d(n)$ é modelado por

$$d(n) = y^o(n) + z(n) = \mathbf{x}_T^T(n) \mathbf{w}_T^o(n) + z(n) \quad (3.3)$$

onde $y^o(n)$ é a saída da planta, $\mathbf{x}_T(n) = \mathbf{T}\mathbf{x}(n) = [x_{T,0}(n) \ x_{T,1}(n) \ \dots \ x_{T,N-1}(n)]^T$ é o vetor do sinal de entrada no domínio transformado, \mathbf{T} denota uma matriz de transformação ortogonal, $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-N+1)]^T$ é o vetor do sinal de entrada, assumindo $\{x(n)\}$ um processo estacionário gaussiano de média zero e variância σ_x^2 , $\mathbf{w}^o(n) = [w_1^o(n) \ w_2^o(n) \ \dots \ w_{N-1}^o(n)]^T$ é o vetor da planta variante no tempo e $\mathbf{w}_T^o(n) = \mathbf{T}\mathbf{w}^o(n) = [w_{T,1}^o(n) \ w_{T,2}^o(n) \ \dots \ w_{T,N-1}^o(n)]^T$, o vetor da planta no domínio transformado. O ruído de medição é denotado por $z(n)$ e tem média zero e variância σ_z^2 .

Finalmente, a lei que descreve a evolução da planta é

$$\mathbf{w}_T^o(n+1) = a\mathbf{w}_T^o(n) + \boldsymbol{\eta}(n) \quad (3.4)$$

onde $\boldsymbol{\eta}(n)$ caracteriza o processo de perturbação da planta, tendo média zero e matriz de autocorrelação $\boldsymbol{\Phi}$, a variável a pode tomar qualquer valor entre 0 e 1, e $\mathbf{w}_T^o(0)$, inicialização do vetor da planta variante podendo tomar um valor arbitrário.

Usualmente, análises de algoritmos adaptativos em ambientes não-estacionários disponíveis na literatura consideram $a=1$ e/ou $\mathbf{w}_T^o(0) = \mathbf{0}$ [31], [32]. Assim, para tal situação, tem-se um modelo restrito, sendo que o valor médio da planta não se modifica ao longo do tempo. Além do mais, o modelo dado em (3.4) para $a=1$ é pouco aplicável, visto que representa um sistema instável (com crescimento irrestrito da variância) [31]. Neste trabalho, as derivações obtidas consideram um valor genérico para a , resultando em um modelo mais realista e geral.

3.3. MODELO PROPOSTO

Nesta seção, são derivadas expressões analíticas para os momentos de primeira e segunda ordens do vetor de pesos adaptativos. Para tal, iniciamos a análise considerando a equação de atualização do vetor de pesos no domínio transformado dada por

$$\mathbf{w}_T(n+1) = \mathbf{w}_T(n) + 2\mu\hat{\mathbf{D}}^{-1}(n)e(n)\mathbf{x}_T(n) \quad (3.5)$$

onde $\mathbf{w}_T(n) = [w_{T,0}(n) \ w_{T,1}(n) \ \cdots \ w_{T,N-1}(n)]^T$ é o vetor de pesos do filtro adaptativo,

$$e(n) = d(n) - y(n) \quad (3.6)$$

com $y(n) = \mathbf{x}_T^T(n) \mathbf{w}_T(n)$, caracteriza o erro de estimação, μ representa o passo de adaptação e $\hat{\mathbf{D}}(n) = \text{diag}[\hat{\sigma}_0^2(n) \ \hat{\sigma}_1^2(n) \ \cdots \ \hat{\sigma}_{N-1}^2(n)]$, a matriz de normalização do passo de adaptação, com $\hat{\sigma}_i^2(n)$ sendo a estimativa instantânea da variância da i -ésima sub-banda. Na prática, uma JDCF é utilizada para estimar a variância de cada sub-banda. Assim,

$$\hat{\sigma}_i^2(n) = \frac{1}{M} \mathbf{x}_{T,i}^T(n) \mathbf{x}_{T,i}(n) \quad (3.7)$$

com

$$\mathbf{x}_{T,i}(n) = [x_{T,i}(n) \ x_{T,i}(n-1) \ \cdots \ x_{T,i}(n-M+1)]^T \quad (3.8)$$

sendo o vetor de sinal de entrada transformado para a i -ésima sub-banda e M denotando o comprimento da janela de observação.

Agora, substituindo (3.3) em (3.6) e a expressão resultante em (3.5), obtém-se

$$\mathbf{w}_T(n+1) = \mathbf{w}_T(n) + 2\mu \hat{\mathbf{D}}^{-1}(n) \mathbf{x}_T(n) [\mathbf{x}_T^T(n) \mathbf{w}_T^o(n) + z(n) - \mathbf{x}_T^T(n) \mathbf{w}_T(n)]. \quad (3.9)$$

Utilizando o vetor de erro nos pesos no domínio transformado, definido como $\mathbf{v}_T(n) = \mathbf{w}_T(n) - \mathbf{w}_T^o(n)$, (3.9) é reescrita como

$$\begin{aligned} \mathbf{v}_T(n+1) = & [\mathbf{I} - 2\mu \hat{\mathbf{D}}^{-1}(n) \mathbf{x}_T(n) \mathbf{x}_T^T(n)] \mathbf{v}_T(n) + \\ & + 2\mu \hat{\mathbf{D}}^{-1}(n) \mathbf{x}_T(n) z(n) - \mathbf{w}_T^o(n+1) + \mathbf{w}_T^o(n) \end{aligned} \quad (3.10)$$

onde \mathbf{I} denota a matriz identidade. Finalmente, substituindo (3.4) em (3.10), a equação de atualização em termos do $\mathbf{v}_T(n)$ é dada por

$$\begin{aligned} \mathbf{v}_T(n+1) = & [\mathbf{I} - 2\mu\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)]\mathbf{v}_T(n) + \\ & + 2\mu\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z(n) + (1-a)\mathbf{w}_T^o(n) - \boldsymbol{\eta}(n). \end{aligned} \quad (3.11)$$

O passo seguinte é então determinar os momentos de primeira e segunda ordens de (3.11).

3.3.1. CONSIDERAÇÕES DE ANÁLISE

Antes de dar continuidade ao desenvolvimento em questão, as seguintes considerações são assumidas:

- i) $\boldsymbol{\eta}(n)$ e $\boldsymbol{\eta}(m)$, para $m \neq n$, são decorrelacionados.
- ii) $\boldsymbol{\eta}(n)$, $\mathbf{v}_T(n)$ e $\mathbf{x}_T(n)$ são estatisticamente independentes entre si.
- iii) $z(n)$ é decorrelacionado com qualquer outro sinal no sistema.
- iv) O vetor $\mathbf{w}_T(n)$ é inicializado para $n = 0$, isto é

$$\mathbf{w}_T(0) = \mathbf{w}_T^o(n) \Big|_{n=0} \quad (3.12)$$

com $\mathbf{w}_T^o(0)$ tomando um valor arbitrário.

3.3.2. MOMENTO DE PRIMEIRA ORDEM DE $\mathbf{v}_T(n)$

Tomando o valor esperado de ambos os lados de (3.11) e levando em conta as considerações (i)-(iv), obtém-se

$$E[\mathbf{v}_T(n+1)] \cong \{\mathbf{I} - 2\mu \underbrace{E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)]}_{\mathbf{P}}\}E[\mathbf{v}_T(n)] + (1-a)a^n\mathbf{w}_T^o(0) \quad (3.13)$$

onde o último termo do lado direito representa a média do processo AR(1) dado por (3.4). Agora, o ponto chave é determinar a matriz \mathbf{P} , cujos elementos são dados por

$$p(i, j) = E \left[\frac{x_{T,i}(n)x_{T,j}(n)}{\hat{\sigma}_i^2(n)} \right]. \quad (3.14)$$

Assim, utilizando a definição de valor esperado [37], (3.14) é calculada como segue:

$$p(i, j) = M \underbrace{\int \cdots \int}_{M+1 \text{ integrais}} \frac{x_{T,i}(n)x_{T,j}(n)}{\sum_{k=0}^{M-1} x_{T,i}^2(n-k)} f_{\mathbf{x}}(\mathbf{x}_{i,j}) d\mathbf{x}_{i,j} \quad (3.15)$$

onde $f_{\mathbf{x}}(\mathbf{x}_{i,j})$ representa a função densidade de probabilidade conjunta para sinais gaussianos do vetor $\mathbf{x}_{i,j}(n) = [x_{T,i}(n) x_{T,i}(n-1) \cdots x_{T,i}(n-M+1) x_{T,j}(n)]^T$, dada por

$$f_{\mathbf{x}}(\mathbf{x}_{i,j}) = \frac{1}{\sqrt{(2\pi)^{M+1} \det(\mathbf{R}_{i,j})}} e^{-\mathbf{x}_{i,j}^T(n) \mathbf{R}_{i,j}^{-1} \mathbf{x}_{i,j}(n)/2} \quad (3.16)$$

onde $\mathbf{R}_{i,j} = E[\mathbf{x}_{i,j}(n)\mathbf{x}_{i,j}^T(n)]$ e a matriz de autocorrelação do vetor $\mathbf{x}_{i,j}(n)$. O Apêndice 1 apresenta em detalhes o procedimento seguido para determinar (3.15), resultando em

$$p(i, j) = M \mathbf{q}_{T,i}^T \mathbf{H}_i \mathbf{Q}_{T,i}^{-1} \mathbf{r}_{i,j} \quad (3.17)$$

onde $\mathbf{Q}_{T,i}$ representa a matriz de autovetores de $\mathbf{R}_{T,i} = E[\mathbf{x}_{T,i}(n)\mathbf{x}_{T,i}^T(n)]$, com $\mathbf{x}_{T,i}(n)$ definido em (3.8), $\mathbf{q}_{T,i}$ denota o vetor que contém a primeira linha de $\mathbf{Q}_{T,i}$, $\mathbf{r}_{i,j} = E[\mathbf{x}_{T,i}(n)x_{T,j}(n)]$ é o vetor de correlação cruzada entre sub-bandas, e \mathbf{H}_i , uma matriz diagonal com elementos dados por

$$h_i(l, l) \cong \frac{1}{2\lambda_{i,l}\sqrt{a_i}} \left[\sum_{q=1}^{M/2} A_{i,l,q} \ln(\lambda'_{i,q}) + B_{i,l} \ln(\lambda_{i,l}) \right] \quad (3.18)$$

com

$$a_i = \prod_{k=1}^M \lambda_{i,k}, \quad (3.19)$$

$$A_{i,l,q} = \frac{\lambda'_{i,q}{}^{M/2} \lambda_{i,l}}{\lambda'_{i,q} - \lambda_{i,l}} \prod_{\substack{j=1 \\ j \neq q}}^{M/2} \frac{\lambda'_{i,j}}{\lambda'_{i,q} - \lambda'_{i,j}} \quad (3.20)$$

e

$$B_{i,l} = \prod_{q=1}^{M/2} \frac{\lambda_{i,l} \lambda'_{i,q}}{\lambda_{i,l} - \lambda'_{i,q}} \quad (3.21)$$

onde $\lambda_{i,k}$, para $k=1, \dots, M$, são os autovalores de $\mathbf{R}_{T,i}$ e $\lambda'_{i,q} = \sqrt{\lambda_{i,2q-1} \lambda_{i,2q}}$ para $q=1, 2, \dots, M/2$ com M par.

Agora, substituindo (3.17) em (3.13) e expressando o resultado em forma escalar, obtém-se

$$E[v_{T,i}(n+1)] = \sum_{j=1}^N (1 - 2\mu M \mathbf{q}_{T,i}^T \mathbf{H}_i \mathbf{Q}_{T,i}^{-1} \mathbf{r}_{i,j}) E[v_{T,j}(n)] + (1-a) a^n w_{T,i}^0(0) \quad (3.22)$$

onde $v_{T,i}(n)$ denota o i -ésimo elemento do vetor $\mathbf{v}_T(n)$.

Na Seção 3.6.1.1 (Exemplo 1), a precisão de (3.22) é verificada. Também, a Seção 3.5 apresenta um modelo simplificado obtido utilizando algumas considerações adicionais às (i)-(iv), apresentadas na Seção 3.3.1.

3.3.3. MOMENTO DE SEGUNDA ORDEM DE $\mathbf{v}_T(n)$

O momento de segunda ordem do vetor de erro nos pesos no domínio transformado é obtido a partir de $\mathbf{K}(n) = E[\mathbf{v}_T(n) \mathbf{v}_T^T(n)]$. Assim, transpondo ambos os lados de (3.11), efetuando o produto $\mathbf{v}_T(n) \mathbf{v}_T^T(n)$ e tomando o valor esperado de ambos os lados da expressão resultante, obtém-se

$$\begin{aligned}
E[\mathbf{v}_T(n+1)\mathbf{v}_T^T(n+1)] &= E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)] + \overbrace{(1-a) E[\mathbf{v}_T(n)\mathbf{w}_T^{\circ T}(n)] - E[\mathbf{v}_T(n)\boldsymbol{\eta}^T(n)]}^{E1} \\
&\quad - \overbrace{2\mu(1-a) E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{w}_T^{\circ T}(n)]}^{E2} \\
&\quad - \overbrace{2\mu(1-a) E[\mathbf{w}_T^{\circ}(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]}^{E3} \\
&\quad + \overbrace{2\mu E[\mathbf{v}_T(n)z(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] - 2\mu E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]}^{E4} \\
&\quad - \overbrace{2\mu E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{v}_T^T(n)]}^{E5} \\
&\quad - 4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}(n)z(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \\
&\quad + 2\mu E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\boldsymbol{\eta}^T(n)] \\
&\quad + 2\mu E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z(n)\mathbf{v}_T^T(n)] \\
&\quad - 4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \\
&\quad + \overbrace{4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]}^{E6} \\
&\quad + \overbrace{4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z^2(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]}^{E7} \\
&\quad + 2\mu(1-a)E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z(n)\mathbf{w}_T^{\circ T}(n)] \\
&\quad - 2\mu E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z(n)\boldsymbol{\eta}^T(n)] \\
&\quad + (1-a)E[\mathbf{w}_T^{\circ}(n)\mathbf{v}_T^T(n)] \\
&\quad + 2\mu(1-a)E[\mathbf{w}_T^{\circ}(n)z(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] + (1-a)^2 E[\mathbf{w}_T^{\circ}(n)\mathbf{w}_T^{\circ T}(n)] \\
&\quad - (1-a)E[\mathbf{w}_T^{\circ}(n)\boldsymbol{\eta}^T(n)] - E[\boldsymbol{\eta}(n)\mathbf{v}_T^T(n)] \\
&\quad + 2\mu E[\boldsymbol{\eta}(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] - (1-a)E[\boldsymbol{\eta}(n)\mathbf{w}_T^{\circ T}(n)] \\
&\quad - 2\mu E[\boldsymbol{\eta}(n)z(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] + E[\boldsymbol{\eta}(n)\boldsymbol{\eta}^T(n)].
\end{aligned} \tag{3.23}$$

3.3.3.1 Aproximações para determinar (3.23)

Os valores esperados do lado direito de (3.23) são determinados utilizando as considerações (i)-(iv) da Seção 3.3.1. Assim, das características de $z(n)$ e $\boldsymbol{\eta}(n)$, todos os termos que contêm tais variáveis são iguais a zero, com exceção daqueles com $z^2(n)$ e $\boldsymbol{\eta}^T(n)\boldsymbol{\eta}(n)$. Também, podem ser obtidas as seguintes aproximações:

Considerando que $\mathbf{v}_T(n)$ e $\mathbf{w}_T^o(n)$ são descorrelacionados, obtém-se

$$E1) E[\mathbf{v}_T(n)\mathbf{w}_T^{oT}(n)] \simeq E[\mathbf{v}_T(n)]E[\mathbf{w}_T^{oT}(n)], \quad (3.24)$$

$$E2) E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{w}_T^{oT}(n)] \cong \mathbf{P}E[\mathbf{v}_T(n)]E[\mathbf{w}_T^{oT}(n)] \quad (3.25)$$

e

$$E3) E[\mathbf{w}_T^o(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \cong E[\mathbf{w}_T^o(n)]E[\mathbf{v}_T^T(n)]\mathbf{P}^T. \quad (3.26)$$

Também, utilizando a consideração (ii), os valores esperados E4 e E5 podem ser aproximados por

$$E4) E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \cong E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)]E[\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \\ = \mathbf{K}(n)\mathbf{P}^T \quad (3.27)$$

e

$$E5) E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{v}_T^T(n)] \cong E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)]E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)] \\ = \mathbf{P}\mathbf{K}(n). \quad (3.28)$$

Considerando que os sinais envolvidos em (E6) são conjuntamente gaussianos [9] (isso torna-se um fato quando o valor de M é grande), obtém-se

$$E6) E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] \\ \cong 2\mathbf{P}\mathbf{K}(n)\mathbf{P}^T + \text{Str}[\mathbf{K}(n)\mathbf{R}_T] \quad (3.29)$$

com

$$\mathbf{S} = E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]. \quad (3.30)$$

Na Seção 3.4.5 é apresentada uma verificação das aproximações (3.24)-(3.29). No Apêndice 2, um método aproximado para calcular \mathbf{S} é apresentado. Na literatura, tal matriz é comumente calculada utilizando o PM para todos os seus elementos; diferentemente aqui, o cálculo dos

elementos da diagonal principal de \mathbf{S} é feito sem utilizar essa aproximação. O PM é reservado para os elementos fora da diagonal principal, sendo que eles têm menos importância na modelagem. Fazendo assim, \mathbf{S} é determinada de forma mais precisa do que utilizando puramente o PM para obter todos os seus elementos (veja Seção 3.6.3), porém, é preciso o cálculo de N integrais hiperelípticas.

Finalmente, considerando as características de $z(n)$, (E7) resulta em

$$\begin{aligned} \text{E7) } E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)z^2(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)] &= \\ &= E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]E[z^2(n)] = \mathbf{S}\sigma_z^2. \end{aligned} \quad (3.31)$$

Assim, pode-se reescrever (3.23) como segue:

$$\begin{aligned} \mathbf{K}(n+1) &= \mathbf{K}(n) - 2\mu\mathbf{K}(n)\mathbf{P}^T - 2\mu\mathbf{P}\mathbf{K}(n) + 4\mu^2\mathbf{S}\sigma_z^2 + (1-a)^2\mathbf{K}^o(n) \\ &\quad + (1-a)E[\mathbf{w}_T^o(n)]E[\mathbf{v}_T^T(n)](\mathbf{I} - 2\mu\mathbf{P}^T) + 8\mu^2\mathbf{P}\mathbf{K}(n)\mathbf{P}^T + \mathbf{\Phi} \\ &\quad + (1-a)(\mathbf{I} - 2\mu\mathbf{P})E[\mathbf{v}_T(n)]E[\mathbf{w}_T^{oT}(n)] + 4\mu^2\mathbf{S}\text{tr}[\mathbf{K}(n)\mathbf{R}_T] \end{aligned} \quad (3.32)$$

com

$$\mathbf{K}^o(n) = E[\mathbf{w}_T^o(n)\mathbf{w}_T^{oT}(n)] = (a^2)^n\mathbf{K}^o(0) + \frac{1-(a^2)^n}{1-a^2}\mathbf{\Phi}. \quad (3.33)$$

A expressão (3.33) é obtida transpondo ambos os lados de (3.4), efetuando o produto $\mathbf{w}_T^o(n)\mathbf{w}_T^{oT}(n)$, aplicando o operador valor esperado a ambos os lados da expressão resultante e utilizando (iv) da Seção 3.3.1. Assim,

$$\mathbf{K}^o(0) = E[\mathbf{w}_T^o(0)\mathbf{w}_T^{oT}(0)]. \quad (3.34)$$

Note que a matriz $\mathbf{\Phi}$ pode ser uma matriz cheia. Por outro lado, na literatura [31], [32] é usualmente assumida diagonal. Esse fato torna (3.33) um modelo mais abrangente.

3.3.4. CURVA DE APRENDIZAGEM

Nesta seção, uma expressão para a curva de aprendizagem é derivada. Substituindo (3.3) em (3.6) e considerando $\mathbf{v}_T(n) = \mathbf{w}_T(n) - \mathbf{w}_T^o(n)$, o sinal de erro pode ser escrito como

$$e(n) = \mathbf{x}_T^T(n) \mathbf{w}_T^o(n) + z(n) - \mathbf{x}_T^T(n) [\mathbf{v}_T(n) + \mathbf{w}_T^o(n)] = z(n) - \mathbf{x}_T^T(n) \mathbf{v}_T(n). \quad (3.35)$$

Elevando ao quadrado (3.35), aplicando valor esperado a ambos os lados da expressão resultante e considerando as propriedades de independência de $z(n)$, tem-se

$$E[e^2(n)] = E[z^2(n)] + E[\mathbf{v}_T^T(n) \mathbf{x}_T(n) \mathbf{x}_T^T(n) \mathbf{v}_T(n)]. \quad (3.36)$$

O segundo termo do lado direito em (3.36) é manipulado como segue:

$$\begin{aligned} E[\mathbf{v}_T^T(n) \mathbf{x}_T(n) \mathbf{x}_T^T(n) \mathbf{v}_T(n)] &= \text{tr} \left\{ E[\mathbf{v}_T(n) \mathbf{v}_T^T(n) \mathbf{x}_T(n) \mathbf{x}_T^T(n)] \right\} \\ &= \text{tr} \left\{ \mathbf{R}_T E[\mathbf{v}_T(n) \mathbf{v}_T^T(n)] \right\} \end{aligned} \quad (3.37)$$

onde $\text{tr}(\cdot)$ é o operador traço. Assim, definindo $\mathbf{K}(n) = E[\mathbf{v}_T(n) \mathbf{v}_T^T(n)]$ e substituindo (3.37) em (3.36), a curva de aprendizagem é dada por

$$E[e^2(n)] = \sigma_z^2 + \text{tr}[\mathbf{R}_T \mathbf{K}(n)]. \quad (3.38)$$

Finalmente, utilizando (3.32) e (3.38), a curva de aprendizagem para cada instante n pode então ser determinada. Assim,

$$\begin{aligned} E[e^2(n)] &= \sigma_z^2 + \text{tr} \left\{ \mathbf{R}_T \mathbf{K}(n-1) - 2\mu \mathbf{R}_T \mathbf{K}(n-1) \mathbf{P}^T + (1-a) \mathbf{R}_T E[\mathbf{v}_T(n-1)] E[\mathbf{w}_T^o(n-1)] \right. \\ &\quad + (1-a) \mathbf{R}_T E[\mathbf{w}_T^o(n-1)] E[\mathbf{v}_T^T(n-1)] - 2\mu \mathbf{R}_T \mathbf{P} \mathbf{K}(n-1) + 4\mu^2 \mathbf{R}_T \mathbf{S} \sigma_z^2 \\ &\quad + 4\mu^2 \{ 2\mathbf{R}_T \mathbf{P} \mathbf{K}(n-1) \mathbf{P}^T + \mathbf{R}_T \mathbf{S} \text{tr}[\mathbf{K}(n-1) \mathbf{R}_T] \} + (1-a)^2 \mathbf{R}_T \mathbf{K}^o(n-1) + \mathbf{R}_T \Phi \\ &\quad \left. - 2\mu(1-a) \mathbf{R}_T \mathbf{P} E[\mathbf{v}_T(n-1)] E[\mathbf{w}_T^o(n-1)] - 2\mu(1-a) \mathbf{R}_T E[\mathbf{w}_T^o(n-1)] E[\mathbf{v}_T^T(n-1)] \mathbf{P}^T \right\}. \end{aligned} \quad (3.39)$$

3.4. ANÁLISE

Nesta seção, expressões para o erro em excesso, desajuste, passo de adaptação para o mínimo erro em excesso e grau de não-estacionaridade são derivadas e discutidas, determinando como tais parâmetros afetam o comportamento do algoritmo.

3.4.1. ERRO EM EXCESSO DE REGIME PERMANENTE

Por definição, o erro quadrático médio em excesso (EQME) é dado por [9]

$$\xi_{\text{exc}}(n) = E[e^2(n)] - \sigma_z^2 = \text{tr}[\mathbf{R}_T \mathbf{K}(n)]. \quad (3.40)$$

O EQME em regime permanente é obtido fazendo $n \rightarrow \infty$. Assim,

$$\xi_{\text{exc}}(\infty) = \lim_{n \rightarrow \infty} E[\xi_{\text{exc}}(n)] = \lim_{n \rightarrow \infty} E[e^2(n)] - \sigma_z^2 = \text{tr}[\mathbf{R}_T \mathbf{K}(\infty)]. \quad (3.41)$$

Então, de (3.11), efetuando o produto $\hat{\mathbf{D}}(n) \mathbf{v}_T(n+1) \mathbf{v}_T^T(n+1)$, tomando o valor esperado da expressão resultante, fazendo $n \rightarrow \infty$ que resulta em $\mathbf{K}(n+1) = \mathbf{K}(n) = \mathbf{K}(\infty)$ e tomando o traço da matriz resultante, obtém-se

$$\begin{aligned} \xi_{\text{exc}}(\infty) &= \frac{1}{4\mu} (1-a) \text{tr}\{E[\hat{\mathbf{D}}(\infty)]E[\mathbf{v}_T(\infty) \mathbf{w}_T^{\circ T}(\infty)]\} \\ &\quad + \overbrace{\mu \text{tr}\{E[\mathbf{x}_T(\infty) \mathbf{x}_T^T(\infty) \mathbf{v}_T(\infty) \mathbf{v}_T^T(\infty) \mathbf{x}_T(\infty) \mathbf{x}_T^T(\infty) \hat{\mathbf{D}}^{-1}(\infty)]\}}^{\text{ES}} \\ &\quad - \frac{1}{2} (1-a) \text{tr}\{\mathbf{R}_T E[\mathbf{v}_T(\infty) \mathbf{w}_T^{\circ T}(\infty)]\} \\ &\quad + \mu \sigma_z^2 \text{tr}(\mathbf{P}) + \frac{1}{4\mu} (1-a) \text{tr}\{E[\hat{\mathbf{D}}(\infty)]E[\mathbf{w}_T^{\circ}(\infty) \mathbf{v}_T^T(\infty)]\} \\ &\quad - \frac{1}{2} (1-a) \text{tr}\{E[\mathbf{w}_T^{\circ}(\infty) \mathbf{v}_T^T(\infty)] \mathbf{R}_T\} + \frac{1}{4\mu} (1-a)^2 \text{tr}\{E[\hat{\mathbf{D}}(\infty)] \mathbf{K}^{\circ}(\infty)\} \\ &\quad + \frac{1}{4\mu} \text{tr}\{E[\hat{\mathbf{D}}(\infty)] \Phi\}. \end{aligned} \quad (3.42)$$

Em (3.42), a propriedade $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$ e as considerações (i)-(iv) da Seção 3.3.1 são utilizadas. Assim, assumindo que $\lim_{n \rightarrow \infty} E[\mathbf{v}_T(n)] = 0$ e, de (3.33), $\lim_{n \rightarrow \infty} \mathbf{K}^o(n) = \frac{1}{1-a^2} \mathbf{\Phi}$, (3.42) pode ser rescrita como

$$\begin{aligned} \xi_{\text{exc}}(\infty) &= \mu \sigma_z^2 \text{tr}(\mathbf{P}) + \frac{1}{2\mu(1+a)} \text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\} \\ &\quad + \mu \overbrace{\text{tr}\{E[\mathbf{x}_T(\infty)\mathbf{x}_T^T(\infty)\mathbf{v}_T(\infty)\mathbf{v}_T^T(\infty)\mathbf{x}_T(\infty)\mathbf{x}_T^T(\infty)\hat{\mathbf{D}}^{-1}(\infty)]\}}^{\text{E8}}. \end{aligned} \quad (3.43)$$

Agora, rearranjando E8 como

$$\begin{aligned} \text{E8)} \quad &\text{tr}\{E[\mathbf{x}_T(\infty)\mathbf{x}_T^T(\infty)\mathbf{v}_T(\infty)\mathbf{v}_T^T(\infty)\mathbf{x}_T(\infty)\mathbf{x}_T^T(\infty)\hat{\mathbf{D}}^{-1}(\infty)]\} \\ &= E\{\mathbf{x}_T^T(\infty)\hat{\mathbf{D}}^{-1}(\infty)\mathbf{x}_T(\infty)\text{tr}\{\mathbf{x}_T^T(\infty)\mathbf{v}_T(\infty)\mathbf{v}_T^T(\infty)\mathbf{x}_T(\infty)\}\} \\ &= \text{tr}(\mathbf{P})\text{tr}[\mathbf{R}_T\mathbf{K}(\infty)] = \text{tr}(\mathbf{P})\xi_{\text{exc}}(\infty). \end{aligned} \quad (3.44)$$

e após algumas simples manipulações algébricas, a expressão resultante para o erro em excesso é

$$\xi_{\text{exc}}(\infty) = \frac{1}{1 - \mu \text{tr}(\mathbf{P})} \left(\mu \sigma_z^2 \text{tr}(\mathbf{P}) + \frac{1}{2\mu(1+a)} \text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\} \right). \quad (3.45)$$

3.4.2. DESAJUSTE

O desajuste \mathcal{M} é obtido a partir de (3.45), o qual é dado por

$$\mathcal{M}(\infty) = \frac{\xi_{\text{exc}}(\infty)}{\xi_{\text{min}}} = \frac{1}{1 - \mu \text{tr}(\mathbf{P})} \left(\mu \text{tr}(\mathbf{P}) + \frac{\sigma_z^{-2}}{2\mu(1+a)} \text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\} \right) \quad (3.46)$$

onde $\xi_{\text{min}} = \sigma_z^2$. Note que (3.46) é composto pela soma de dois termos

$$\mathcal{M}_1 = \frac{\mu \text{tr}(\mathbf{P})}{1 - \mu \text{tr}(\mathbf{P})} \quad (3.47)$$

e

$$\mathcal{M}_2 = \frac{[2\mu(1+a)\sigma_z^2]^{-1} \text{tr}\{E[\hat{\mathbf{D}}(\infty)]\Phi\}}{1 - \mu \text{tr}(\mathbf{P})}. \quad (3.48)$$

O primeiro é igual ao desajuste no caso estacionário (está sempre presente no sistema), o segundo é um termo originado da não-estacionaridade. Assim, o desajuste para o caso não-estacionário é maior do que aquele observado no caso estacionário.

Na Figura 3.1, é ilustrada a superfície de desajuste obtida variando os parâmetros μ e a . Os valores dos parâmetros utilizados na Figura 3.1 são os mesmos do Exemplo 5, Seção 3.6.1.5. Observe que o desajuste é uma função convexa em relação ao passo de adaptação μ , mostrando a existência de um passo de adaptação μ_{me} para o mínimo desajuste (que é coincidente com o mínimo erro em excesso) como uma função do parâmetro a . Além do mais, pode-se inferir que o desajuste tem um comportamento inverso em relação ao parâmetro a . Na Figura 3.2, curvas individuais para (3.47) e (3.48), como também para (3.46) são ilustradas, utilizando como variável independente μ para um dado valor de $a = 0,999$. Note que o termo devido à parte estacionária \mathcal{M}_1 é uma função crescente do passo de adaptação. Assim, para adaptação lenta, tal termo tende a ser pequeno. Por outro lado, a parte não-estacionária \mathcal{M}_2 é uma função convexa do passo de adaptação. A curva combinada, dada por (3.46), é também uma função convexa de μ , apresentando sempre um valor maior do que o observado no caso estacionário.

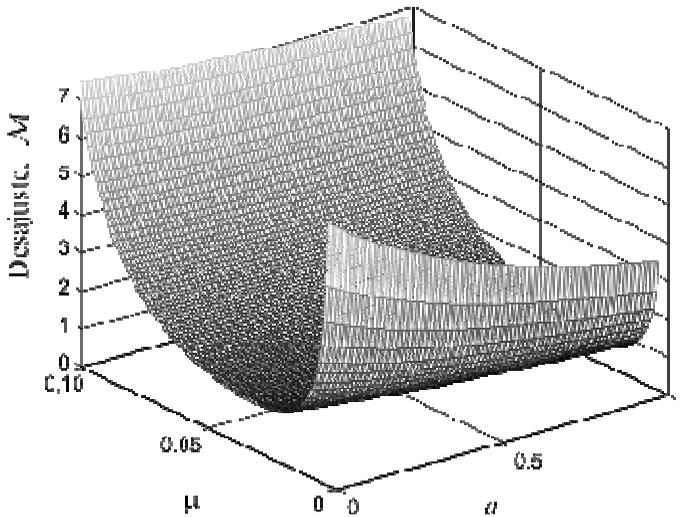


Figura 3.1. Desajuste (3.46) como uma função de μ e a .

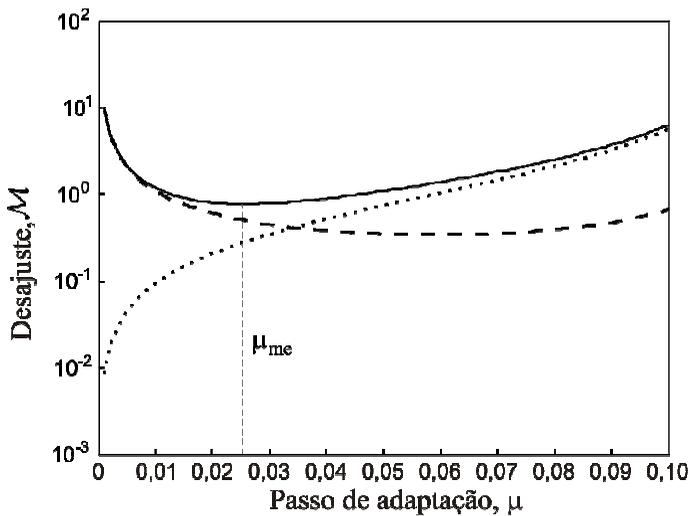


Figura 3.2. Desajuste como uma função de μ . (Linha pontilhada) \mathcal{M}_1 dada por (3.47). (Linha tracejada) \mathcal{M}_2 da (3.48). (Linha contínua) \mathcal{M} obtida por (3.46).

3.4.3. PASSO DE ADAPTAÇÃO PARA O MÍNIMO ERRO EM EXCESSO

O passo de adaptação que minimiza o EQME é obtido derivando (3.45) em relação a μ e igualando o resultado a zero. Dessa forma, tem-se

$$\frac{\partial \xi_{\text{exc}}(\infty)}{\partial \mu} = \frac{\sigma_z^2 c_1 \mu^2 + 2c_1 c_2 \mu - c_2}{(1 - \mu c_1)^2} = 0 \quad (3.49)$$

onde $c_1 = \text{tr}(\mathbf{P})$ e $c_2 = \frac{1}{2(1+a)} \text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\}$. Resolvendo (3.49) para μ , obtém-se

$$\mu_{\text{me}} = \frac{\text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\} \sqrt{1 + \frac{\sigma_z^2 2(1+a)}{\text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\} \text{tr}(\mathbf{P})} - 1}}{2(1+a) \sigma_z^2}. \quad (3.50)$$

Agora, considerando que $\mathbf{\Phi} = \sigma_\eta^2 \mathbf{I}$, sendo σ_η^2 a variância do vetor de perturbação da planta, resulta em

$$\mu_{\text{me}} = \frac{N \sigma_x^2 \sigma_\eta^2 \sqrt{1 + \frac{\sigma_z^2 2(1+a)}{N \sigma_x^2 \sigma_\eta^2 \text{tr}(\mathbf{P})} - 1}}{2(1+a) \sigma_z^2}. \quad (3.51)$$

3.4.3.1. Comentários sobre (3.51)

Nas Figuras 3.3(a) e 3.3(b) é ilustrado o passo de adaptação ótimo como função da variância da perturbação da planta σ_η^2 e o fator a , respectivamente. Observando a Figura 3.3(a), pode-se verificar que (3.50) é uma função crescente, apresentando saturações para níveis muito baixos e muito altos de σ_η^2 . Esse comportamento pode ser interpretado como discutido a seguir. Primeiramente, para valores muito baixos de σ_η^2 , o cenário é de um ambiente estacionário. Nessa situação, quanto menor for o passo de adaptação menor será o desajuste e,

portanto o passo para mínimo erro é igual a zero. Segundo, na medida em que σ_{η}^2 aumenta, um maior valor do passo de adaptação é necessário para obter o mínimo erro em excesso. Esse é um resultado intuitivo, sendo que um aumento em σ_{η}^2 gera mudanças mais rápidas na planta e, portanto o filtro adaptativo deve variar também mais rápido para poder acompanhar a planta. Finalmente, para valores altos σ_{η}^2 , o desajuste \mathcal{M} é principalmente afetado pelo termo não-estacionário \mathcal{M}_2 . Nessa situação, o valor do passo de adaptação para o mínimo erro em excesso, ou desajuste, pode ser obtido como segue:

$$\frac{\partial \mathcal{M}_2}{\partial \mu} = -\frac{\text{tr}\{E[\hat{\mathbf{D}}(\infty)]\mathbf{\Phi}\}}{2\sigma_z^2(1+a)} \frac{\partial}{\partial \mu} \left[\frac{1}{\text{tr}(\mathbf{P})\mu^2 - \mu} \right] \quad (3.52)$$

o qual, após fazer $\frac{\partial \mathcal{M}_2}{\partial \mu} = 0$, resulta

$$\mu_{\text{me}}^{\mathcal{M}_2} = \frac{1}{2\text{tr}(\mathbf{P})}. \quad (3.53)$$

Note que (3.53) só depende das estatísticas do sinal de entrada. Por outro lado, da Figura 3(b), observa-se que reduzindo o valor de a (o qual está relacionado com mudanças rápidas dos valores dos pesos da planta) um maior valor do passo de adaptação é requerido para acompanhar tais mudanças.

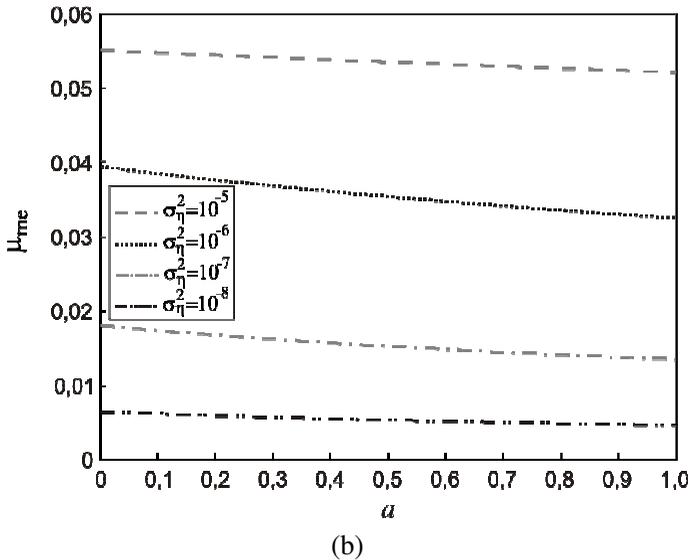
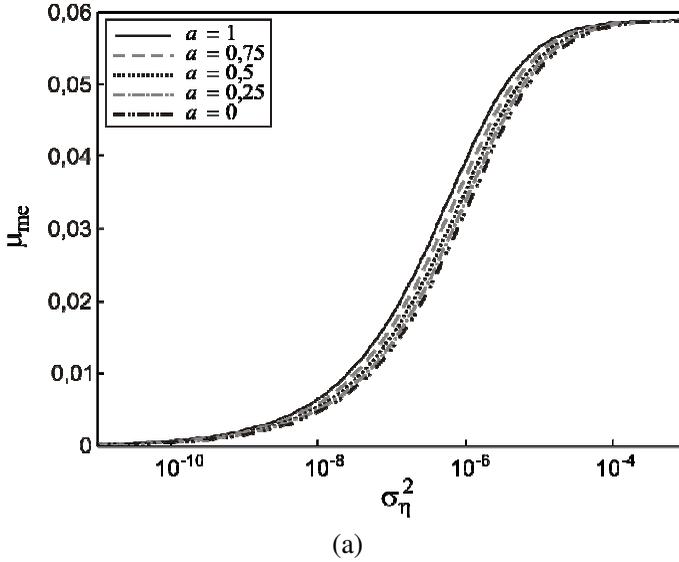


Figura 3.3. Curvas do passo de adaptação para o mínimo erro em excesso: (a) σ_{η}^2 variável e a fixo; (b) σ_{η}^2 fixo e a variável.

3.4.4. GRAU DE NÃO-ESTACIONARIDADE

Nesta seção, uma expressão para o grau de não-estacionaridade é derivada. Esse parâmetro é definido como [32]

$$\Psi(n) \triangleq \left\{ \frac{E[y_{o,inc}^2(n)]}{E[z^2(n)]} \right\}^{1/2} \quad (3.54)$$

onde

$$y_{o,inc}(n) = [\mathbf{w}_T^o(n+1) - \mathbf{w}_T^o(n)]^T \mathbf{x}_T(n) \quad (3.55)$$

é a saída devido à diferença $\mathbf{w}_T^o(n+1) - \mathbf{w}_T^o(n)$ (filtro incremental). O numerador de (3.54) representa a potência de saída do filtro incremental (indicativo das mudanças da planta) e o seu denominador denota o EQM mínimo. Agora, substituindo (3.4) em (3.55), obtém-se

$$y_{o,inc}(n) = [(a-1)\mathbf{w}_T^o(n) + \boldsymbol{\eta}(n)]^T \mathbf{x}_T(n) = \{(a-1)[a^n \mathbf{w}_T^o(0) + \sum_{k=0}^{n-1} a^k \boldsymbol{\eta}(n-1-k)] + \boldsymbol{\eta}(n)\}^T \mathbf{x}_T(n). \quad (3.56)$$

Note que (3.56) é derivado para $a \neq 1$, contrastando com o que é visto na literatura [31], [32]. Considerando a independência entre $\boldsymbol{\eta}(n)$ e $\mathbf{x}_T(n)$ e a hipótese (iii), (3.56) pode ser reescrita como

$$E[y_{o,inc}^2(n)] = (1-a)^2 a^{2n} \text{tr}[\mathbf{K}^o(0)\mathbf{R}_T] + \frac{2-a^{2n} + a^{2n+1}}{1+a} \text{tr}(\boldsymbol{\Phi}\mathbf{R}_T). \quad (3.57)$$

Agora, substituindo (3.57) em (3.54) e lembrando que σ_z^2 é a variância de $z(n)$, tem-se

$$\Psi(n) = \left\{ \frac{(1-a)^2 a^{2n}}{\sigma_z^2} \text{tr}[\mathbf{K}^o(0)\mathbf{R}_T] + \frac{2-a^{2n} + a^{2n+1}}{\sigma_z^2(1+a)} \text{tr}(\boldsymbol{\Phi}\mathbf{R}_T) \right\}^{1/2}. \quad (3.58)$$

De (3.58), vemos que $\Psi(n)$ apresenta as seguintes características:

- a) Evolução exponencial, iniciando pelo valor máximo

$$\Psi(0) = \frac{1}{\sigma_z^2} \{(1-a)^2 \text{tr}[\mathbf{K}^0(0)\mathbf{R}_T] + \text{tr}(\Phi\mathbf{R}_T)\}^{1/2} \quad (3.59)$$

e convergindo assintoticamente para

$$\Psi(\infty) = \frac{1}{\sigma_z} \left[\frac{2\text{tr}(\Phi\mathbf{R}_T)}{1+a} \right]^{1/2}. \quad (3.60)$$

- b) Para $a \neq 1$, o grau de não-estacionaridade é maior que para $a = 1$.
 c) Para $a = 1$, (3.59) e (3.60) resultam iguais a

$$\Psi|_{a=1} = \Psi(0) = \Psi(\infty) = \frac{1}{\sigma_z} \sqrt{\text{tr}(\Phi\mathbf{R}_T)} \quad (3.61)$$

que é freqüentemente dada na literatura [31], [32].

3.4.5. RELAÇÃO ENTRE O DESAJUSTE E GRAU DE NÃO-ESTACIONARIDADE

Nesta seção, uma relação entre o desajuste e o grau de não-estacionaridade é estabelecida considerando o ambiente não-estacionário dado por (3.3) e (3.4). Para tal, consideramos uma expressão alternativa para o erro em excesso, dada por

$$\xi_{\text{exc}}(n) = E\left(\{\mathbf{x}_T^T(n)[\mathbf{w}_T(n) - \mathbf{w}_T^0(n)]\}^2\right). \quad (3.62)$$

Substituindo (3.4) em (3.62) e rearranjando a expressão resultante, obtém-se

$$\xi_{\text{exc}}(n) = E\left(\{\mathbf{x}_T^T(n)[\mathbf{w}_T(n) - a\mathbf{w}_T^0(n-1)]\}^2\right) + E\{\{\mathbf{x}_T^T(n)\boldsymbol{\eta}(n-1)\}^2\}. \quad (3.63)$$

Multiplicando ambos os lados de (3.62) por a^2 e considerando (3.4), resulta

$$\begin{aligned} a^2 \xi_{\text{exc}}(n) &= E\left(\{\mathbf{x}_T^T(n)[a\mathbf{w}_T(n) - a\mathbf{w}_T^o(n)]\}^2\right) \\ &= E\left(\{\mathbf{x}_T^T(n)[a\mathbf{w}_T(n) - \mathbf{w}_T^o(n+1)]\}^2\right) + E\{\{\mathbf{x}_T^T(n)\boldsymbol{\eta}(n)\}^2\}. \end{aligned} \quad (3.64)$$

Assim, adicionando (3.63) com (3.64) e fazendo uso das propriedades de $\boldsymbol{\eta}(n)$, obtém-se

$$\begin{aligned} \xi_{\text{exc}}(n) &= \frac{1}{1+a^2} \left[E\left(\{\mathbf{x}_T^T(n)[\mathbf{w}_T(n) - a\mathbf{w}_T^o(n-1)]\}^2\right) \right. \\ &\quad \left. + E\left(\{\mathbf{x}_T^T(n)[a\mathbf{w}_T(n) - \mathbf{w}_T^o(n+1)]\}^2\right) + 2\text{tr}(\boldsymbol{\Phi}\mathbf{R}_T) \right]. \end{aligned} \quad (3.65)$$

De (3.65), a seguinte desigualdade pode ser estabelecida:

$$\xi_{\text{exc}}(n) \geq \frac{2}{1+a^2} \text{tr}(\boldsymbol{\Phi}\mathbf{R}_T) > \frac{2}{1+a} \text{tr}(\boldsymbol{\Phi}\mathbf{R}_T) \quad (3.66)$$

sendo $1+a^2 < 1+a$. Finalmente, dividindo (3.66) por ξ_{min} , tem-se

$$\frac{\xi_{\text{exc}}(n)}{\xi_{\text{min}}} > \frac{2}{(1+a)\sigma_z^2} \text{tr}(\boldsymbol{\Phi}\mathbf{R}_T). \quad (3.67)$$

O lado esquerdo de (3.67) é igual ao desajuste do algoritmo e o lado direito é igual ao quadrado do grau de não-estacionaridade em regime permanente, i. e.,

$$\mathcal{M}(n) > \psi^2(\infty). \quad (3.68)$$

Assim, de (3.68), podemos concluir que:

- 1) Se $\psi(\infty) \gg 1$, as variações na planta são muito rápidas para que o filtro seja capaz de acompanhá-las, resultando em um grande valor de desajuste.

- 2) Se $\xi_{\text{exc}}(n) > \xi_{\text{min}}$, que implica $\mathcal{M}(n) > 1$, o sinal $d(n)$ é uma melhor estimativa de $y^{\circ}(n)$ do que $y(n)$ gerado pelo filtro adaptativo [32].
- 3) Finalmente, se $\psi(\infty) \ll 1$, o algoritmo pode acompanhar as variações da planta com um mínimo erro de estimação.

3.5. MODELOS SIMPLIFICADOS

Nesta seção, modelos simplificados utilizando algumas aproximações adicionais as (i)-(iv) da Seção 3.3.1 são derivados.

3.5.1. MOMENTO DE PRIMEIRA ORDEM UTILIZANDO O PM

O modelo apresentado em [24] é modificado para considerar o comportamento do algoritmo em ambos os casos, ambiente estacionário e não-estacionário. Assim, aplicando o PM no termo P de (3.13), obtém-se

$$E[\mathbf{v}_T^{\text{PM}}(n+1)] \cong \{\mathbf{I} - 2\mu E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T\}E[\mathbf{v}_T^{\text{PM}}(n)] + (1-a)a^n \mathbf{w}_T^{\circ}(0) \quad (3.69)$$

onde os elementos de $E[\hat{\mathbf{D}}^{-1}(n)]$ são calculados como indicado em [24], resultando em

$$E\left[\frac{1}{\hat{\sigma}_i^2(n)}\right] = \frac{M}{2\sqrt{a_i}} \sum_{q=1}^{M/2} C_{i,q} \ln(\lambda'_{i,q}) \quad i=1,2,\dots,N, \quad (3.70)$$

com

$$C_{i,q} = \prod_{\substack{j=1 \\ j \neq q}}^{M/2} \frac{\lambda'_{i,q} \lambda'_{i,j}}{\lambda'_{i,j} - \lambda'_{i,q}}. \quad (3.71)$$

3.5.2. MOMENTO DE SEGUNDA ORDEM E CURVA DE APRENDIZAGEM

De forma similar ao realizado em (3.69), aplica-se o PM à (3.32) para obter

$$\begin{aligned}
\mathbf{K}^{\text{PM}}(n+1) &= \mathbf{K}^{\text{PM}}(n) - 2\mu\mathbf{K}^{\text{PM}}(n)\mathbf{R}_T E[\hat{\mathbf{D}}^{-1}(n)] - 2\mu E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T \mathbf{K}^{\text{PM}}(n) \\
&+ (1-a)E[\mathbf{v}_T^{\text{PM}}(n)]E[\mathbf{w}_T^{\text{oT}}(n)] + (1-a)E[\mathbf{w}_T^{\text{o}}(n)]E[\mathbf{v}_T^{\text{PM T}}(n)] \\
&+ 4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)]\{2\mathbf{R}_T \mathbf{K}^{\text{PM}}(n)\mathbf{R}_T + \mathbf{R}_T \text{tr}[\mathbf{R}_T \mathbf{K}^{\text{PM}}(n)]\}E[\hat{\mathbf{D}}^{-1}(n)] \\
&+ 4\mu^2 E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T E[\hat{\mathbf{D}}^{-1}(n)]\sigma_z^2 + (1-a)^2 \mathbf{K}^{\text{o}}(n) \\
&- 2\mu(1-a)E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T E[\mathbf{v}_T^{\text{PM}}(n)]E[\mathbf{w}_T^{\text{oT}}(n)] \\
&- 2\mu(1-a)E[\mathbf{w}_T^{\text{o}}(n)]E[\mathbf{v}_T^{\text{PM T}}(n)]\mathbf{R}_T E[\hat{\mathbf{D}}^{-1}(n)] + \Phi
\end{aligned} \tag{3.72}$$

onde $\mathbf{K}^{\text{PM}}(n) = E[\mathbf{v}^{\text{PM}}(n)\mathbf{v}^{\text{PM}}(n)]$.

Finalmente, de (3.72) e da definição da curva de aprendizagem considerando o PM, resulta

$$E[e_{\text{PM}}^2(n)] = \sigma_z^2 + \text{tr}[\mathbf{R}_T \mathbf{K}^{\text{PM}}(n)]. \tag{3.73}$$

3.5.3. ERRO EM EXCESSO

A expressão para o EQME utilizando o PM é obtida generalizando o procedimento apresentado em [24], resultando em

$$\xi_{\text{exc}}^{\text{PM}} = \frac{1}{1 - \mu \text{tr}\{E[\mathbf{D}^{-1}(n)]\mathbf{R}_T\}} \left[\mu \sigma_z^2 \text{tr}\{E[\mathbf{D}^{-1}(n)]\mathbf{R}_T\} + \frac{1}{2\mu(1+a)} \text{tr}\left(\{E[\hat{\mathbf{D}}^{-1}(n)]\}^{-1}\Phi\right) \right]. \tag{3.74}$$

Note que, invocando o PM, os valores esperados \mathbf{P} e \mathbf{S} são determinados separando-os como $E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T$ e $E[\hat{\mathbf{D}}^{-1}(n)]\mathbf{R}_T E[\hat{\mathbf{D}}^{-1}(n)]$, respectivamente. O modelo simplificado baseado no PM se aproxima assintoticamente do modelo apresentado quando o comprimento da janela de observação M é aumentado ($M > 32$ aproximadamente), nesse caso ambos os modelos são

coincidentes. Assim, podemos concluir que, nessa situação, o PM provê uma predição razoável, utilizando expressões menos complexas. Na Seção 3.6.3, são apresentados, para propósito de comparação, os valores de P e S obtidos utilizando simulações de MC, PM e o método aqui proposto.

3.6. RESULTADOS DE SIMULAÇÃO

Nesta seção, o modelo proposto é aplicado a um problema de identificação de sistema visando avaliar sua precisão para sinais de entrada gaussiano correlacionado, obtido a partir de um processo auto-regressivo AR(2) dado por

$$x(n) = a_1 x(n-1) + a_2 x(n-2) + v(n) \quad (3.75)$$

onde $v(n)$ é um ruído branco com variância σ_v^2 , o qual é ajustado para que a variância de $x(n)$ seja unitária, $a_1 = 0,18$ e $a_2 = -0,85$ são os pesos do processo AR(2). A dispersão dos autovalores do sinal de entrada para $N = 8$ é $\chi = 46,1$. Os pesos da planta variante são obtidos de acordo com (3.4). Seus valores iniciais são selecionados como

$$\mathbf{w}_T^o(0) = \frac{\mathbf{w}_{aux}}{\sqrt{\mathbf{w}_{aux}^T \mathbf{w}_{aux}}} \quad (3.76)$$

com o vetor $\mathbf{w}_{aux} = \mathbf{T}[\text{sinc}(0) \text{ sinc}(1/N) \cdots \text{sinc}(N-1/N)]^T$. Os elementos de $\boldsymbol{\eta}(n)$ são amostras de um ruído branco com matriz de autocorrelação $\boldsymbol{\Phi} = \sigma_\eta^2 \mathbf{I}$. Nos exemplos considerados, são usados $a = 0,999$ e dois níveis de ruído aditivo, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$, resultando, respectivamente, em graus de não-estacionaridade inicial $\psi(0) = 0,2075$ e $\psi(0) = 0,588$, para cada nível de ruído aplicado, e final $\psi(\infty) = 0,200$, para as duas condições. A transformada do cosseno discreta (TCD) é a transformada ortogonal aqui utilizada. A TCD é uma das aproximações mais utilizadas para a transformada de Karhunen-Loève (TKL). Simulações de Monte Carlo (MC) são obtidas considerando 200 realizações independentes.

Para os casos não-estacionários, os pesos são inicializados por

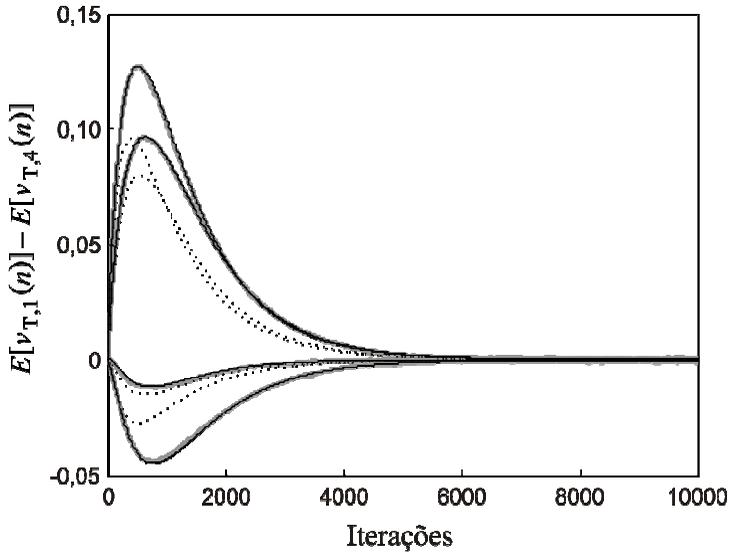
$$\mathbf{w}_T(0) = \mathbf{w}_T^o(n) \Big|_{n=0} \quad (3.77)$$

visando avaliar o comportamento de rastreamento do algoritmo. No caso estacionário, os pesos são inicializados com zeros para se avaliar o comportamento do algoritmo no modo de aquisição [31]. Para propósitos de comparação, o comportamento do algoritmo predito pelo modelo baseado no PM (veja Seção 3.3.10) é também apresentado.

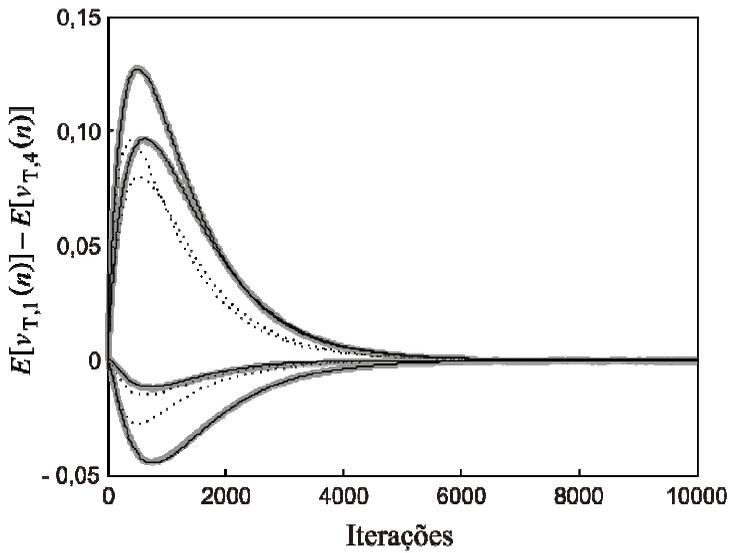
3.6.1. VERIFICAÇÃO DO MODELO PROPOSTO

3.6.1.1. Exemplo 1

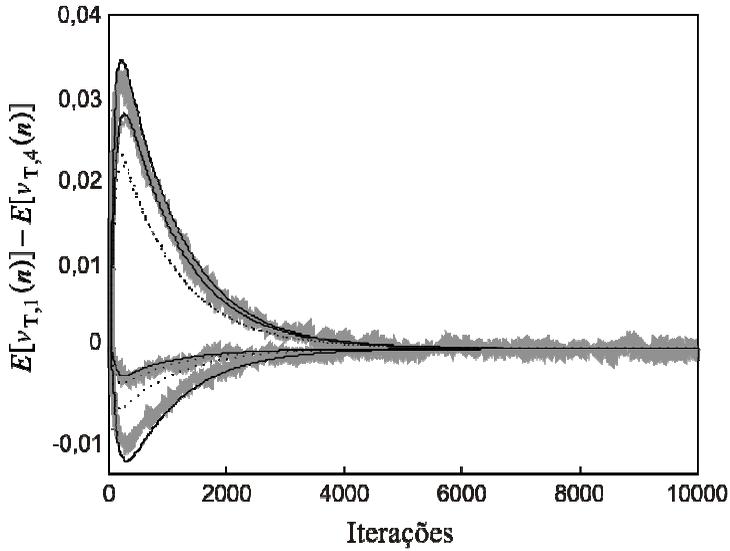
Neste exemplo, um filtro com $N=8$ pesos e uma janela de observação de $M=8$ amostras são considerados. Nessa condição, o passo de adaptação para o mínimo erro em excesso é $\mu_{me} = 0,0257$. Na Figura 3.4, o momento de primeira ordem do vetor de erro nos pesos é mostrado para $\sigma_z^2 = -40$ dB e $\mu = 0,1\mu_{me}$ [veja Figura 3.4(a)], $\sigma_z^2 = -60$ dB e $\mu = 0,1\mu_{me}$ [veja Figura 3.4(b)] e $\sigma_z^2 = -40$ dB e $\mu = 0,5\mu_{me}$ [veja Figura 3.4(c)]. Para uma melhor visualização, somente quatro pesos são apresentados. A Figura 3.5 mostra a curva de aprendizagem (EQM) para os dois níveis de ruído e os dois passos de adaptação indicados anteriormente. Para esse exemplo, o método proposto mostra-se superior ao baseado no PM, atingindo um excelente casamento com as simulações de MC. Também, observa-se uma diminuição na precisão dos modelos apresentados quando o passo de adaptação é aumentado.



(a)

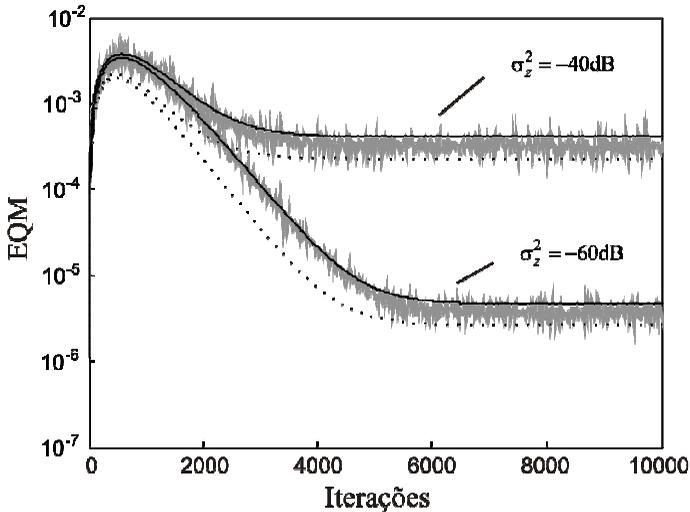


(b)

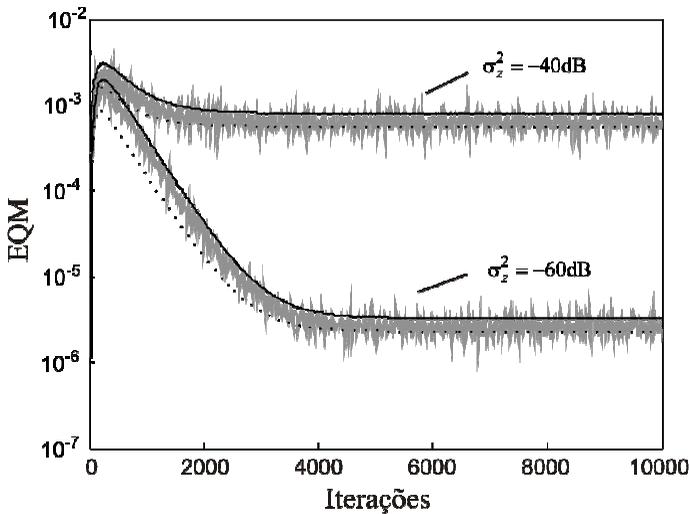


(c)

Figura 3.4. Exemplo 1. Curvas do comportamento médio do vetor de erro. De cima para baixo: $E[v_{T,i}(n)]$, para $i = 1, 2, 4, 3$. (Linhas cinza) simulações de MC. (Linhas pontilhadas escuras) modelo baseado no PM (3.69). (Linhas contínuas escuras) modelo proposto (3.22). (a) $\mu = 0,1\mu_{me}$ e $\sigma_z^2 = -40$ dB. (b) $\mu = 0,1\mu_{me}$ e $\sigma_z^2 = -60$ dB. (c) $\mu = 0,5\mu_{me}$ e $\sigma_z^2 = -40$ dB.



(a)

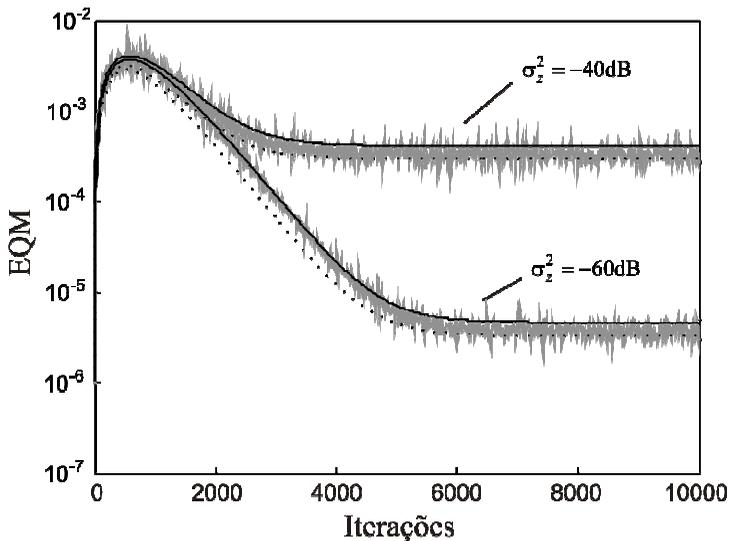


(b)

Figura 3.5. Exemplo 1. Curvas do EQM para $M = 8$. (Linhas cinza) simulações de MC. (Linhas pontilhadas escuras) modelo baseado no PM (3.73). (Linhas contínuas escuras) modelo proposto (3.38). (a) $\mu = 0,1\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$. (b) $\mu = 0,5\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$.

3.6.1.2. Exemplo 2

Este exemplo utiliza os mesmos parâmetros do Exemplo 1, exceto M , que é aumentado para 16 visando atingir uma melhor estimativa da variância. As curvas de EQM são ilustradas na Figura 3.6, observando-se agora um bom ajuste das simulações de MC para os dois modelos considerados, baseado no PM e proposto.



(a)

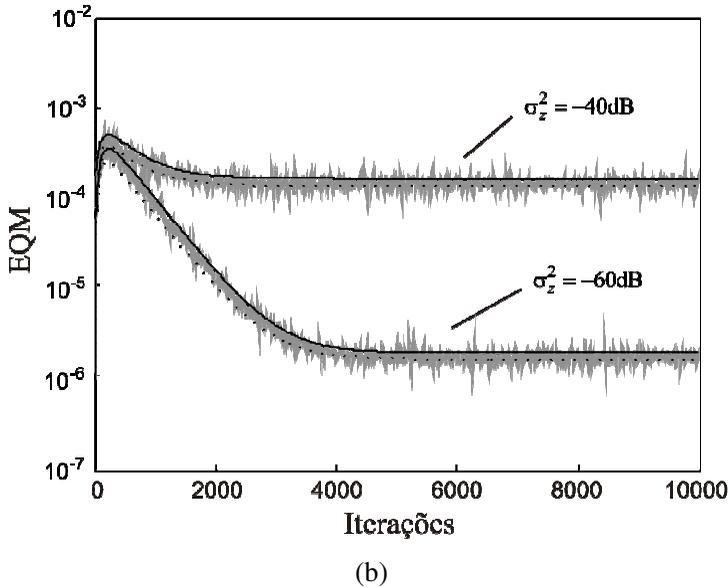
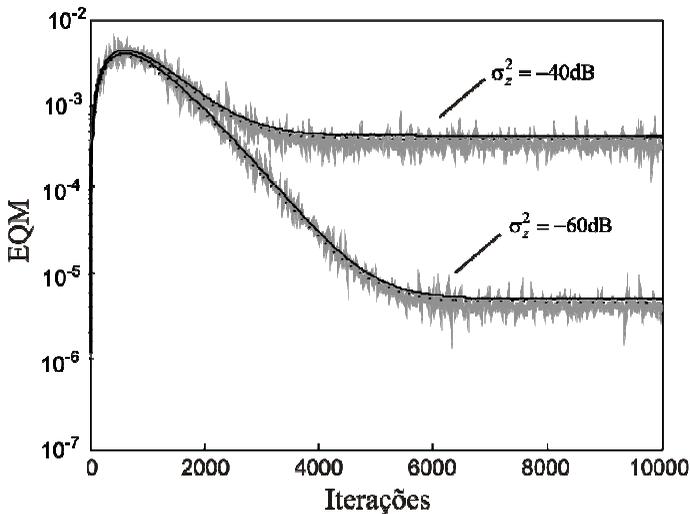


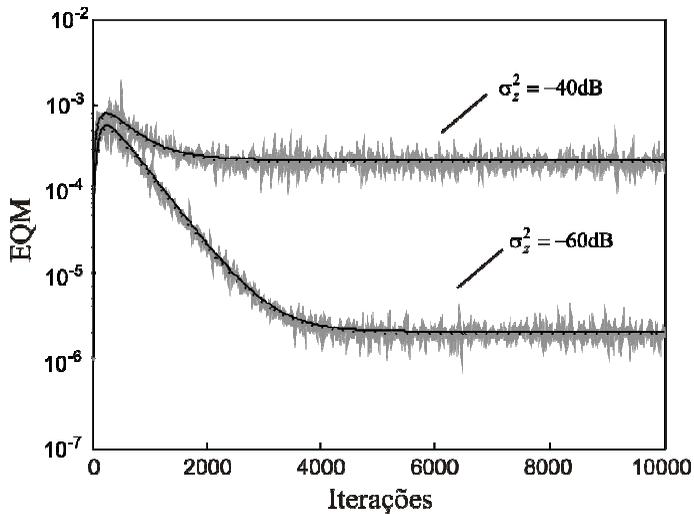
Figura 3.6. Exemplo 2. Curvas do EQM para $M = 16$. (Linhas cinza) simulações de MC. (Linhas pontilhadas escuras) modelo baseado no PM (3.73). (Linhas contínuas escuras) modelo proposto (3.38). (a) $\mu = 0,1\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$. (b) $\mu = 0,5\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$.

3.6.1.3. Exemplo 3

Neste exemplo, a maior parte dos parâmetros do Exemplo 1 é mantida, exceto M , que é aumentado para $M = 64$, visando obter uma melhor estimativa da variância. Os resultados obtidos para a evolução do EQM são mostrados na Figura 3.7. Note que, nesse caso, o modelo baseado no PM e o modelo proposto apresentam quase o mesmo comportamento, visto que agora o valor de M favorece a aproximação do PM.



(a)

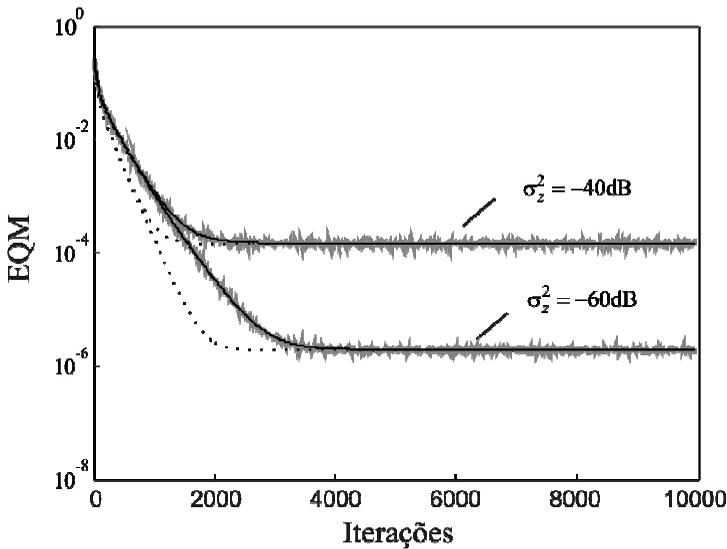


(b)

Figura 3.7. Exemplo 3. Curvas do EQM para $M = 64$. (Linhas cinza) simulações de MC. (Linhas pontilhadas escuras) modelo baseado no PM (3.73). (Linhas contínuas escuras) modelo proposto (3.38). (a) $\mu = 0,1\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$. (b) $\mu = 0,5\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$.

3.6.1.4. Exemplo 4

Neste exemplo, o caso estacionário é considerado como um caso particular do não-estacionário ($a=1$ e grau de não-estacionaridade $\psi=0$). De novo, é considerado dois valores de passo de adaptação, $\mu=0,00257$ e $\mu=0,01285$. Outros parâmetros utilizados são: $N=8$ e $M=8$. Os resultados obtidos são mostrados na Figura 3.8. Note que agora uma redução no valor do EQM é observada, visto que o termo correspondente a não-estacionaridade em (3.45) é zero. Note também que o modelo proposto apresenta um casamento muito bom com as simulações de MC.



(a)

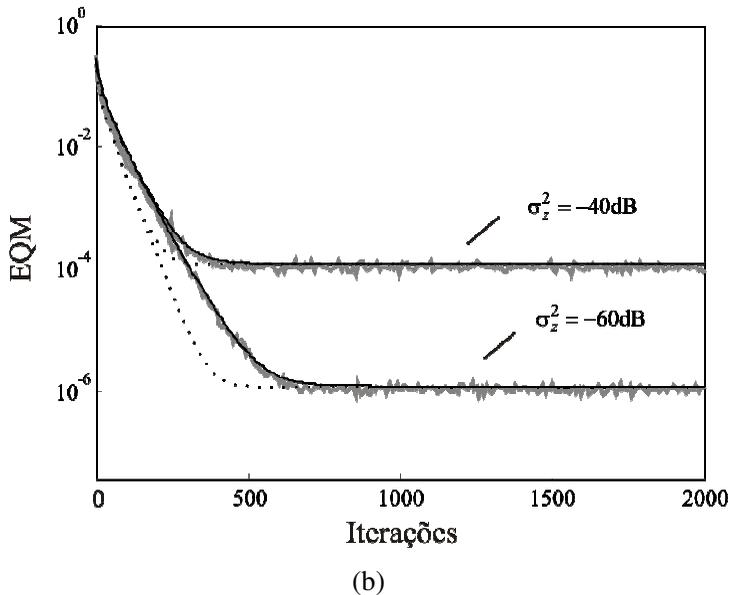
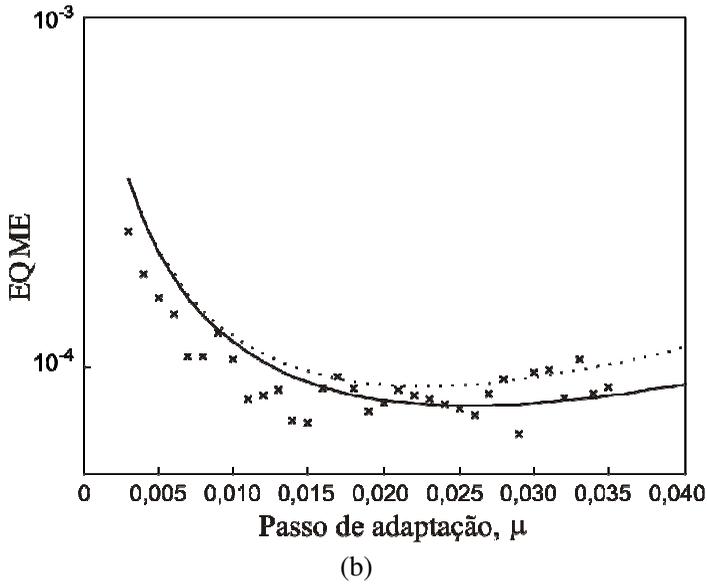
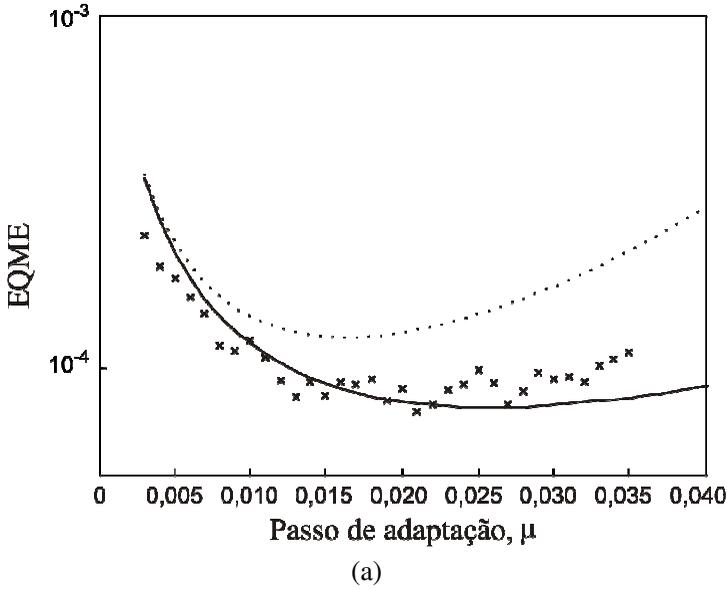
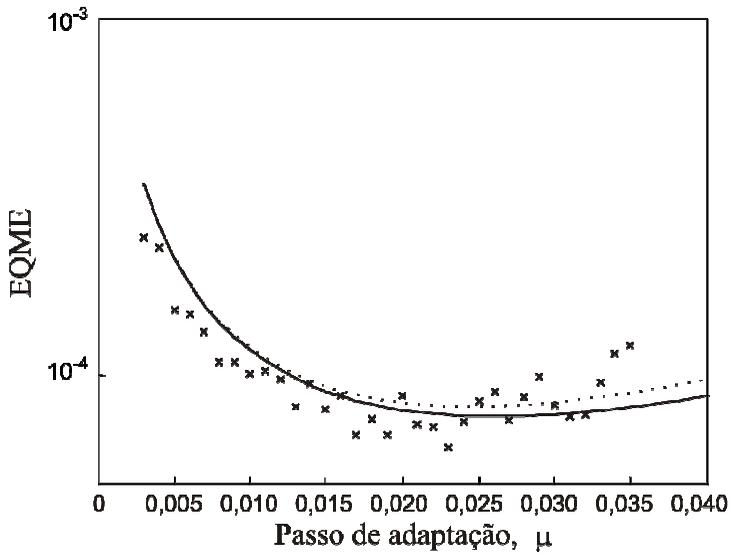


Figura 3.8. Exemplo 4. Curvas do EQM para o caso estacionário. (Linhas cinza) simulações de MC. (Linhas pontilhadas escuras) modelo baseado no PM (3.73). (Linhas contínuas escuras) modelo proposto (3.38). (a) $\mu = 0,1\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$. (b) $\mu = 0,5\mu_{me}$, $\sigma_z^2 = -40\text{dB}$ e $\sigma_z^2 = -60\text{dB}$.

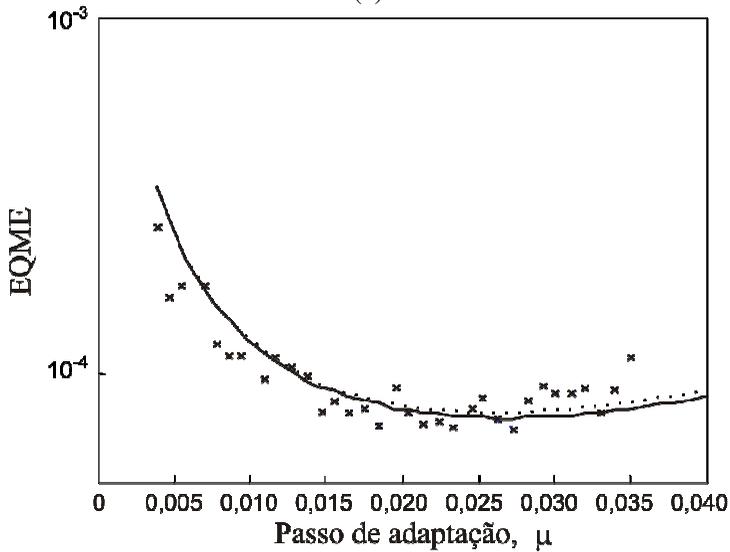
3.6.1.5. Exemplo 5

Neste exemplo, resultados de simulações de MC para o erro em excesso são comparados com os resultados preditos a partir de (3.45) e (3.74). A mesma entrada correlacionada dos exemplos anteriores com $\sigma_z^2 = -40\text{dB}$ é utilizada. São consideradas janelas de observação que vão de $M = 8$ até $M = 128$. A Figura 3.9 ilustra os resultados obtidos, mostrando uma satisfatória predição obtida a partir de (3.45) em todos os casos e um aumento gradual de precisão da predição de (3.74) a medida que o valor de M é aumentado.





(c)

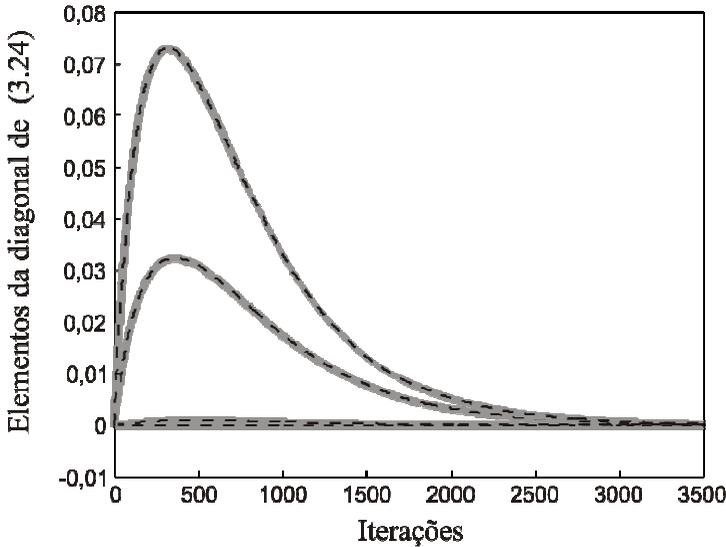


(d)

Figura 3.9. Exemplo 5. Resultados de comparação para o erro em excesso. (Valores marcados com \times) simulações de MC. (Linhas pontilhadas) modelo baseado no PM (3.74). (Linhas contínuas) modelo proposto (3.45). (a) $M = 8$. (b) $M = 32$. (c) $M = 64$. (d) $M = 128$.

3.6.2. VERIFICAÇÃO DAS APROXIMAÇÕES DA SEÇÃO 3.3.3

Nesta seção, simulações numéricas demonstrando a validade das aproximações (3.24), (3.25), (3.27) e (3.29) são apresentadas. As aproximações (3.26) e (3.28) são semelhantes às (3.25) e (3.27), respectivamente, e, portanto não serão aqui verificadas. A Figura 3.10 ilustra (3.24) considerando os casos apresentados no Exemplo 1 e Exemplo 3, utilizando $M = 8$ e $M = 64$, respectivamente, e $\sigma_z^2 = -40$ dB. Na Figura 3.10(a), é apresentada a evolução dos primeiros quatro elementos da diagonal principal das matrizes de cada lado de (3.24) para o Exemplo 1. A Figura 3.10(b) mostra o mesmo padrão de apresentação da Figura 3.10(a) agora para o Exemplo 3. A partir dos gráficos obtidos, verifica-se a pertinência da aproximação utilizada, levando em conta o excelente casamento entre as curvas.



(a)

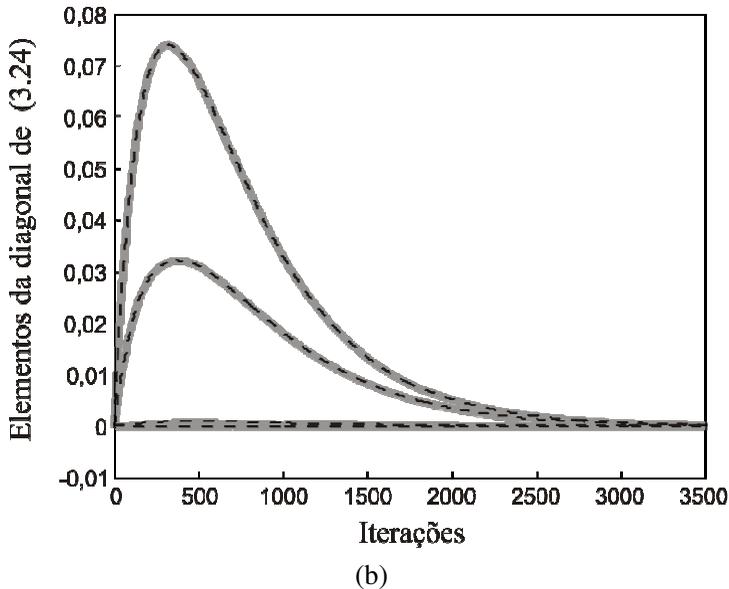
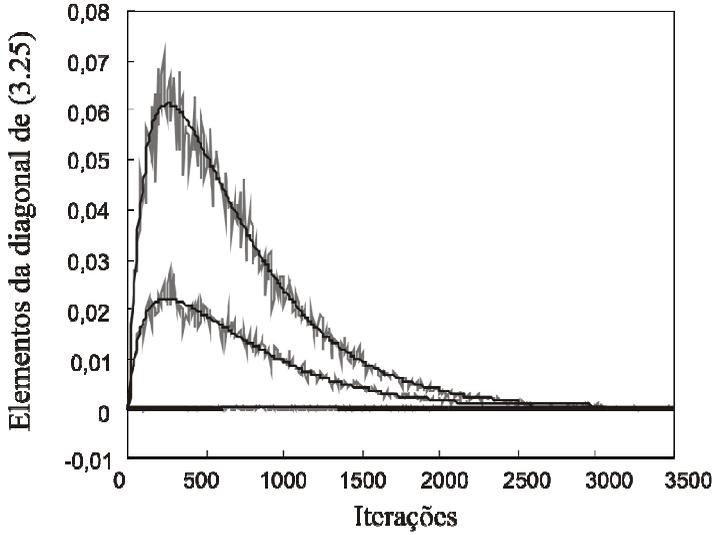
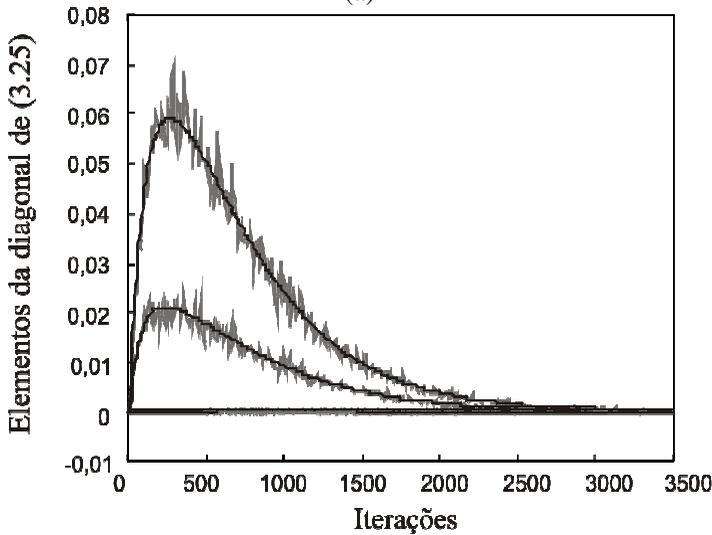


Figura 3.10. Teste da Aproximação (3.24). (Linhas cinza contínuas) $\{E[\mathbf{v}_T(n)\mathbf{w}_T^{\circ T}(n)]\}_{i,i}$ (determinado por simulações de MC). (Linhas tracejadas escuras) $\{E[\mathbf{v}_T(n)]E[\mathbf{w}_T^{\circ T}(n)]\}_{i,i}$. De cima para baixo: $i = 1, 2, 3, 4$. (a) $M = 8$ e (b) $M = 64$.

Nas Figuras 3.11, 3.12 e 3.13, o padrão de apresentação da Figura 3.10 é reproduzido, dessa vez para as aproximações (3.25), (3.27) e (3.29). Em geral, é observado um bom casamento entre simulações e as expressões utilizadas. Só é percebida uma pequena discrepância na Fig. 3.13(a), decorrente do fato dos sinais envolvidos não serem gaussianos para $M = 8$. Pode ser concluído que todas as aproximações utilizadas são válidas dentro das condições em que foram aplicadas.

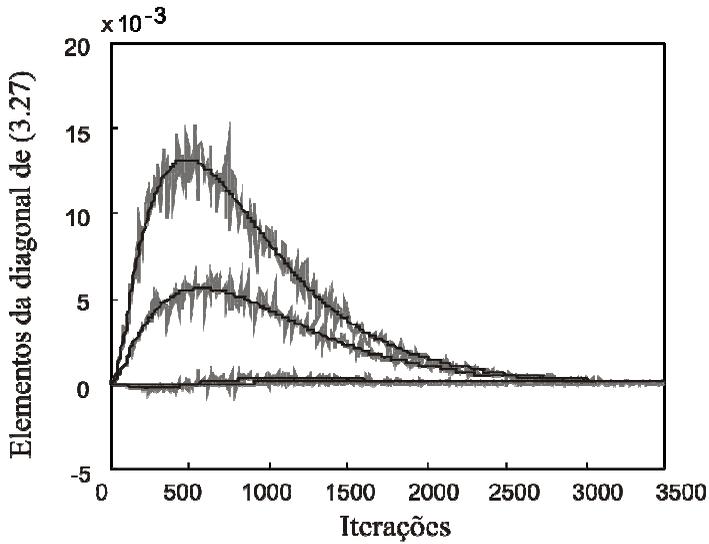


(a)

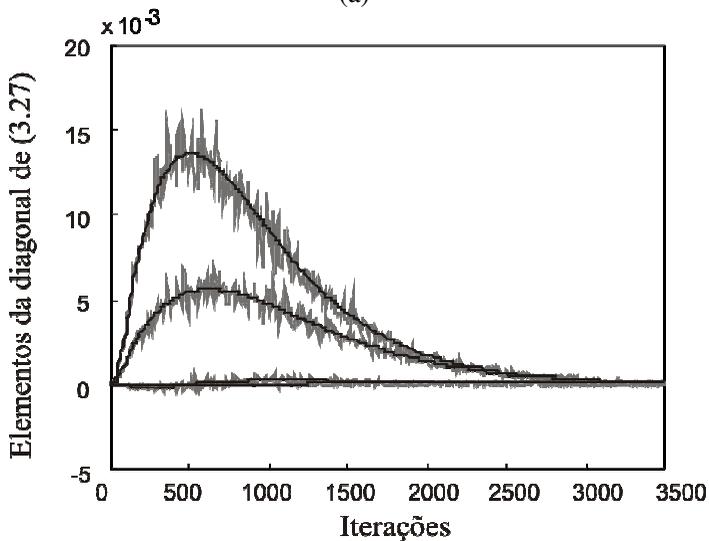


(b)

Figura 3.11. Teste da Aproximação (3.25). (Linhas cinza) $\{E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{w}_T^{\text{OT}}(n)]\}_{i,i}$ (determinado por simulações de MC). (Linhas tracejadas escuras) $\{\mathbf{PE}[\mathbf{v}_T(n)]E[\mathbf{w}_T^{\text{OT}}(n)]\}_{i,i}$. De cima para baixo: $i = 1, 2, 3, 4$. (a) $M = 8$ e (b) $M = 64$.

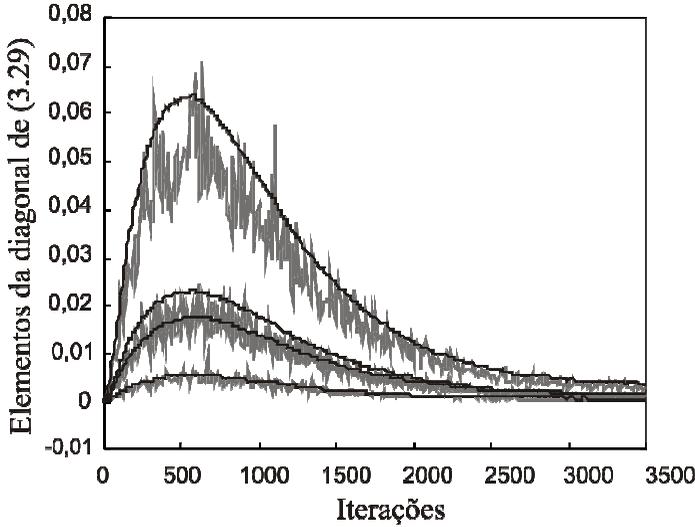


(a)

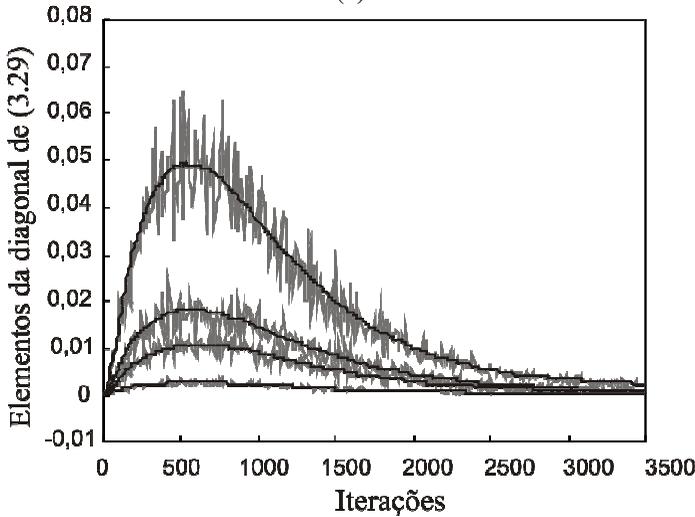


(b)

Figura 3.12. Teste da Aproximação (3.27). (Linhas cinza) $\{E[\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]\}_{i,i}$ (determinado por simulações de MC). (Linhas tracejadas escuras) $[\mathbf{K}(n)\mathbf{P}^T]_{i,i}$. De cima para baixo: $i=1,2,3,4$. (a) $M=8$ e (b) $M=64$.



(a)



(b)

Figura 3.13. Teste da Aproximação (3.29). (Linhas cinza) $\{E[\hat{\mathbf{D}}^{-1}(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\mathbf{v}_T(n)\mathbf{v}_T^T(n)\mathbf{x}_T(n)\mathbf{x}_T^T(n)\hat{\mathbf{D}}^{-1}(n)]\}_{i,i}$ (determinado por simulações de MC). (Linhas tracejadas escuras) $\{2\mathbf{PK}(n)\mathbf{P}^T + \text{Str}[\mathbf{K}(n)\mathbf{R}_T]\}_{i,i}$. De cima para baixo: $i=1,2,3,4$. (a) $M=8$ e (b) $M=64$.

3.6.3. RESULTADOS DE COMPARAÇÃO PARA AS MATRIZES \mathbf{P} E \mathbf{S}

Nesta seção, são apresentados os resultados de comparação entre os elementos da diagonal principal das matrizes \mathbf{P} e \mathbf{S} calculados utilizando simulações de MC, o PM e o método proposto. Os parâmetros considerados são os mesmos do Exemplo 1. Os resultados obtidos para as matrizes \mathbf{P} e \mathbf{S} são mostrados nas Figuras 3.14 e 3.15, respectivamente.

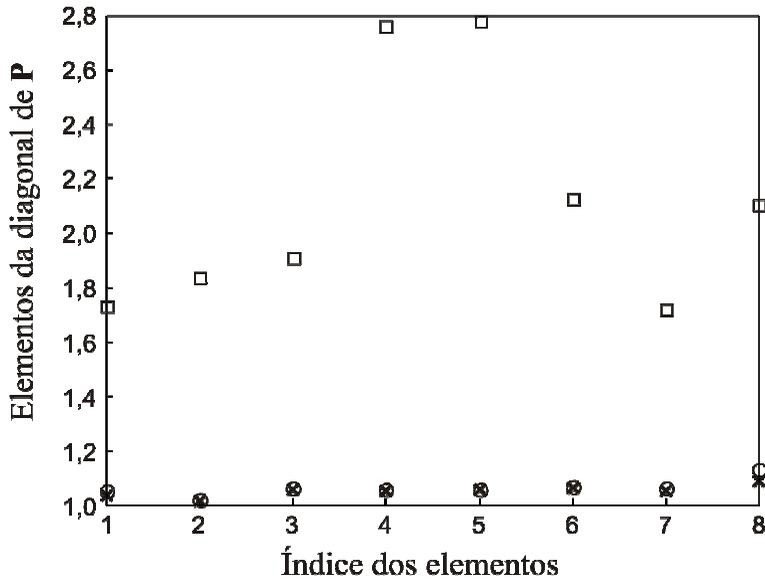


Figura 3.14. Teste de aproximação da matriz \mathbf{P} (3.14). ('x') Simulações de MC. ('□') PM. ('o') Método proposto.

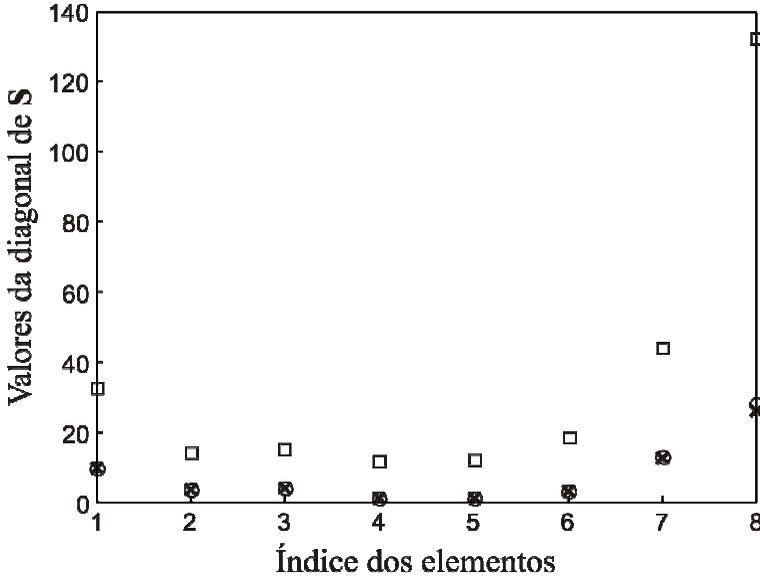


Figura 3.15. Teste de aproximação da matriz S (3.30). ('x') Simulações de MC. ('□') PM. ('o') Método proposto.

3.7. CONCLUSÕES

Este capítulo apresentou um modelo estatístico para o algoritmo LMS-DT operando em ambiente não-estacionário. Adicionalmente, o caso estacionário pode ser obtido a partir do modelo desenvolvido. A análise realizada é independente da ordem do filtro e do comprimento da janela de observação usada para estimar a potência do sinal em cada sub-banda. Expressões para o erro em excesso, desajuste e passo de adaptação para o erro mínimo em excesso são também derivadas. Além do mais, uma expressão matemática que relaciona o grau de não-estacionaridade e o desajuste do algoritmo foi apresentada. Comparações entre os resultados obtidos a partir do modelo proposto e simulações de MC confirmam uma precisão muito boa das expressões apresentadas.

4.1. INTRODUÇÃO

Nos últimos anos, as redes de comunicação sem fio vêm experimentando um aumento crescente no número de usuários, gerando, assim, sérios problemas em ambientes densamente urbanizados, visto que o espectro de frequências está se aproximando do limite de sua capacidade [33]; nesses casos, células setorizadas são utilizadas. Para superar a barreira de limitação do espectro, é necessário reduzir o tamanho das células, aumentando assim a reutilização de frequências. Por exemplo, em um esquema de sete células por *cluster*, com três setores de 120° por célula, a quantidade de frequências portadoras oferecida a cada setor é $1/21$ do total de frequências disponíveis. Um esquema de três células por *cluster* incrementa a capacidade por um fator de aproximadamente $7/3$ (veja Figura 4.1). Porém, a distância entre células com cocanais interferentes decresce por um fator de $\sqrt{21/9}$, aumentando o nível de interferência entre canais [33].

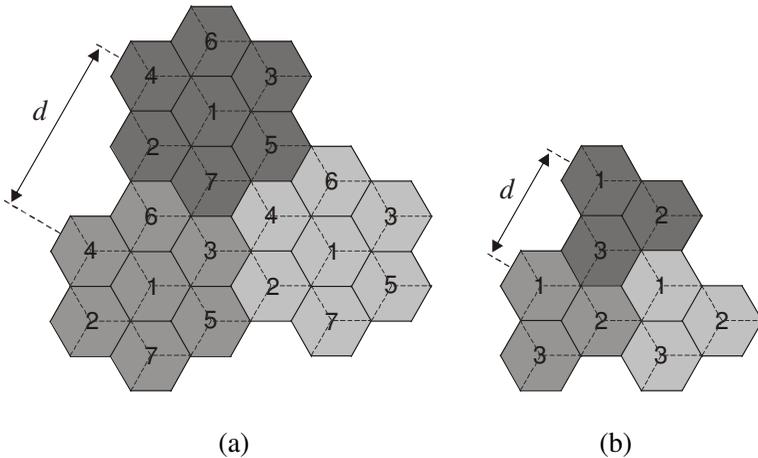


Figura 4.1. Configurações do sistema celular: (a) esquema de 7 células por *cluster*; (b) esquema de 3 células por *cluster*.

Uma estratégia para reduzir a interferência entre canais é utilizar, na estação de rádio base (ERB), arranjos de antenas com diagramas de irradiação conformados segundo a posição dos terminais móveis da própria célula e do cocanal.

Idealmente, toda ERB deveria conhecer a posição de todos os terminais móveis da área, obtendo assim uma solução ótima global. No estado de desenvolvimento atual, isso não é ainda possível de ser implementado. Nos dias de hoje, tudo o que uma ERB conhece sobre os terminais móveis das células vizinhas é determinado através dos sinais do enlace reverso (*uplink*) do arranjo. Com essa informação disponível, pode ser usada uma abordagem em que as células vizinhas reduzam a potência transmitida na direção dos terminais móveis do cocanal; assim, a interferência é reduzida globalmente.

Nos sistemas em que a comunicação bidirecional (*full-duplex*) é realizada através da técnica de divisão de frequência (FDD – *frequency division duplex*), um canal é utilizado para o enlace reverso e outro próximo para o enlace direto (*downlink*). Para sinais com frequências na faixa dos gigahertz, comumente usadas em comunicações celulares, as frequências portadoras dos enlaces (reverso e direto) estão suficientemente separadas para que o desvanecimento (*fading*) não afete ambos os enlaces da mesma forma. Apesar disso, as bandas de ambos os enlaces em FDD estão geralmente próximas, o que faz com que a geometria do ambiente de propagação seja similar, resultando assim em matrizes de covariância similares [34]. Em [35] e [36], essa propriedade é explorada para o controle do arranjo adaptativo considerando estimativas das matrizes de covariância, tanto do sinal do enlace direto quanto do sinal interferente (ambas obtidas a partir dos sinais do enlace reverso). Em [25], esse procedimento é utilizado para estabelecer uma função objetivo, cuja maximização aproxima uma solução ótima global em uma rede cooperativa, sem a necessidade de comunicação entre células. A partir dessa função objetivo, é obtido o algoritmo do gradiente estocástico com restrições (CSG). O CSG é um algoritmo de baixa complexidade computacional, com uma velocidade de convergência satisfatória [25].

O desempenho do arranjo adaptativo pode ser avaliado através da função de distribuição de probabilidade (FDP) da SINR (*signal-to-interference-plus-noise ratio*), denotada por $F(\gamma)$, levando em conta um conjunto amplo de condições de operação. Assim, faz-se necessário considerar um número grande de combinações de ângulos de chegada tanto para o sinal do enlace reverso quanto para os sinais

interferentes de cocanal. Dessa forma, a avaliação de desempenho do algoritmo requer um número grande de simulações; considerando, além das muitas combinações de ângulos de chegada, numerosas realizações do algoritmo para se obter valores médios. Nesse caso, torna-se importante dispor de um modelo apropriado que descreva o comportamento do algoritmo adaptativo, permitindo obter essa informação utilizando agora expressões analíticas ao invés de simulações. Assim, neste trabalho, são propostos modelos analíticos para o momento de primeira ordem do vetor de pesos do arranjo adaptativo e para a curva de SINR. Esses modelos são derivados assumindo passos de adaptação pequenos, condição essa em que o sistema apresenta uma convergência suave.

4.2. ALGORITMO CSG

O diagrama de blocos de um arranjo adaptativo de antenas é mostrado na Figura 4.2, no qual o sinal de enlace direto é dividido em N ramos. Cada um desses sinais é multiplicado pelo seu correspondente peso complexo $w_i(n)$ com $i = 1, \dots, N$. O objetivo do algoritmo adaptativo é ajustar os pesos do arranjo, visando aumentar a potência irradiada na direção do terminal móvel da própria célula P_{IC} e reduzir a potência do sinal transmitido a terminais móveis P_{CC} das células vizinhas (cocanal). Tais potências são expressas, respectivamente, como

$$P_{IC} = \mathbf{w}^H \mathbf{R}_{IC} \mathbf{w} \quad (4.1)$$

e

$$P_{CC} = \mathbf{w}^H \mathbf{R}_{CC} \mathbf{w} \quad (4.2)$$

onde $\mathbf{w} = [w_1 \ w_2 \ \dots \ w_N]^T$ denota o vetor de pesos complexos, e \mathbf{R}_{IC} e \mathbf{R}_{CC} são, respectivamente, a matriz de covariância do sinal do enlace reverso e a matriz de covariância do sinal interferente de cocanal. Assim, o algoritmo de adaptação deve maximizar a seguinte função objetivo [25]:

$$\xi = \frac{\mathbf{w}^H \mathbf{R}_{IC} \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{CC} + \mathbf{R}_{\eta}) \mathbf{w}} \quad (4.3)$$

onde \mathbf{R}_{η} é a matriz de covariância do ruído aditivo. Aqui será considerado que o ruído aditivo é i.i.d com variância unitária; portanto $\mathbf{R}_{\eta} = \mathbf{I}$, sendo \mathbf{I} a matriz identidade. Note que o vetor de pesos complexos \mathbf{w} em (4.3) é a variável independente daquela função objetivo determinística. O algoritmo CSG, dado em [25], é obtido a partir de uma interpretação intuitiva de (4.3), considerando $\mathbf{w}(n)$ no lugar de \mathbf{w} , onde $\mathbf{w}(n)$ é agora variante no tempo, resultante do processo adaptativo discutido a seguir. Assim, utilizando o algoritmo de descida mais íngreme (*steepest descent algorithm*) para a potência do sinal interferente e o algoritmo de subida mais íngreme (*steepest ascent algorithm*) para o sinal do enlace reverso, consegue-se maximizar (4.3). Entretanto, essa estratégia pode resultar em baixa velocidade de convergência quando a atualização dos pesos fizer com que os termos do numerador e do denominador de (4.3), simultaneamente, aumentem (ou diminuam).

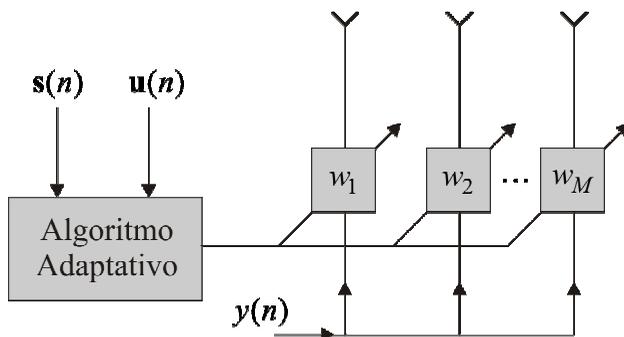


Figura 4.2. Diagrama de blocos de um arranjo adaptativo de antenas.

Uma solução heurística para tal problema é utilizar um esquema de atualização em dois estágios. No primeiro estágio, o numerador de (4.3) é adaptado mantendo o denominador fixo; no segundo, o denominador é adaptado, mantendo o numerador fixo. Matematicamente, esse processo pode ser implementado incorporando uma matriz de projeções na equação de atualização. O algoritmo CSG

utiliza o sinal de enlace reverso $s(n)$ e o sinal interferente de cocanal $u(n)$ para estimar as matrizes de covariância \mathbf{R}_{IC} e \mathbf{R}_{CC} , respectivamente. Assim, para o caso de dois interferentes, as equações de atualização dos pesos são, conforme [25], dadas por

$$\mathbf{v}_1(n) = \mathbf{w}(n) + \mu_s \left[\mathbf{I} - \frac{\hat{\mathbf{R}}_{u_1}(n)}{\|\mathbf{u}_1(n)\|^2} - \frac{\hat{\mathbf{R}}_{u_2}(n)}{\|\mathbf{u}_2(n)\|^2} \right] \hat{\mathbf{R}}_s(n) \mathbf{w}(n), \quad (4.4)$$

$$\mathbf{v}_2(n) = \mathbf{v}_1(n) - \mu_u \left[\mathbf{I} - \frac{\hat{\mathbf{R}}_s(n)}{\|\mathbf{s}(n)\|^2} \right] [\hat{\mathbf{R}}_{u_1}(n) + \hat{\mathbf{R}}_{u_2}(n)] \mathbf{w}(n) \quad (4.5)$$

e

$$\mathbf{w}(n+1) = \frac{\mathbf{v}_2(n)}{\|\mathbf{v}_2(n)\|} \quad (4.6)$$

onde $\hat{\mathbf{R}}_s = \mathbf{s}(n)\mathbf{s}^H(n)$ e $\hat{\mathbf{R}}_{u_k} = \mathbf{u}_k(n)\mathbf{u}_k^H(n)$ são estimativas instantâneas de \mathbf{R}_{IC} e \mathbf{R}_{CC} , respectivamente. Em (4.4) e (4.5), μ_s e μ_u são os passos de adaptação. O vetor de pesos é normalizado a cada atualização, mantendo o termo $\mathbf{w}^H \mathbf{I} \mathbf{w}$ em (4.3) invariante.

4.3. MODELAGEM ESTATÍSTICA DO ALGORITMO CSG

Nesta seção, é desenvolvido um modelo estocástico que descreve o comportamento do vetor de pesos, sendo utilizado para determinar expressões para a SINR. Para tal, é assumido que os elementos dos vetores de entrada $\mathbf{s}(n)$ e $\mathbf{u}_k(n)$, para $k=1,2$, são amostras de um processo estocástico. Em consequência, a mesma suposição é considerada para o vetor de pesos.

4.3.1. MOMENTO DE PRIMEIRA ORDEM DO VETOR DE PESOS

O primeiro passo para se obter o modelo que descreve o comportamento dos pesos é aplicar o operador valor esperado em (4.6). Assim,

$$E[\mathbf{w}(n+1)] = E\left[\frac{\mathbf{v}_2(n)}{\|\mathbf{v}_2(n)\|}\right]. \quad (4.7)$$

O cálculo do valor esperado (4.7) não é uma tarefa trivial. Contudo, considerando que a evolução de $\mathbf{v}_2(n)$ é suave (condição de dispersão pequena, isto é, quando o desvio padrão é muito menor do que a média), a seguinte aproximação pode ser aplicada [37]:

$$E[\mathbf{w}(n+1)] \cong \frac{E[\mathbf{v}_2(n)]}{\|E[\mathbf{v}_2(n)]\|}. \quad (4.8)$$

Uma dispersão pequena em $\mathbf{v}_2(n)$ é garantida na condição de passo de adaptação pequeno. O vetor de pesos e o sinal recebido são assumidos independentes, o qual é também justificado na condição de passo de adaptação pequeno [8]. Então, tomando o valor esperado de ambos os lados das expressões (4.4) e (4.5), obtém-se

$$E[\mathbf{v}_1(n)] = E[\mathbf{w}(n)] + \mu_s (\mathbf{I} - \mathbf{R}_{u_1}^N - \mathbf{R}_{u_2}^N) \mathbf{R}_s E[\mathbf{w}(n)] \quad (4.9)$$

e

$$E[\mathbf{v}_2(n)] = E[\mathbf{v}_1(n)] - \mu_u (\mathbf{I} - \mathbf{R}_s^N) (\mathbf{R}_{u_1} + \mathbf{R}_{u_2}) E[\mathbf{w}(n)] \quad (4.10)$$

onde $\mathbf{R}_s^N = E\left[\frac{\mathbf{s}(n)\mathbf{s}^H(n)}{\mathbf{s}^H(n)\mathbf{s}(n)}\right]$ e $\mathbf{R}_{u_k}^N = E\left[\frac{\mathbf{u}_k(n)\mathbf{u}_k^H(n)}{\mathbf{u}_k^H(n)\mathbf{u}_k(n)}\right]$ são matrizes de covariância normalizadas e $\mathbf{R}_s = E[\hat{\mathbf{R}}_s(n)]$ e $\mathbf{R}_{u_k} = E[\hat{\mathbf{R}}_{u_k}(n)]$. O cálculo das matrizes \mathbf{R}_s^N e $\mathbf{R}_{u_k}^N$ é apresentado no Apêndice 3.

Agora, substituindo (4.9) em (4.10) e a expressão resultante em (4.8), obtém-se a expressão final para o momento de primeira ordem do vetor de pesos. Assim,

$$E[\mathbf{w}(n+1)] = \frac{\mathbf{A}E[\mathbf{w}(n)]}{\sqrt{E[\mathbf{w}^H(n)]\mathbf{B}E[\mathbf{w}(n)]}} \quad (4.11)$$

com

$$\mathbf{A} = [\mathbf{I} + \mu_s (\mathbf{I} - \mathbf{R}_{u_1}^N - \mathbf{R}_{u_2}^N) \mathbf{R}_s - \mu_u (\mathbf{I} - \mathbf{R}_s^N) (\mathbf{R}_{u_1} + \mathbf{R}_{u_2})] \quad (4.12)$$

e

$$\mathbf{B} = \mathbf{A}^H \mathbf{A}. \quad (4.13)$$

A expressão (4.11) também pode ser interpretada através do método da potência, utilizado para obter o autovetor associado ao autovalor dominante de uma matriz diagonalizável. Assim, o valor em regime permanente para o vetor de pesos médio é o autovetor associado ao autovalor dominante de \mathbf{A} [38].

Neste ponto, é importante observar que a análise do vetor de erro nos pesos, definida como a diferença entre o valor atual do vetor de pesos e os pesos ótimos, não pode ser aqui aplicada, visto que para o caso de arranjo de antenas não existe somente um vetor de pesos ótimo [25].

4.3.2. RAZÃO SINAL/RUÍDO-MAIS-INTERFERÊNCIA (SINR)

Maximizar a SINR é o objetivo fundamental do arranjo adaptativo no problema aqui tratado e, portanto, o comportamento da SINR é um importante indicador do desempenho do algoritmo adaptativo. Por definição, a SINR é dada por

$$\gamma = \frac{E[P_{IC}(n)]}{E[P_{CC}(n)] + \sigma_\eta^2} \quad (4.14)$$

onde $P_{IC}(n)$ é a potência instantânea do sinal útil, $P_{CC}(n)$, a potência instantânea do sinal interferente e σ_η^2 , a potência do ruído aditivo. Por simplicidade, aqui será considerado $\sigma_\eta^2 = 1$. Assumindo que o vetor de pesos e os sinais de enlace direto são independentes, a SINR pode ser expressa como

$$\gamma(n) = \frac{E[\mathbf{w}^H(n) \mathbf{R}_{IC} \mathbf{w}(n)]}{E[\mathbf{w}^H(n) \mathbf{R}_{CC} \mathbf{w}(n)] + 1}. \quad (4.15)$$

A expressão (4.15) pode ser reescrita por

$$\gamma(n) = \frac{\text{tr}[\mathbf{R}_s \mathbf{K}(n)]}{\text{tr}[\mathbf{R}_u \mathbf{K}(n)] + 1} \quad (4.16)$$

onde $\mathbf{K}(n) = E[\mathbf{w}(n)\mathbf{w}^H(n)]$ é o momento de segunda ordem do vetor de pesos.

De (4.16), conclui-se que conhecendo o comportamento do momento de segunda ordem do vetor de pesos pode-se também conhecer o comportamento da SINR.

Uma das características do algoritmo CSG é a pouca variabilidade obtida entre diferentes realizações. Então, um modelo que descreva o comportamento do algoritmo deve ser simples o suficiente para justificar sua aplicação no lugar de simulações de Monte Carlo. Visando a simplicidade do modelo, a seguinte aproximação é considerada:

$$E[\mathbf{K}(n)] = E[\mathbf{w}(n)\mathbf{w}^H(n)] \cong E[\mathbf{w}(n)]E[\mathbf{w}^H(n)]. \quad (4.17)$$

Tal aproximação é bastante utilizada na análise do algoritmo LMS, apresentando resultados aceitáveis [39], sendo válida para passos de adaptação pequenos. Assim, o comportamento da SINR é determinado a partir de

$$\gamma(n) \cong \frac{E[\mathbf{w}^H(n)]\mathbf{R}_{IC}E[\mathbf{w}(n)]}{E[\mathbf{w}^H(n)]\mathbf{R}_{CC}E[\mathbf{w}(n)] + 1}. \quad (4.18)$$

Finalmente, para obter (4.18) pode ser utilizado o modelo do momento de primeira ordem do vetor de pesos (4.11).

4.4. RESULTADOS DE SIMULAÇÃO

Nesta seção, são apresentadas comparações entre resultados obtidos por simulações de Monte Carlo (MC) e a partir do modelo proposto. Os resultados são divididos em três grupos: FDP da SINR em regime permanente, convergência da SINR e convergência dos pesos. Em todas as simulações desta seção são consideradas duas fontes interferentes e as matrizes de covariância são obtidas utilizando o modelo de sinal apresentado em [25].

4.4.1. FUNÇÃO DE DISTRIBUIÇÃO DE PROBABILIDADES DA SINR

Para o enlace reverso, é assumido que o ângulo de chegada do sinal do terminal móvel da própria célula θ_s e os ângulos de chegada dos sinais interferentes, θ_{u_1} e θ_{u_2} , são tomados em valores discretos (uniformemente espaçados e quantizados a cada 4°) dos conjuntos $\Theta_s = (-60^\circ, 60^\circ)$, $\Theta_{u_1} = (-36^\circ, -12^\circ)$ e $\Theta_{u_2} = (12^\circ, 36^\circ)$, respectivamente. Para cada combinação de ângulo de chegada, duas matrizes \mathbf{K} em regime permanente são determinadas, uma por simulações de MC e outra utilizando o modelo proposto. As matrizes \mathbf{K} obtidas são substituídas em (4.16), obtendo assim as correspondentes SINR. O conjunto resultante de SINR é utilizado para calcular a FDP segundo o método proposto em [25], tanto para os resultados obtidos por simulações de MC (200 realizações independentes) quanto para os obtidos pelo modelo proposto. São consideradas quatro situações resultantes da combinação de dois valores de abertura de ângulo de chegada, $\Delta_s = \Delta_u = 3^\circ$ e 30° , e dois valores para o número de elementos por arranjo, $N = 4$ e $N = 8$. O conjunto de parâmetros para os quatro casos é: $I = 12$ caminhos de desvanecimento independentes e passos de adaptação $\mu_s = \mu_u = 0,01$. Para os sinais de enlace direto, é considerada uma razão sinal-ruído $SNR = 36$ dB e uma razão interferência-ruído $INR = 15$ dB, representando uma situação prática usual [25]. A Figura 4.3 mostra uma predição muito boa obtida com o modelo proposto. Cabe aqui ressaltar que para obter a FDP é necessário considerar diversas condições de operação do algoritmo e, portanto, a proximidade entre as curvas da FDP obtidas por simulação e através do modelo proposto reflete uma predição adequada em cada uma das situações consideradas. Adicionalmente, pode-se observar que as curvas da FDP para oito elementos estão concentradas mais à direita do gráfico do que as correspondentes FDP para quatro elementos. Esse fato indica um melhor desempenho do arranjo para um maior número de elementos.

4.4.2. CONVERGÊNCIA DA SINR

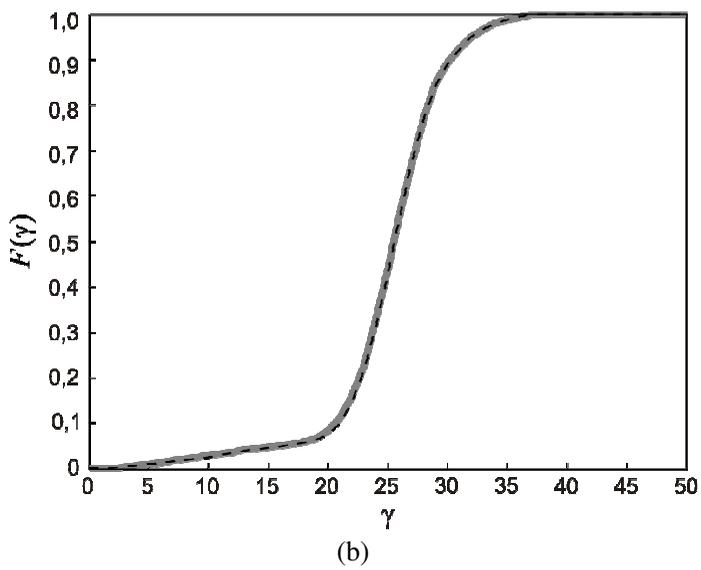
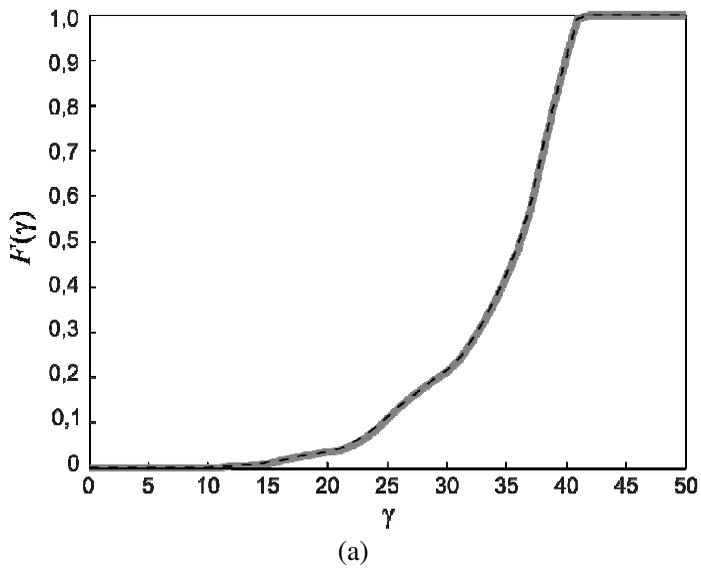
Para avaliar a convergência da SINR, denotada por $\gamma(n)$, são consideradas as mesmas quatro situações descritas anteriormente; porém, é selecionada apenas uma combinação de ângulos de chegada

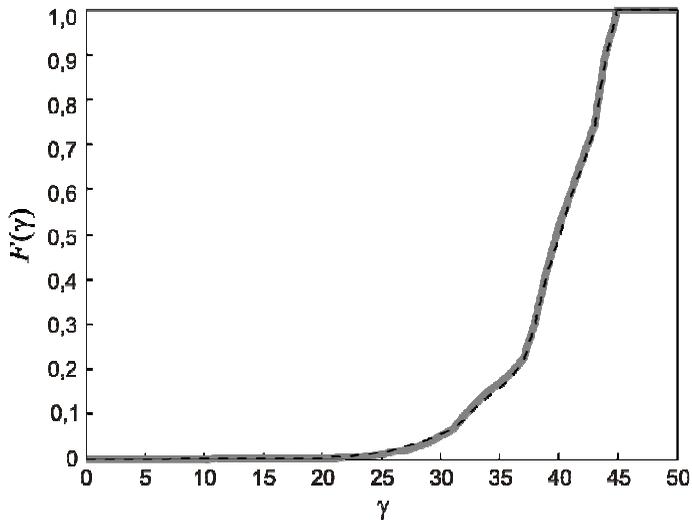
θ_s , θ_{u_1} e θ_{u_2} . Também, o valor do passo de adaptação é modificado, visando obter o maior valor de SINR em regime permanente para a combinação de ângulos considerada. Os demais parâmetros são mantidos. A Figura 4.4(a) mostra os resultados de simulações de MC e do modelo proposto para valores de abertura de ângulo de chegada iguais a $\Delta_s = \Delta_u = 3^\circ$ para $N = 4$ e $N = 8$. Para o caso de $N = 4$, é utilizado $\mu_s = \mu_u = 0,05 = 0,1\mu_{\max}$, onde μ_{\max} é o passo de adaptação máximo que pode ser considerado sem que se tenha divergência do algoritmo, e, para $N = 8$, é considerado $\mu_s = \mu_u = 0,01 = 0,1\mu_{\max}$. A combinação de ângulos de chegada selecionados são: $\theta_s = 60^\circ$, $\theta_{u_1} = -16^\circ$ e $\theta_{u_2} = 36^\circ$. A Figura 4.4(b) apresenta os resultados para $\Delta_s = \Delta_u = 30^\circ$, considerando novamente duas situações: $N = 4$ e $N = 8$. Para o caso de $N = 4$, é utilizado $\mu_s = \mu_u = 0,01 = 0,1\mu_{\max}$ e, para $N = 8$, $\mu_s = \mu_u = 0,007 = 0,1\mu_{\max}$.

Nota-se, neste exemplo, um casamento satisfatório entre simulação e modelo proposto. Os pequenos descasamentos observados podem ser atribuídos à utilização da aproximação (4.17). A Figura 4.4 ilustra o comportamento do algoritmo sob uma das condições de trabalho consideradas para obter a Figura 4.3. Novamente, verifica-se um melhor comportamento do algoritmo quando utilizado um maior número de antenas no arranjo. Além disso, é ainda observada uma redução de desempenho do algoritmo com o aumento da abertura do ângulo de chegada.

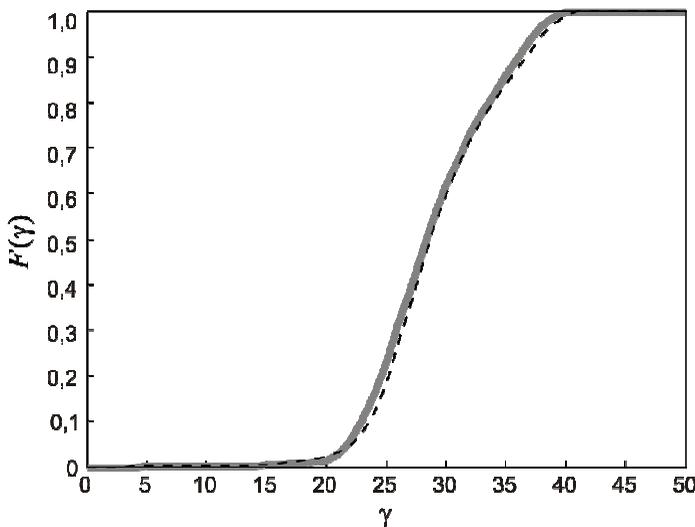
4.4.3. CONVERGÊNCIA DOS PESOS

Para avaliar a precisão do modelo do vetor de pesos, são consideradas as duas situações descritas na Figura 4.4(a), mostrando, em cada caso, os resultados de simulações de MC e do modelo proposto, tanto para a parte real quanto para a parte imaginária dos pesos. Todos esses resultados são verificados na Figura 4.5, onde é constatado um excelente casamento tanto em regime transiente quanto em regime permanente. Esse fato é consequência da forma de cálculo das matrizes de covariância normalizadas \mathbf{R}_i^N considerada. Qualquer imprecisão no cálculo das matrizes de covariância normalizadas levaria o modelo a valores em regime permanente diferentes dos obtidos por simulação.



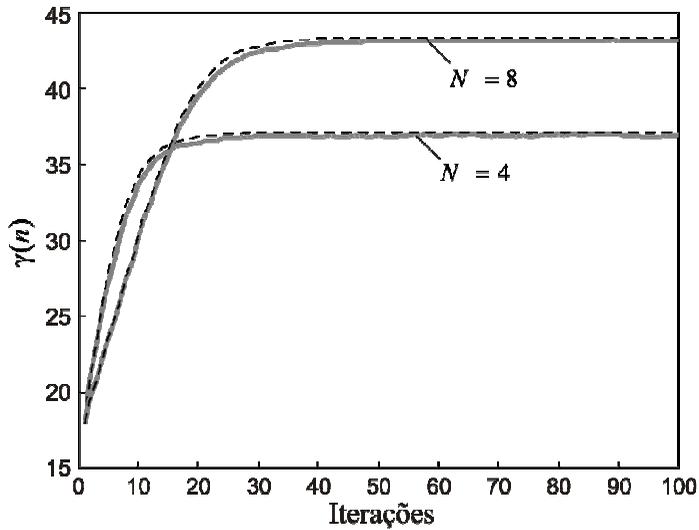


(c)

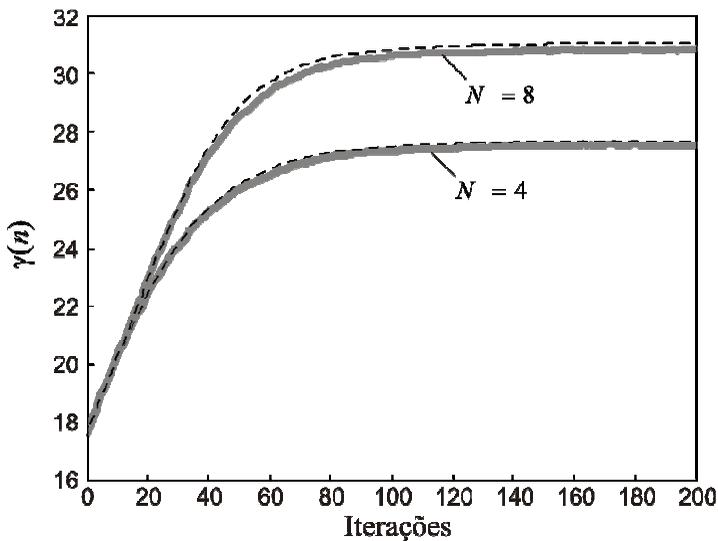


(d)

Figura 4.3. Comportamento da função distribuição de probabilidade da SINR. Simulações de MC (linha cinza); modelo proposto (linha tracejada escura). (a) FDP para quatro elementos e $\Delta_s = \Delta_u = 3^\circ$. (b) FDP para quatro elementos e $\Delta_s = \Delta_u = 30^\circ$. (c) FDP para oito elementos e $\Delta_s = \Delta_u = 3^\circ$. (d) FDP para oito elementos e $\Delta_s = \Delta_u = 30^\circ$.

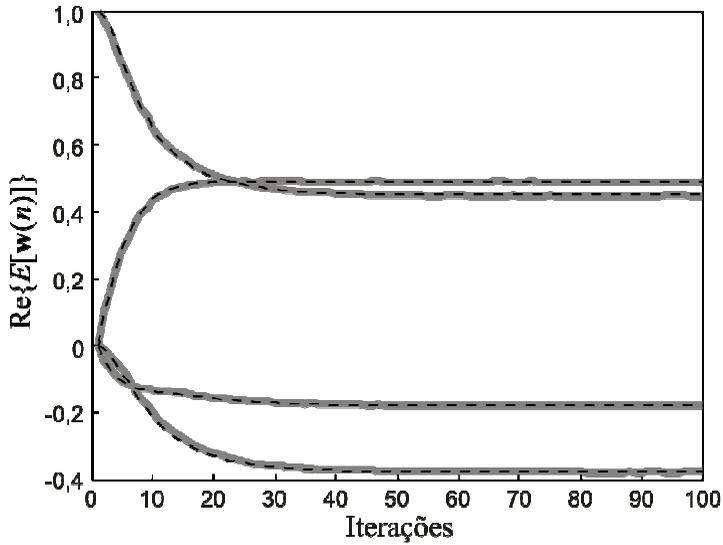


(a)

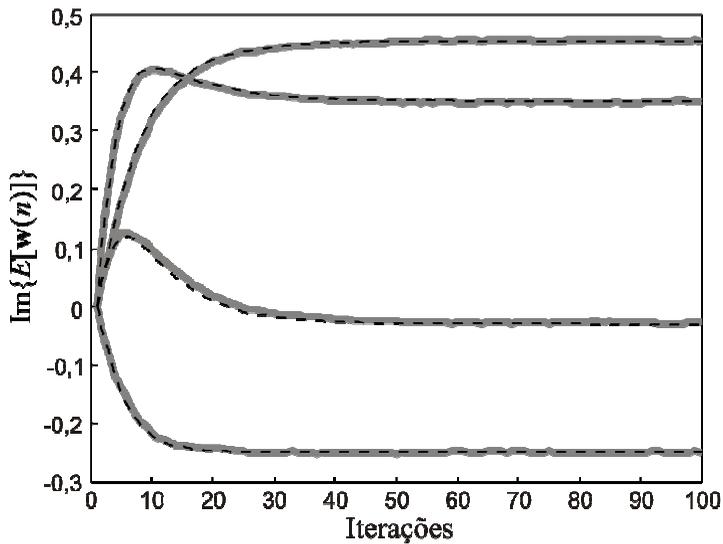


(b)

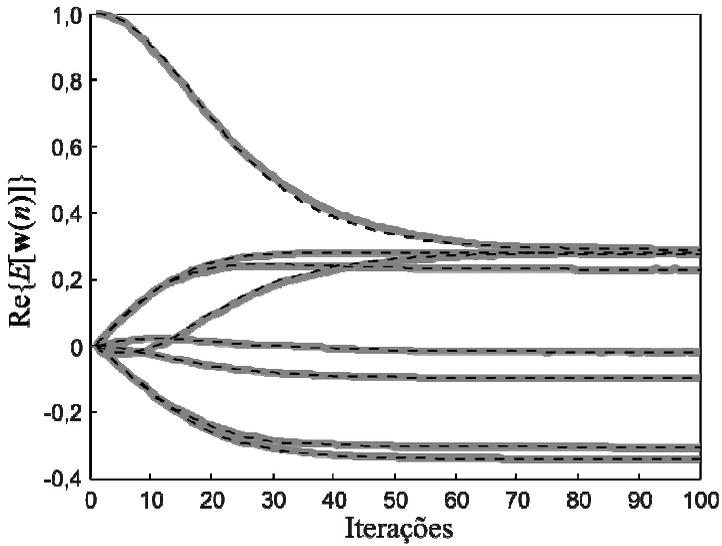
Figura 4.4 Comportamento de convergência de $\gamma(n)$. Simulações de MC (linha cinza); modelo proposto (linha tracejada escura). (a) Comportamento para $\Delta_s = \Delta_u = 3^\circ$. (b) Comportamento para $\Delta_s = \Delta_u = 30^\circ$.



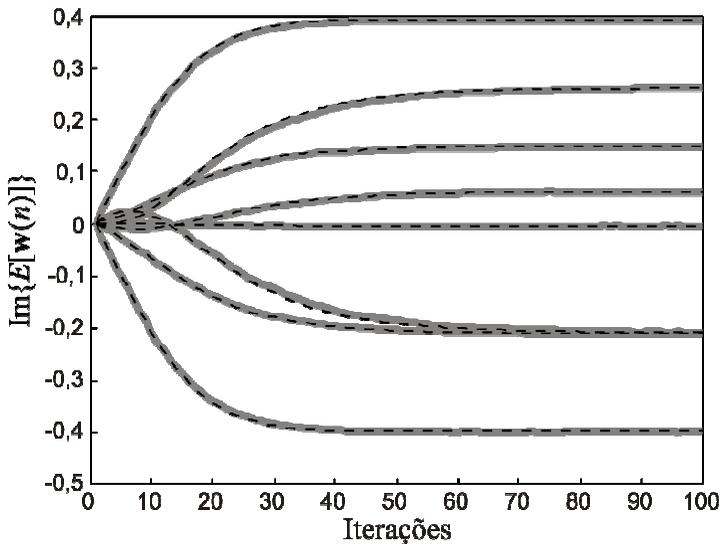
(a)



(b)



(c)



(d)

Figura 4.5. Comportamento dos pesos. Simulações de MC (linha cinza); modelo proposto (linha tracejada escura). Situação para $N = 4$: (a) Parte real dos pesos. (b) Parte imaginária dos pesos. Situação para $N = 8$: (c) Parte real dos pesos. (d) Parte imaginária dos pesos.

4.5. CONCLUSÕES

Neste capítulo, é apresentado um modelo estatístico para o momento de primeira ordem do vetor de pesos do algoritmo CSG considerando hipótese de adaptação lenta e um problema de controle de arranjos de antenas. A partir desse modelo, pode ser predito o comportamento da SINR tanto em regime transiente quanto em regime permanente. O modelo apresentado prediz satisfatoriamente o comportamento do algoritmo para uma ampla variedade de condições de trabalho, como pode ser verificado através da função de distribuição de probabilidade da SINR. Tal precisão é consequência da forma de cálculo das matrizes de covariância normalizadas, a qual é uma das contribuições deste trabalho. O estudo do algoritmo CSG utilizando o modelo proposto permitiu identificar um comportamento anômalo de seu funcionamento e, portanto, sugerir uma forma de compensá-lo, a qual é o objeto do próximo capítulo.

5.1. INTRODUÇÃO

Neste capítulo, utilizando as expressões do modelo estatístico proposto no Capítulo 4, verifica-se a existência de um comportamento anômalo do algoritmo CSG, aqui denominado comportamento não-balanceado. Tal comportamento tem origem na não maximização da função objetivo original do algoritmo CSG. As expressões que descrevem o comportamento médio dos pesos permitem obter dois importantes objetivos neste trabalho, a saber:

- i) Explicar o comportamento não-balanceado do algoritmo CSG.
- ii) Sugerir uma modificação no algoritmo CSG padrão visando compensar o comportamento não-balanceado.

O algoritmo modificado é chamado *improved* CSG (ICSG). As expressões apresentadas apenas consideram o caso de dois interferentes; contudo, o procedimento pode ser generalizado para três ou mais interferentes. Resultados de simulações numéricas são utilizados para mostrar o desempenho do algoritmo proposto.

5.2. EQUAÇÕES DE ATUALIZAÇÃO

Para facilitar a leitura desta tese, as equações de adaptação dos pesos do algoritmo CSG são reproduzidas a seguir:

$$\mathbf{v}_1(n) = \mathbf{w}(n) + \mu_s \left[\mathbf{I} - \frac{\hat{\mathbf{R}}_{u_1}(n)}{\|\mathbf{u}_1(n)\|^2} - \frac{\hat{\mathbf{R}}_{u_2}(n)}{\|\mathbf{u}_2(n)\|^2} \right] \hat{\mathbf{R}}_s(n) \mathbf{w}(n) \quad (5.1)$$

e

$$\mathbf{v}_2(n) = \mathbf{v}_1(n) - \mu_u \left[\mathbf{I} - \frac{\hat{\mathbf{R}}_s(n)}{\|\mathbf{s}(n)\|^2} \right] [\hat{\mathbf{R}}_{u_1}(n) + \hat{\mathbf{R}}_{u_2}(n)] \mathbf{w}(n) \quad (5.2)$$

com

$$\mathbf{w}(n+1) = \frac{\mathbf{v}_2(n)}{\|\mathbf{v}_2(n)\|} \quad (5.3)$$

onde $\hat{\mathbf{R}}_s = \mathbf{s}(n)\mathbf{s}^H(n)$ e $\hat{\mathbf{R}}_{u_k} = \mathbf{u}_k(n)\mathbf{u}_k^H(n)$ denotam matrizes de covariância estimadas. Os parâmetros μ_s e μ_u são os correspondentes passos de adaptação do algoritmo. O vetor de pesos é normalizado a cada iteração, mantendo o termo $\mathbf{w}^H(n)\mathbf{I}\mathbf{w}(n)$ da função custo constante.

5.3. COMPORTAMENTO NÃO-BALANCEADO

O propósito desta seção é identificar, através de exemplos, a degradação de desempenho do algoritmo CSG para algumas condições especiais de operação. Posteriormente, mostra-se que tal fenômeno tem uma explicação teórica, decorrente de uma combinação de dois efeitos: (a) um ou mais ângulos de chegada do sinal interferente apresentam valores numéricos próximos ao do ângulo de chegada do sinal de enlace reverso; e (b) os ângulos de dispersão dos sinais envolvidos são pequenos. Em todas as simulações apresentadas neste capítulo, as matrizes de covariância são determinadas segundo o procedimento discutido no Capítulo 4, considerando doze caminhos de dispersão independentes ($I=12$). Para os sinais de enlace direto, é considerada uma razão sinal-ruído $SNR=36$ dB e uma razão interferência-ruído $INR=15$ dB, representando uma situação prática usual [25]. Para as simulações de Monte Carlo (MC), são consideradas 100 realizações independentes.

Definidas as condições de operação, primeiramente, é considerado um caso no qual o algoritmo CSG tem um comportamento padrão e, em seguida, outra situação ocorrendo um comportamento não-balanceado.

5.3.1. CASO I: θ_s , θ_{u_1} e θ_{u_2} DISTANTES

Para o canal de enlace reverso, o ângulo de chegada do sinal é $\theta_s = 60^\circ$ e os ângulos de chegada dos interferentes são $\theta_{u_1} = -12^\circ$ e $\theta_{u_2} = 36^\circ$. Outros parâmetros são: dispersão do ângulo de chegada

$\Delta_s = \Delta_u = 3^\circ$, número de elementos do arranjo $N = 4$, passos de adaptação $\mu_s = \mu_u = 0,05 = 0,1\mu_{\max}$ (veja Figura 5.1).

5.3.2. CASO II: $\theta_s = \theta_{u_1}$

Os parâmetros utilizados são os mesmos que os considerados no Caso I, exceto que agora $\theta_s = \theta_{u_1} = -12^\circ$. A Figura 5.1 mostra a evolução da SINR do algoritmo CSG para os Casos I e II avaliados. Note que há uma nítida degradação de desempenho do algoritmo, observada pela curva da SINR para as condições definidas no Caso II. Na Figura 5.2, apenas a potência radiada para o terminal móvel (TM) dentro da própria célula é mostrada. A Figura 5.3 apresenta o diagrama de radiação do arranjo de antenas para o Caso II, considerando que o algoritmo adaptativo atingiu o regime permanente. De acordo com essa figura, observa-se que, nas condições para o Caso II, o TM não recebe potência suficiente da ERB, indicando que o serviço foi interrompido.

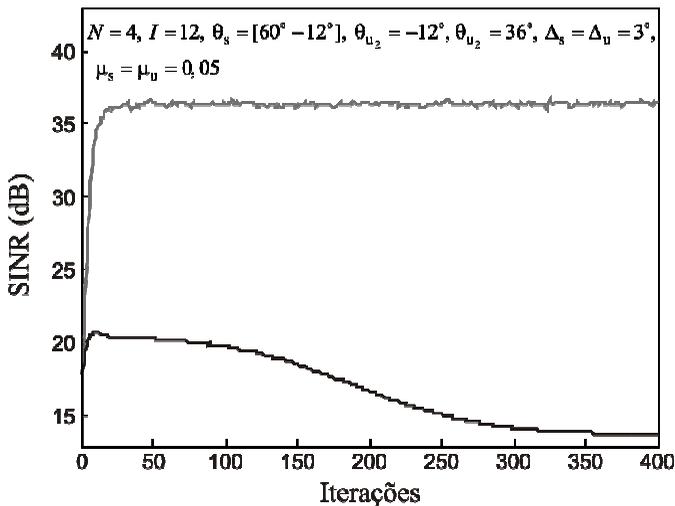


Figura 5.1. Evolução da SINR para os casos considerados. (Linha cinza) Caso I. (Linha escura) Caso II.

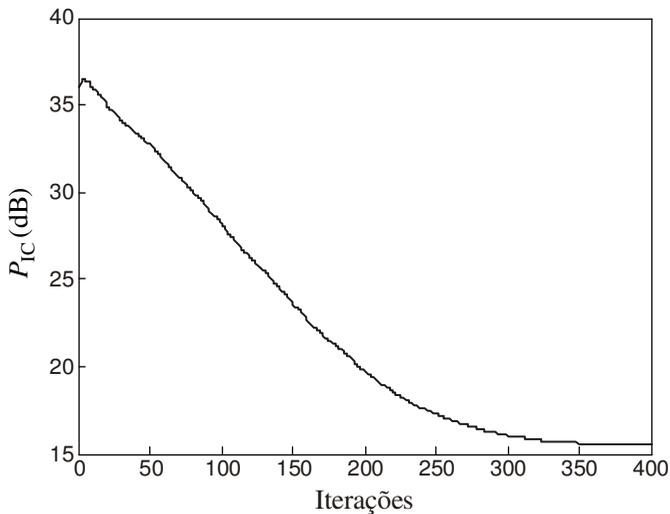


Figura 5.2. Evolução da P_{IC} para o Caso II.

$$N = 4, I = 12, \theta_s = -12^\circ, \theta_{u_1} = -12^\circ, \theta_{u_2} = 36^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,05$$

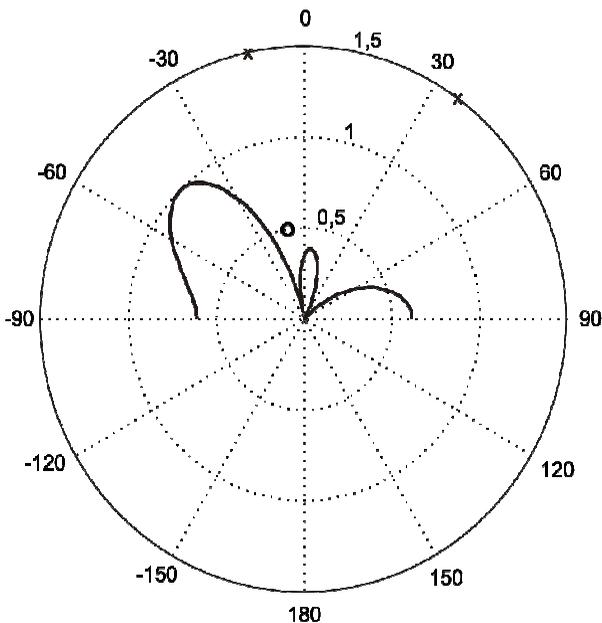


Figura 5.3. Caso II. Diagrama de radiação. Terminal móvel da própria célula (o). Interferentes (x).

5.4. ANÁLISE

Nesta seção, é discutida a causa da degradação do algoritmo CSG, usando o modelo do comportamento médio dos pesos do referido algoritmo apresentado no Capítulo 4. A partir dessa discussão, é proposta uma modificação no algoritmo CSG, melhorando significativamente o seu desempenho para algumas condições especiais de operação.

5.4.1. COMPORTAMENTO MÉDIO DOS PESOS

As expressões do modelo que descrevem o comportamento médio dos pesos são reproduzidas a seguir:

$$E[\mathbf{w}(n+1)] \cong \frac{E[\mathbf{v}_2(n)]}{\|E[\mathbf{v}_2(n)]\|} \quad (5.4)$$

sendo

$$E[\mathbf{v}_2(n)] = E[\mathbf{v}_1(n)] - \mu_u (\mathbf{I} - \mathbf{R}_s^N) (\mathbf{R}_{u_1} + \mathbf{R}_{u_2}) E[\mathbf{w}(n)] \quad (5.5)$$

e

$$E[\mathbf{v}_1(n)] = E[\mathbf{w}(n)] + \mu_s (\mathbf{I} - \mathbf{R}_{u_1}^N - \mathbf{R}_{u_2}^N) \mathbf{R}_s E[\mathbf{w}(n)] \quad (5.6)$$

onde $\mathbf{R}_s^N = E \left[\frac{\mathbf{s}(n)\mathbf{s}^H(n)}{\mathbf{s}^H(n)\mathbf{s}(n)} \right]$ e $\mathbf{R}_{u_k}^N = E \left[\frac{\mathbf{u}_k(n)\mathbf{u}_k^H(n)}{\mathbf{u}_k^H(n)\mathbf{u}_k(n)} \right]$ são as matrizes

de covariância amostral normalizada.

Então, substituindo (5.6) em (5.5) e a expressão resultante em (5.4), obtém-se a seguinte expressão para o comportamento médio dos pesos:

$$E[\mathbf{w}(n+1)] = \frac{\mathbf{A} E[\mathbf{w}(n)]}{\sqrt{E[\mathbf{w}^H(n)] \mathbf{A}^H \mathbf{A} E[\mathbf{w}(n)]}} \quad (5.7)$$

com

$$\mathbf{A} = \mathbf{I} + \mu_s (\mathbf{I} - \mathbf{R}_{u_1}^N - \mathbf{R}_{u_2}^N) \mathbf{R}_s - \mu_u (\mathbf{I} - \mathbf{R}_s^N) (\mathbf{R}_{u_1} + \mathbf{R}_{u_2}). \quad (5.8)$$

5.4.2. DISCUSSÃO SOBRE O COMPORTAMENTO NÃO-BALANCEADO

O Caso II, apresentado na Seção 5.3, indica a existência de uma perda evidente de desempenho do algoritmo CSG para algumas condições especiais de operação. Nesta seção, esse comportamento anômalo do algoritmo é discutido.

O algoritmo adaptativo, que controla o arranjo de antenas, deve ajustar os pesos do arranjo de forma que a potência radiada para o TM da própria célula seja maximizada e que a potência radiada para os TMs de cocanal, minimizada. Assim, quando ocorre o comportamento não-balanceado, o algoritmo CSG prioriza a minimização da P_{CC} em detrimento da maximização da P_{IC} . Com a ajuda do modelo do algoritmo, é possível entender tal comportamento, o qual é discutido a seguir.

No caso em que o sinal $s(n)$ tem o mesmo ângulo de chegada do sinal interferente $u_1(n)$ e os ângulos de dispersão dos sinais envolvidos são semelhantes, as matrizes de covariância \mathbf{R}_s e \mathbf{R}_{u_1} são aproximadamente iguais. O mesmo ocorre para as matrizes de covariância das amostras normalizadas \mathbf{R}_s^N e $\mathbf{R}_{u_1}^N$. Assim, substituindo \mathbf{R}_{u_1} por \mathbf{R}_s e $\mathbf{R}_{u_1}^N$ por \mathbf{R}_s^N em (5.6) e após aplicar a propriedade distributiva no segundo termo, obtém-se

$$E[\mathbf{v}_1(n)] \cong E[\mathbf{w}(n)] + \mu_s \{ \mathbf{R}_s E[\mathbf{w}(n)] - \mathbf{R}_s^N \mathbf{R}_s E[\mathbf{w}(n)] - \mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)] \}. \quad (5.9)$$

Considerando que qualquer matriz \mathbf{R} e sua correspondente versão amostral normalizada \mathbf{R}^N têm os mesmos autovetores (veja Apêndice 3), então, para cada produto da forma $\mathbf{R}^N \mathbf{R}$, podemos escrever

$$\mathbf{R}^N \mathbf{R} = \mathbf{Q} \mathbf{\Lambda}^N \mathbf{Q}^H \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^H = \mathbf{Q} \mathbf{\Lambda}^N \mathbf{\Lambda} \mathbf{Q}^H \quad (5.10)$$

onde \mathbf{Q} é a matriz de autovetores de \mathbf{R} , e $\mathbf{\Lambda}$ e $\mathbf{\Lambda}^N$ são matrizes diagonais contendo os autovalores λ_i e λ_i^N (para $i=1, 2, \dots, N$) de \mathbf{R}

e \mathbf{R}^N , respectivamente. Quando a dispersão do ângulo de chegada for pequena (por exemplo, $\Delta = 3^\circ$), as matrizes de covariância obtidas a partir do modelo de sinal utilizado em [25] têm uma grande dispersão entre seus autovalores, resultando em um autovalor dominante λ_1 , isto é, $\lambda_1 \gg \lambda_2 > \dots > \lambda_N$. Por sua vez, a matriz \mathbf{R}^N tem também um autovalor dominante cujo valor é ligeiramente menor do que 1 (veja Apêndice 4). Como resultado, podemos considerar as seguintes aproximações: $\mathbf{\Lambda}^N \mathbf{\Lambda} \cong \alpha \mathbf{\Lambda}$ e $\mathbf{R}^N \mathbf{R} \cong \alpha \mathbf{R}$, para $0 < \alpha < 1$, obtendo-se

$$\mathbf{R}_s^N \mathbf{R}_s E[\mathbf{w}(n)] \cong \alpha \mathbf{R}_s E[\mathbf{w}(n)]. \quad (5.11)$$

Substituindo (5.11) em (5.9) resulta

$$E[\mathbf{v}_1(n)] \cong E[\mathbf{w}(n)] + \mu_s \{ (1 - \alpha) \mathbf{R}_s E[\mathbf{w}(n)] - \mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)] \}. \quad (5.12)$$

Analisando (5.12) observa-se:

- i) O termo $\mathbf{R}_s E[\mathbf{w}(n)]$ objetiva maximizar a potência radiada ao TM da própria célula. Isso é inferido do fato que o gradiente do numerador da função custo (2.1) é $\mathbf{R}_{IC} E[\mathbf{w}(n)]$ e \mathbf{R}_s é equivalente a \mathbf{R}_{IC} .
- ii) O termo $\mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)]$ evita radiar potência ao TM de cocanal. Esta característica é intrínseca à natureza do algoritmo.

Como resultado, o termo $\mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)]$ em (5.12) é dominante quando comparado com $(1 - \alpha) \mathbf{R}_s E[\mathbf{w}(n)]$, sendo que sob as condições do comportamento não-balanceado, α torna-se aproximadamente igual a 1. Nesse caso, constata-se através das expressões do modelo que o algoritmo CSG dá mais importância à minimização da interferência de cocanal, não entregando potência suficiente para o TM da própria célula. Um efeito similar é observado quando $\mathbf{R}_{u_2} \cong \mathbf{R}_s$ ou $\mathbf{R}_{u_1} \cong \mathbf{R}_{u_2} \cong \mathbf{R}_s$.

5.4.3. ALGORITMO CSG MODIFICADO

Nesta seção, uma modificação no algoritmo CSG é proposta. O algoritmo modificado é denominado *improved* CSG (ICSG), compensando o comportamento não-balanceado do algoritmo CSG padrão discutido anteriormente. Assim, considerando as expressões do modelo para o caso crítico $\mathbf{R}_{u_1} \cong \mathbf{R}_s$, uma possível compensação para o comportamento não-balanceado consiste na redução do efeito de $\mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)]$. Tal redução deve ser proporcional à diferença entre $\mathbf{R}_s E[\mathbf{w}(n)]$ e $\mathbf{R}_{u_1}^N \mathbf{R}_s E[\mathbf{w}(n)]$. Uma forma direta para fazer isso é incluir o fator $(1 - \alpha)$ multiplicando o termo $\mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)]$ em (5.12). Assim, obtém-se a expressão compensada como

$$E[\mathbf{v}_1(n)] \cong E[\mathbf{w}(n)] + \mu_s \{ (1 - \alpha) \mathbf{R}_s E[\mathbf{w}(n)] - (1 - \alpha) \mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)] \}. \quad (5.13)$$

Esta solução é determinada utilizando as expressões do modelo. Agora, devemos determinar como as equações de atualização do algoritmo CSG devem ser modificadas para incluir o efeito da compensação. Dessa forma, será utilizado o caminho inverso ao seguido para obter (5.12). Assim,

$$\alpha \mathbf{R}_{u_2}^N \mathbf{R}_s \cong \mathbf{R}_{u_2}^N \mathbf{R}_s^N \mathbf{R}_s \quad (5.14)$$

e

$$\mathbf{R}_{u_2}^N \mathbf{R}_s^N \mathbf{R}_s \cong \mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N \mathbf{R}_s. \quad (5.15)$$

Então, substituindo (5.15) em (5.14) e a expressão resultante em (5.13), obtém-se

$$E[\mathbf{v}_1(n)] = E[\mathbf{w}(n)] + \mu_s \{ \mathbf{R}_s E[\mathbf{w}(n)] - \mathbf{R}_{u_1}^N \mathbf{R}_s E[\mathbf{w}(n)] - \mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)] + \mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N \mathbf{R}_s E[\mathbf{w}(n)] \}. \quad (5.16)$$

Portanto, a inclusão do vetor $\mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N \mathbf{R}_s E[\mathbf{w}(n)]$ em (5.6) compensa o comportamento não-balanceado quando $\mathbf{R}_{u_1} \cong \mathbf{R}_s$.

Similarmente, para o caso $\mathbf{R}_{u_2} \cong \mathbf{R}_s$, o termo $\mathbf{R}_{u_1}^N \mathbf{R}_{u_2}^N \mathbf{R}_s E[\mathbf{w}(n)]$ deve também ser incluído na expressão de atualização (5.6), obtendo-se

$$E[\mathbf{v}_1(n)] = E[\mathbf{w}(n)] + \mu_s (\mathbf{I} - \mathbf{R}_{u_1}^N - \mathbf{R}_{u_2}^N + \mathbf{R}_{u_1}^N \mathbf{R}_{u_2}^N + \mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N) \mathbf{R}_s E[\mathbf{w}(n)]. \quad (5.17)$$

Em situações não críticas, o produto de matrizes $\mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N \mathbf{R}_s$ e $\mathbf{R}_{u_1}^N \mathbf{R}_{u_2}^N \mathbf{R}_s$ pode ser desprezado em comparação com \mathbf{R}_s , $\mathbf{R}_{u_1}^N \mathbf{R}_s$ e $\mathbf{R}_{u_2}^N \mathbf{R}_s$, levando a um comportamento similar ao algoritmo CSG padrão. Visto que um melhor desempenho é verificado considerando as expressões do modelo (5.17), podemos então formular o algoritmo ICSG incluindo estimativas instantâneas das matrizes $\mathbf{R}_{u_1}^N \mathbf{R}_{u_2}^N$ e $\mathbf{R}_{u_2}^N \mathbf{R}_{u_1}^N$ nas expressões do algoritmo CSG. Portanto, (5.1) é modificada para [40]

$$\mathbf{v}_1(n) = \mathbf{w}(n) + \mu_s \left[\mathbf{I} - \frac{\hat{\mathbf{R}}_{u_1}(n)}{\|\mathbf{u}_1(n)\|^2} - \frac{\hat{\mathbf{R}}_{u_2}(n)}{\|\mathbf{u}_2(n)\|^2} + \frac{\hat{\mathbf{R}}_{u_1}(n)}{\|\mathbf{u}_1(n)\|^2} \cdot \frac{\hat{\mathbf{R}}_{u_2}(n)}{\|\mathbf{u}_2(n)\|^2} + \frac{\hat{\mathbf{R}}_{u_2}(n)}{\|\mathbf{u}_2(n)\|^2} \cdot \frac{\hat{\mathbf{R}}_{u_1}(n)}{\|\mathbf{u}_1(n)\|^2} \right] \hat{\mathbf{R}}_s(n) \mathbf{w}(n) \quad (5.18)$$

com as expressões (5.2) e (5.3) permanecendo inalteradas.

O algoritmo proposto também melhora o desempenho do algoritmo CSG padrão para $\mathbf{R}_{u_1} \cong \mathbf{R}_{u_2} \cong \mathbf{R}_s$. Note que o algoritmo ICSG tem carga computacional similar à do algoritmo CSG, com ordem de complexidade $O(N)$.

5.5. DESEMPENHO DO ALGORITMO ICSG

Esta seção avalia o desempenho do algoritmo ICSG comparando-o com aquele do algoritmo CSG padrão. Assim, três exemplos são apresentados. Para cada algoritmo, as curvas obtidas a partir de simulações de MC e através de predições do modelo são mostradas objetivando validar a abordagem proposta.

5.5.1. EXEMPLO 1

Neste exemplo, o caso $\mathbf{R}_{u_1} \cong \mathbf{R}_s$ é considerado. Os parâmetros utilizados são $N = 4$, $\theta_s = \theta_{u_1} = -12^\circ$, $\theta_{u_2} = 36^\circ$, $\Delta_s = \Delta_u = 3^\circ$ e $\mu_s = \mu_u = 0,05 = 0,1\mu_{\max}$. A Figura 5.4 mostra a evolução da SINR para os algoritmos CSG e ICSG. Na Figura 5.5, o diagrama de radiação do arranjo de antenas é mostrado considerando que os pesos do arranjo tenham atingido o regime permanente.

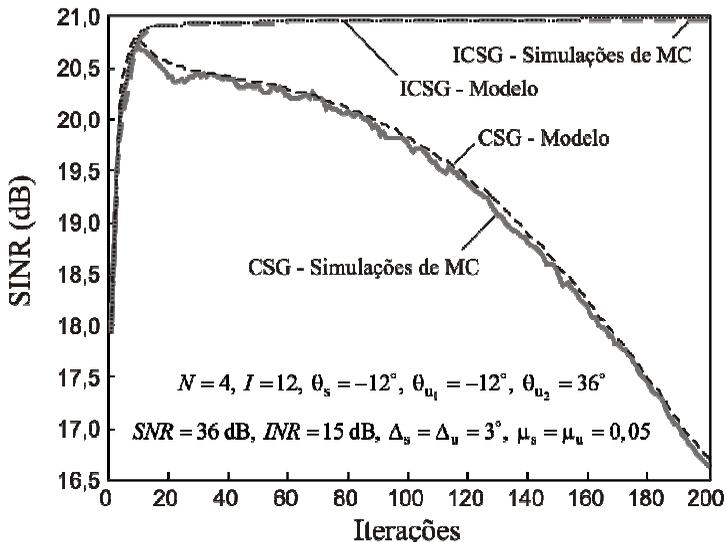
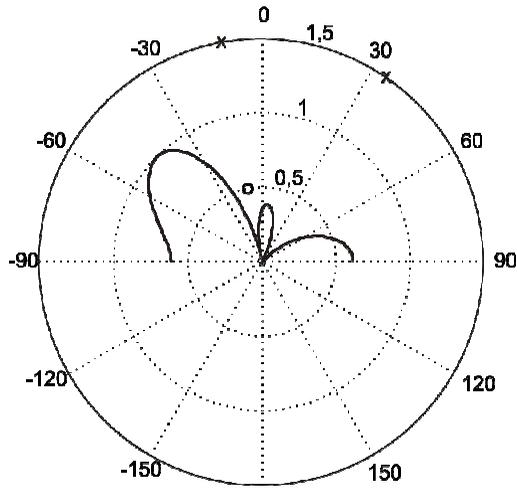


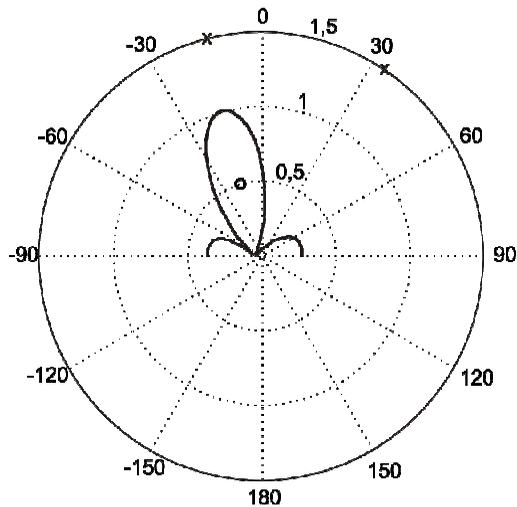
Figura 5.4. Exemplo 1. Curva da SINR. Algoritmo CSG: (linhas cinza) simulações de MC; (linhas tracejadas escuras) predições do modelo. Algoritmo ICSG: (linhas tracejadas cinza) simulações de MC; (linhas pontilhadas escuras) predições do modelo.

$M = 4, I = 12, \theta_s = -12^\circ, \theta_{u_1} = -12^\circ, \theta_{u_2} = 36^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,05$



(a)

$M = 4, I = 12, \theta_s = -12^\circ, \theta_{u_1} = -12^\circ, \theta_{u_2} = 36^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,05$



(b)

Figura 5.5. Exemplo 1. Diagrama de radiação. Terminal móvel da própria célula (o). Interferentes (x). (a) Algoritmo CSG. (b) Algoritmo ICSG.

5.5.2. EXEMPLO 2

Neste exemplo, o caso $\mathbf{R}_{u_2} \cong \mathbf{R}_s$ é avaliado. Os parâmetros utilizados são $N=8$, $\theta_s = \theta_{u_2} = 36^\circ$, $\theta_{u_1} = -36^\circ$, $\Delta_s = \Delta_u = 3^\circ$ e $\mu_s = \mu_u = 0,01 = 0,1\mu_{\max}$. A Figura 5.6 mostra a curva da SINR para os algoritmos CSG e ICSG e a Figura 5.7, os diagramas de radiação correspondentes para o regime permanente.

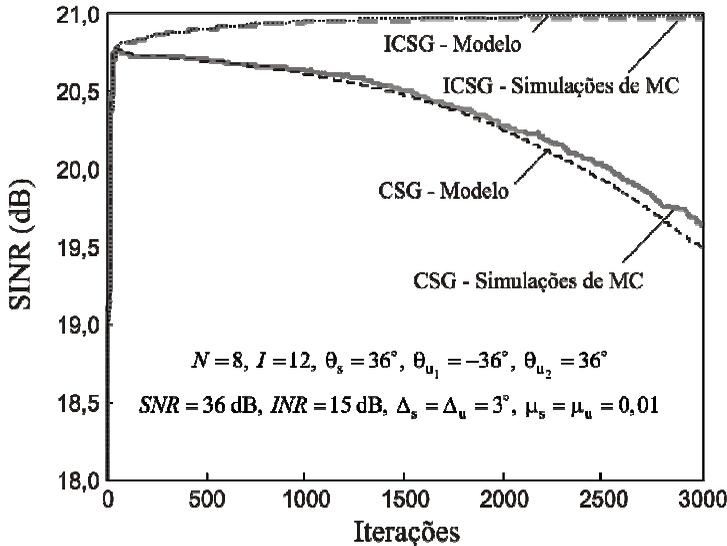
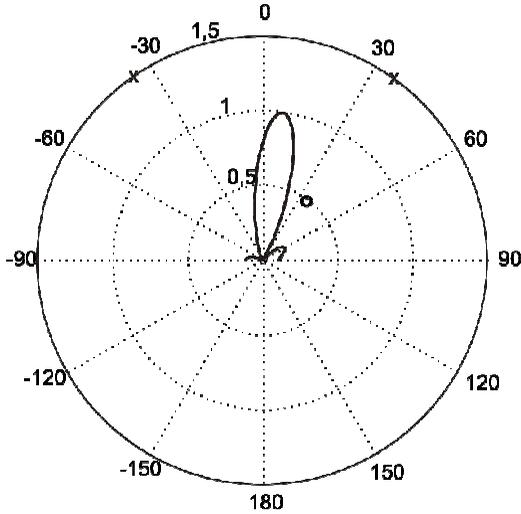


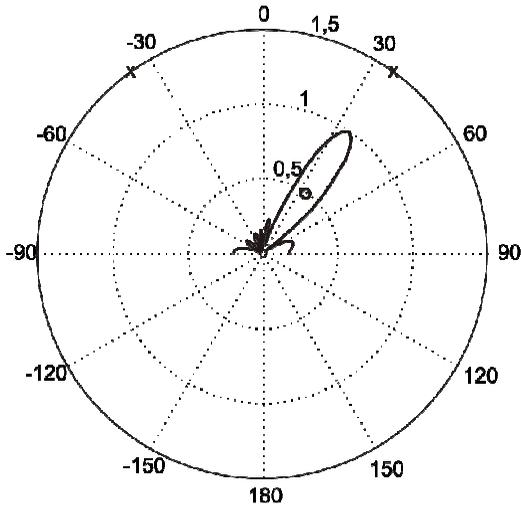
Figura 5.6. Exemplo 2. Curvas da SINR. Algoritmo CSG: (linhas cinza) simulações de MC; (linhas tracejadas escuras) modelo. Algoritmo ICSG: (linhas tracejadas cinza) simulações de MC; (linhas pontilhadas escuras) modelo.

$$M = 8, I = 12, \theta_s = 36^\circ, \theta_{u_1} = -36^\circ, \theta_{u_2} = 36^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,01$$



(a)

$$M = 8, I = 12, \theta_s = 36^\circ, \theta_{u_1} = -36^\circ, \theta_{u_2} = 36^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,01$$



(b)

Figura 5.7. Exemplo 2. Diagramas de radiação. Terminal móvel da própria célula (o). Interferentes (x). (a) Algoritmo CSG. (b) Algoritmo ICSG.

5.5.3. EXEMPLO 3

Neste exemplo, o caso $\mathbf{R}_{u_2} \cong \mathbf{R}_{u_1} \cong \mathbf{R}_s$ é considerado. Os parâmetros utilizados são $N=8$, $\theta_s=0^\circ$, $\theta_{u_1}=-12^\circ$, $\theta_{u_2}=12^\circ$, $\Delta_s=\Delta_u=3^\circ$ e $\mu_s=\mu_u=0,05=0,1\mu_{\max}$. A Figura 5.8 mostra as curvas da SINR para os algoritmos CSG e ICSG e a Figura 5.9, os correspondentes diagramas de radiação em regime permanente. As figuras apresentadas mostram um bom casamento entre simulação numérica e o modelo proposto, a pesar de pequenos desvios para o algoritmo CSG padrão, validando assim os aspectos teóricos considerados.

Os diagramas de radiação, apresentados nas Figuras 5.5(a), 5.7(a) e 5.9(a), mostram que durante o comportamento não-balanceado do algoritmo CSG padrão, o TM não recebe potência suficiente da ERB, indicando que o serviço foi interrompido. Por outro lado, as antenas controladas pelo algoritmo ICSG [veja Figuras 5.5(b), 5.7(b) e 5.9(b)] fornecem potência suficiente ao TM em todas as condições de operação consideradas.

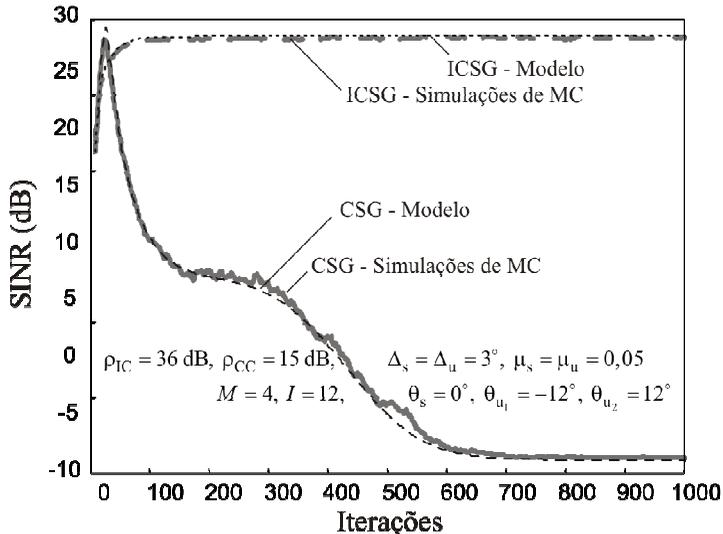
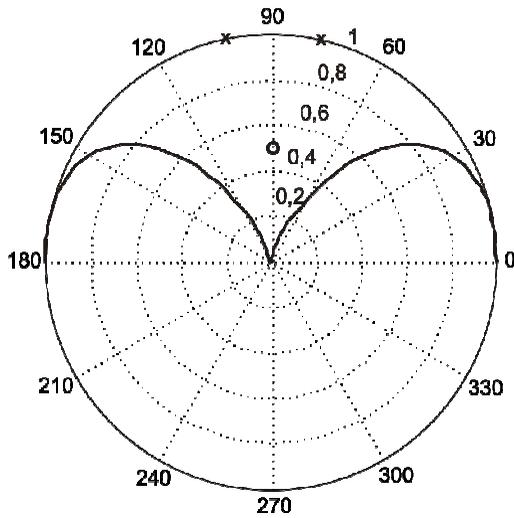


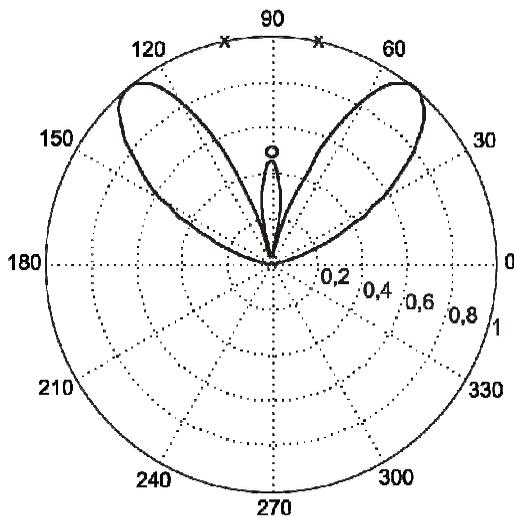
Figura 5.8. Exemplo 3. Curvas da SINR. Algoritmo CSG: (linhas cinza) simulações de MC; (linhas tracejadas escuras) modelo. Algoritmo ICSG: (linhas tracejadas cinza) simulações de MC; (linhas pontilhadas escuras) modelo.

$$M = 4, I = 12, \theta_s = 0^\circ, \theta_{u_1} = -12^\circ, \theta_{u_2} = 12^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,05$$



(a)

$$M = 4, I = 12, \theta_s = 0^\circ, \theta_{u_1} = -12^\circ, \theta_{u_2} = 12^\circ, \Delta_s = \Delta_u = 3^\circ, \mu_s = \mu_u = 0,05$$



(b)

Figura 5.9. Exemplo 3. Diagramas de radiação. Terminal móvel da própria célula (o). Interferentes (x). (a) Algoritmo CSG. (b) Algoritmo ICSG.

5.6. CONCLUSÕES

Neste Capítulo, foi discutido um comportamento indesejado do algoritmo CSG. Tal anomalia surge quando estão combinadas as seguintes situações: um ou mais ângulos de chegada dos sinais interferentes apresentam valores numéricos próximos ao ângulo de chegada do sinal útil. A essa condição tem-se associada também uma dispersão pequena do ângulo de chegada dos referidos sinais. Então, baseado no modelo que descreve o comportamento médio dos pesos, o comportamento não-balanceado do algoritmo CSG foi explicado e uma versão melhorada do algoritmo CSG (ICSG) foi apresentada.

6.1. CONCLUSÕES

Neste trabalho, foi desenvolvido e aplicado um conjunto de ferramentas de análise que auxiliam na modelagem estatística de três destacados algoritmos adaptativos. Em particular, foram considerados aqueles algoritmos baseados no gradiente, que usam o sinal de entrada normalizado. Eles são os algoritmos NLMS, LMS-DT e CSG.

No Capítulo 2, foi apresentado um modelo estocástico para o algoritmo NLMS, assumindo apenas uma hipótese simplificativa: independência entre os pesos do filtro adaptativo e as amostras do sinal de entrada. Decorrente dessa abordagem para obter o modelo, as expressões obtidas são matematicamente mais complexas do que outros modelos similares disponíveis na literatura. Entretanto, os modelos derivados são mais precisos para ambos os regimes, transiente e permanente. Essa precisão é devido ao fato que as matrizes envolvidas no modelo são calculadas sem utilizar o Princípio da Média (PM).

No Capítulo 3, foi discutido um modelo estatístico para o algoritmo LMS-DT. Para tornar a análise mais geral, o caso não-estacionário foi considerado; adicionalmente, o caso estacionário é obtido a partir do modelo desenvolvido como um caso particular. A análise realizada é independente da ordem do filtro e do comprimento da janela de observação utilizada para estimar a potência do sinal em cada sub-banda. Os modelos resultantes apresentaram um casamento muito bom com os resultados obtidos por simulações de Monte Carlo. Expressões para o erro em excesso, desajuste, passo de adaptação para o mínimo erro em excesso e a relação entre o grau de não-estacionaridade e o desajuste foram também derivadas.

No Capítulo 4, modelos analíticos para o momento de primeira ordem do vetor de pesos e para a curva de SINR do algoritmo CSG foram derivados, considerando a hipótese de adaptação lenta. Em geral, foi observado um bom casamento entre as curvas descritas por simulações e aquelas obtidas através do modelo, tanto em regime transiente quanto permanente. Utilizando as expressões determinísticas obtidas, foram avaliadas as particularidades do comportamento do

algoritmo CSG, quando utilizado em arranjos de antenas aplicados em sistemas celulares.

No Capítulo 5, foi apontado um comportamento indesejado do algoritmo CSG. Tal anomalia surge quando estão combinadas as seguintes situações: i) o sinal útil e um ou mais sinais interferentes vêm aproximadamente da mesma direção (ângulos de chegada próximos); ii) o ângulo de dispersão de cada um dos sinais é pequeno. Então, baseado no modelo que descreve o comportamento médio dos pesos, o comportamento não-balanceado do algoritmo CSG foi explicado, permitindo que uma versão melhorada do algoritmo CSG fosse proposta.

O presente trabalho fornece, também, novas ferramentas de análise para serem utilizadas na modelagem de outros algoritmos tipo-NLMS. Também servem para melhorar a precisão de modelos existentes obtidos através de outras estratégias, tais como os baseados em equações diferenciais estocásticas [17] e na conservação de energia [1], requerendo o cálculo de valores esperados semelhantes aos considerados nesta tese.

Especificamente, o fato de considerar valores esperados como $E\{\mathbf{x}(n)\mathbf{x}^T(n) / [\mathbf{x}^T(n)\mathbf{x}(n)]\}$ sem o uso do PM não apenas possibilita modelagens mais precisas (como as discutidas nos Capítulos 2 e 3), permitindo também evidenciar o comportamento não-balanceado do algoritmo CSG, não sendo possível usando o PM.

Os resultados obtidos deste trabalho de pesquisa deram origem a 12 artigos publicados em eventos nacionais, internacionais e revista. A seguir, são listados tais artigos:

- 1) J. E. Kolodziej, O. J. Tobias e R. Seara, "Stochastic analysis of the modified DNLMS algorithm for Gaussian data," *Proc. IEEE International Telecommunication Symposium (ITS 2006)*, Fortaleza, CE, Brasil, Sep. 2006, pp. 1–5.
- 2) J. E. Kolodziej, O. J. Tobias e R. Seara, "Cálculo aproximado de integrais hiperelípticas aplicado à modelagem estatística do algoritmo NLMS," *Anais do XIII Encuentro de Educación Matemática en Carreras de Ingeniería – IV Internacional (EMCI 13)*, Oberá, Argentina, Out. 2006, pp. 1-11.

- 3) J. E. Kolodziej, O. J. Tobias e R. Seara, "Modelagem de algoritmos adaptativos normalizados para sinais gaussianos e não gaussianos," *Anais do 5º Encuentro del Grupo Latinoamericano de Emisión Acústica (E-GLEA5)*, Oberá, Argentina, Ago. 2007, pp. 1-6.
- 4) J. E. Kolodziej, O. J. Tobias e R. Seara, "An improved stochastic model of the NLMS algorithm for correlated input data," *Proc. 15th European Signal Processing Conference (EUSIPCO 2007)*, Poznań, Polônia, Set. 2007, pp. 360-364.
- 5) J. E. Kolodziej, O. J. Tobias e R. Seara, "Modelagem estatística do algoritmo CSG aplicada para um arranjo adaptativo de antenas," *Anais do XXV Simpósio Brasileiro de Telecomunicações*, Recife, PE, Brasil, Set. 2007, pp. 1-6.
- 6) J. E. Kolodziej, O. J. Tobias, R. Seara e D. R. Morgan, "On the constrained stochastic gradient algorithm behavior in wireless communication networks," *Proc. 9th IEEE Workshop on Signal Processing Advances for Wireless Communications*, Recife, PE, Brasil, Jul. 2008, pp. 1-5.
- 7) J. E. Kolodziej, O. J. Tobias, R. Seara e D. R. Morgan, "An analytical model for the constrained stochastic gradient algorithm," *Proc. 16th European Signal Processing Conference*, Lausanne, Suíça, Ago. 2008, pp. 1-5.
- 8) J. E. Kolodziej, O. J. Tobias e R. Seara, "Considerações sobre o comportamento do algoritmo CSG aplicado em sistemas de comunicação móvel celular," *Anais do XXVI Simpósio Brasileiro de Telecomunicações*, Rio de Janeiro, RJ, Brasil, Set. 2008, pp. 1-6.
- 9) J. E. Kolodziej, O. J. Tobias e R. Seara, "Um modelo preciso para o algoritmo LMS normalizado," *Anais do XVII Congresso Brasileiro de Automática*, Juiz de Fora, MG, Brasil, Set. 2008, pp. 1-6.

- 10) J. E. Kolodziej, O. J. Tobias, R. Seara e D. R. Morgan, "On the stochastic gradient algorithm: model, performance and improved version," *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1304-1315, Apr. 2009.
- 11) J. E. Kolodziej, O. J. Tobias e R. Seara, "Stochastic analysis of the transform domain LMS algorithm for a non-stationary environment," *Proc. 17th European Signal Processing Conference (EUSIPCO 2009)*, Glasgow, Escócia, Ago. 2009, pp. 1-5.
- 12) J. E. Kolodziej, O. J. Tobias e R. Seara, "Análise estocástica do algoritmo LMS no domínio transformado para ambiente não-estacionário," *Anais do XXVII Simpósio Brasileiro de Telecomunicações*, Blumenau, SC, Brasil, Out. 2009, pp. 1-6.

6.2. TRABALHOS FUTUROS

Dentre as propostas para dar continuidade ao presente trabalho de tese seguem algumas sugestões:

- Aplicar as ferramentas aqui desenvolvidas a outros algoritmos adaptativos. Por exemplo, os algoritmos adaptativos em sub-banda que guardam muita proximidade com o LMS-DT. Tais algoritmos têm muita importância prática devido à tendência atual de utilizar algoritmos facilmente paralelizáveis.
- Derivar modelos que relacionem a velocidade de convergência com valores da SINR em regime permanente para os algoritmos CSG e ICSG, permitindo assim, a partir delas, obter expressões para o passo de adaptação ótimo.
- Modelar os algoritmos adaptativos aqui estudados em condições de adaptação rápida. Nessa condição de trabalho, a hipótese de independência entre o vetor de pesos e os dados não poderia ser mais considerada.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] A. H. Sayed, *Adaptive Filters*. NJ: John Wiley & Sons, 2008.
- [2] J. H. Husoy and M. S. E. Adabi, "Unified approach to adaptive filters and their performance," *IET Signal Process.*, vol. 2, no. 2, pp. 97-109, Jun. 2008.
- [3] A. N. Kolmogorov, "Sur l'interpolation et extrapolation des suites stationnaires," *C.R. Acad. Sci., Paris*, vol. 208, pp. 2043-2045, 1939.
- [4] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series, with Engineering Applications*. Cambridge, MA: MIT Press, 1949.
- [5] N. Wiener and E. Hopf, "On a class of singular integral equations," *Proc. Prussian Acad. Math-Phys. Ser.*, pp. 696, 1931.
- [6] N. Levinson, "The Wiener RMS (root-mean-square) error criterion in filter design and prediction," *J. Math Phys.*, vol. 25, pp. 261-278, 1947.
- [7] B. Widrow and J. M. E. Hoff, "Adaptive switching circuits," *Proc. IRE WESCON Conv. Rec.*, vol. 4, Aug. 1960, pp. 96-104.
- [8] S. Haykin, *Adaptive Filter Theory*, 4th ed. NJ: Prentice-Hall, 2002.
- [9] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications*. New York: John Wiley & Sons, 1998.
- [10] J. I. Nagumo and A. Noda, "A learning method for system identification," *IEEE Trans. Autom. Control*, vol. 12, no. 3, pp. 282-287, Jun. 1967.
- [11] A. E. Albert and J. L. S. Gardner, *Stochastic Approximation and Nonlinear Regression*. Cambridge, MA: MIT Press, 1967.

- [12] R. R. Bitmead and B. D. O. Anderson, "Performance of adaptive estimation algorithms in dependent random environments," *IEEE Trans. Automat. Contr.*, vol. 25, no. 4, pp. 788-794, Aug. 1980.
- [13] M. Rupp, "The behavior of LMS and NLMS algorithms in the presence of spherically invariant processes," *IEEE Trans. Signal Process.*, vol. 41, no. 3, pp. 1149-1160, Mar. 1993.
- [14] M. H. Costa and J. C. M. Bermudez, "An improved model for the normalized LMS algorithm with Gaussian inputs and large number of coefficients," *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Orlando, FL, May 2002, pp. 1385-1388.
- [15] E. M. Lobato, O. J. Tobias, and R. Seara, "Stochastic modeling of the transform-domain ϵ LMS algorithm for a time-varying environment," *Proc. 13th European Signal Processing Conference*, Antalya, Turkey, Sep. 2005, pp. 1-4.
- [16] E. M. Lobato, O. J. Tobias, and R. Seara, "Stochastic model for the NLMS algorithm with correlated Gaussian data," *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Toulouse, France, May 2006, pp. 760-763.
- [17] G. Barrault, M. H. Costa, J. C. M. Bermudez, and A. Lenzi, "A new analytical model for the NLMS algorithm," *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Philadelphia, Mar. 2005, pp. 41-44.
- [18] E. Eweda, "A new approach for analyzing the limiting behavior of the normalized LMS algorithm under weak assumptions," *Signal Process.*, vol. 89, no. 11, pp. 2143-2151, Nov. 2009.
- [19] C. G. Samson and U. Reddy, "Fixed point error analysis of the normalized ladder algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 31, no. 5, pp. 1177-1191, Oct. 1983.
- [20] S. S. Narayan, A. M. Peterson, and M. J. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 31, no. 3, pp. 609-615, Jun. 1983.

- [21] S. Hosur and A. H. Tewfik, "Wavelet transform domain adaptive FIR filtering," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 617-630, Mar. 1997.
- [22] S. J. Chern, J. C. Horng, and K. M. Wong, "The performance of the hybrid LMS adaptive algorithm," *Signal Process.*, vol. 44, no. 1, pp. 67-88, Jun. 1995.
- [23] D. I. Kim and P. D. Wilde, "Performance analysis of the DCT-LMS adaptive filtering algorithm," *Signal Process.*, vol. 80, no. 8, pp. 1629-1654, Aug. 2000.
- [24] E. M. Lobato, O. J. Tobias, and R. Seara, "Stochastic modeling of the transform domain ϵ LMS algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1840-1852, May 2008.
- [25] D. R. Morgan, "Downlink adaptive array algorithms for cellular mobile communications," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 476-488, Mar. 2003.
- [26] D. R. Morgan, "Adaptive algorithms for solving generalized eigenvalue signal enhancement problems," *Signal Process.*, vol. 84, no. 6, pp. 957-968, Mar. 2004.
- [27] G. Mathew and V. U. Reddy, "A quasi-Newton adaptive algorithm for generalized symmetric eigenvalue problem," *IEEE Trans. Signal Process.*, vol. 44, no. 10, pp. 2413-2422, Oct. 1996.
- [28] S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*, Academic Press., 2000.
- [29] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 2002.
- [30] H. Brehm and W. Stamler, "Description and generation of spherically invariant speech-model signals," *Signal Process.*, vol. 12, pp. 119-141, Mar. 1987.
- [31] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing*. New York: McGraw-Hill, 2000.

- [32] O. Macchi, *Adaptive Processing: The Least Mean Squares Approach with Applications in Transmission*. New York: John Wiley & Sons, 1995.
- [33] L. C. Godara, "Cellular Systems," *Handbook in Antennas in Wireless Communications*, L. C. Godara, ed., Boca Raton, FL: CRC Press, 2001.
- [34] J. M. Goldberg and J. R. Fonollosa, "Downlink beamforming for spatially distributed sources in cellular mobile communications," *Signal Process.*, vol. 65, no. 2, pp. 181-197, Mar. 1998.
- [35] Y.-C. Liang and F. P. S. Chin, "Downlink channel covariance matrix (DCCM) estimation and its applications in wireless DS-CDMA systems," *IEEE J. Select. Areas Commun.*, vol. 19, no. 2, pp. 222-232, Feb. 2001.
- [36] B. M. Hochwald and T. L. Marzetta, "Adapting a downlink array from uplink measurements," *IEEE Trans. Signal Process.*, vol. 49, no. 3, pp. 642-653, Mar. 2001.
- [37] C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1992.
- [38] J. Demmel, *Applied Numerical Linear Algebra*. Philadelphia: SIAM, 1997.
- [39] N. J. Bershad, P. Celka, and J. -M. Vesin, "Stochastic analysis of gradient adaptive identification of nonlinear systems with memory for Gaussian data and noisy input and output measurements," *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 675-689, Mar. 1999.
- [40] J. E. Kolodziej, O. J. Tobias, R. Seara, and D. R. Morgan, "On the stochastic gradient algorithm: model, performance and improved version," *Trans. Signal Process.*, vol. 57, no. 4, pp. 1304-1315, Apr. 2009.

Apêndice 1 - Cálculo da Matriz **P**

Apêndice 2 - Determinação da Matriz **S**

Apêndice 3 - Cálculo Exato das Matrizes de Covariância Normalizadas

Apêndice 4 - Verificação do Autovalor Dominante

APÊNDICE 1
CÁLCULO DA MATRIZ P

Neste apêndice, é proposto um método aproximado para o cálculo da matriz **P**, cujos elementos estão dados por

$$p(i, j) = E \left[\frac{x_{T,i}(n)x_{T,j}(n)}{\hat{\sigma}_i^2(n)} \right]. \quad (\text{A1.1})$$

Para tal, é definido um vetor, dado por

$$\mathbf{x}_{i,j}(n) = [x_{T,i}(n) \ x_{T,i}(n-1) \dots \ x_{T,i}(n-M+1) \ x_{T,j}(n)]^T, \text{ de forma que}$$

$$\hat{\sigma}_i^2(n) = \frac{1}{M} \sum_{k=0}^{M-1} x_{T,i}^2(n-k) = \frac{1}{M} \mathbf{x}_{i,j}^T(n) \mathbf{I}_s \mathbf{x}_{i,j}(n) \quad (\text{A1.2})$$

com $\mathbf{I}_s = \text{diag}[\underbrace{1 \dots 1}_M 0]$. Utilizando (A1.2) e considerando processos conjuntamente gaussianos, (A1.1) pode ser calculado através da seguinte expressão:

$$p(i, j) = \frac{M}{\sqrt{(2\pi)^{M+1} \det(\mathbf{R}_{i,j})}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M+1 \text{ integrais}} \frac{x_{T,i} x_{T,j}}{\mathbf{x}_{i,j}^T \mathbf{I}_s \mathbf{x}_{i,j}} e^{-\mathbf{x}_{i,j}^T \mathbf{R}_{i,j}^{-1} \mathbf{x}_{i,j} / 2} d\mathbf{x}_{i,j} \quad (\text{A1.3})$$

onde $\mathbf{R}_{i,j} = E[\mathbf{x}_{i,j} \mathbf{x}_{i,j}^T]$ é a matriz de autocorrelação de $\mathbf{x}_{i,j}$. Em (A1.3), o índice de tempo n foi desconsiderado por simplicidade. Para determinar (A1.3), utiliza-se um procedimento similar ao apresentado no Capítulo 2. Então, definindo uma função auxiliar, denotada por $f_{T,i,j}(\omega)$, com elementos dados por

$$f_{T i,j}(\omega) = \frac{M}{\sqrt{(2\pi)^{M+1} \det(\mathbf{R}_{i,j})}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M+1 \text{ integrais}} \frac{x_{T,i} x_{T,j}}{\mathbf{x}_{i,j}^T \mathbf{I}_s \mathbf{x}_{i,j}} e^{-\mathbf{x}_{i,j}^T \mathbf{L}_{i,j}^{-1}(\omega) \mathbf{x}_{i,j}/2} d\mathbf{x}_{i,j} \quad (\text{A1.4})$$

com

$$\mathbf{L}_{i,j}(\omega) = (2\omega \mathbf{I}_s + \mathbf{R}_{i,j}^{-1})^{-1} \quad (\text{A1.5})$$

o valor esperado (A1.1) pode ser alternativamente escrito como

$$p(i, j) = f_{T i,j}(0). \quad (\text{A1.6})$$

Essa função auxiliar permite resolver (A1.3) considerando uma única integral, como mostrado na seqüência. Assim, diferenciando (A1.4) em relação a ω , o termo $\mathbf{x}_{i,j}^T \mathbf{I}_s \mathbf{x}_{i,j}$ do denominador é eliminado, resultando em

$$\frac{\partial f_{T i,j}(\omega)}{\partial \omega} = \frac{-M \sqrt{\det[\mathbf{L}_{i,j}(\omega)]}}{\sqrt{\det(\mathbf{R}_{i,j})}} \alpha_{i,j}(\omega) \quad (\text{A1.7})$$

com

$$\alpha_{i,j}(\omega) = \frac{1}{\sqrt{(2\pi)^{M+1} \det[\mathbf{L}_{i,j}(\omega)]}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M+1 \text{ integrais}} x_{T,i} x_{T,j} e^{-\mathbf{x}_{i,j}^T \mathbf{L}_{i,j}^{-1}(\omega) \mathbf{x}_{i,j}/2} d\mathbf{x}_{i,j}. \quad (\text{A1.8})$$

A expressão (A1.8) pode ser vista como a correlação cruzada entre $x_{T,i}$ e $x_{T,j}$, sendo essas variáveis amostras de um processo conjuntamente gaussiano com matriz de covariância $\mathbf{L}_{i,j}(\omega)$. Como $x_{T,i}$ e $x_{T,j}$ são, respectivamente, o primeiro e o $(M+1)$ -ésimo

elemento do vetor $\mathbf{x}_{i,j}$, o fator $\alpha_{i,j}(\omega)$ pode ser substituído pelo $(1, M + 1)$ -ésimo elemento da matriz $\mathbf{L}_{i,j}(\omega)$. Assim,

$$\frac{\partial f_{T i,j}(\omega)}{\partial \omega} = \frac{-M[(2\omega \mathbf{R}_{i,j} \mathbf{I}_s + \mathbf{I}_{M+1})^{-1} \mathbf{R}_{i,j}]_{1,M+1}}{\sqrt{\det(2\omega \mathbf{R}_{i,j} \mathbf{I}_s + \mathbf{I}_{M+1})}} \quad (\text{A1.9})$$

onde \mathbf{I}_{M+1} denota uma matriz identidade de dimensões $(M + 1) \times (M + 1)$. Considerando a autodecomposição $\mathbf{R}_{i,j} \mathbf{I}_s = \mathbf{Q}_{s,i,j} \mathbf{\Lambda}_{s,i,j} \mathbf{Q}_{s,i,j}^{-1}$, onde $\mathbf{Q}_{s,i,j}$ e $\mathbf{\Lambda}_{s,i,j}$ são, respectivamente, a matriz de autovetores e a matriz de autovalores de $\mathbf{R}_{i,j} \mathbf{I}_s$, obtém-se

$$\frac{\partial f_{i,j}(\omega)}{\partial \omega} = \frac{-M[\mathbf{Q}_{s,i,j} (2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})^{-1} \mathbf{Q}_{s,i,j}^{-1} \mathbf{R}_{i,j}]_{1,M+1}}{\sqrt{\det(2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})}}. \quad (\text{A1.10})$$

A expressão (A1.10) pode ser reescrita de forma mais compacta como segue:

$$\frac{\partial f_{i,j}(\omega)}{\partial \omega} = \frac{-M \mathbf{q}_{s,i,j}^T (2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})^{-1} \mathbf{Q}_{s,i,j}^{-1} \mathbf{r}_{e,i,j}}{\sqrt{\det(2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})}} \quad (\text{A1.11})$$

onde $\mathbf{q}_{s,i,j}$ é um vetor contendo a primeira linha de $\mathbf{Q}_{s,i,j}$ e $\mathbf{r}_{e,i,j}$ é um vetor contendo a última coluna de $\mathbf{R}_{i,j}$. Simplificações adicionais podem ser obtidas analisando a estrutura de $\mathbf{Q}_{s,i,j}$. Da definição da autodecomposição temos que

$$\mathbf{Q}_{s,i,j} = \begin{bmatrix} \mathbf{Q}_{T,i} & \mathbf{0} \\ \mathbf{r}_{i,j}^T \mathbf{Q}_{T,i} \mathbf{\Lambda}_{T,i}^{-1} & 1 \end{bmatrix} \quad (\text{A1.12})$$

onde $\mathbf{Q}_{T,i}$ e $\mathbf{\Lambda}_{T,i}$ são matrizes contendo os autovetores e os autovalores, respectivamente, de $\mathbf{R}_{T,i}$, $\mathbf{r}_{i,j} = E[\mathbf{x}_{T,i}(n) x_{T,j}(n)]$ é um

vetor de correlação cruzada entre bandas e $\mathbf{0}$ é um vetor nulo de comprimento M . Considerando agora que $\mathbf{Q}_{s,i,j} \mathbf{Q}_{s,i,j}^{-1} = \mathbf{I}_{M+1}$, podemos obter

$$\mathbf{Q}_{s,i,j}^{-1} = \begin{bmatrix} \mathbf{Q}_{T,i}^T & \mathbf{0} \\ -\mathbf{r}_{i,j}^T \mathbf{Q}_{T,i} \mathbf{\Lambda}_{T,i}^{-1} \mathbf{Q}_{T,i}^T & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{T,i}^T & \mathbf{0} \\ -\mathbf{r}_{i,j}^T \mathbf{R}_{T,i}^{-1} & 1 \end{bmatrix}. \quad (\text{A1.13})$$

Além disso, podem ser estabelecidas as seguintes relações:

$$\mathbf{r}_{e,i,j} = \begin{bmatrix} \mathbf{r}_{i,j} \\ \sigma_j^2 \end{bmatrix} \quad (\text{A1.14})$$

e

$$\mathbf{q}_{s,i,j} = \begin{bmatrix} \mathbf{q}_{T,i} \\ 0 \end{bmatrix} \quad (\text{A1.15})$$

com $\mathbf{q}_{T,i} = [q_{i,1} \ q_{i,2} \ \dots \ q_{i,M}]^T$ contendo o primeiro elemento de cada autovetor de $\mathbf{R}_{T,i}$.

Agora, substituindo (A1.13), (A1.14) e (A1.15) em (A1.11), resulta

$$\frac{\partial f_{i,j}(\omega)}{\partial \omega} = -M \begin{bmatrix} \mathbf{q}_{T,i} \\ 0 \end{bmatrix}^T \frac{(2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})^{-1}}{\sqrt{\det(2\omega \mathbf{\Lambda}_{s,i,j} + \mathbf{I}_{M+1})}} \begin{bmatrix} \mathbf{Q}_{T,i}^T \mathbf{r}_{i,j} \\ -\mathbf{r}_{i,j}^T \mathbf{R}_{T,i}^{-1} \mathbf{r}_{i,j} + \sigma_j^2 \end{bmatrix}. \quad (\text{A1.16})$$

Considerando a estrutura de $\mathbf{R}_{e,i,j} \mathbf{I}_s$, os primeiros M elementos (autovalores) de $\mathbf{\Lambda}_{s,i,j}$ são coincidentes com os autovalores de $\mathbf{R}_{T,i}$ e o último é igual a zero, levando a

$$\frac{\partial f_{i,j}(\omega)}{\partial \omega} = \frac{-M \mathbf{q}_{T,i}^T (2\omega \mathbf{\Lambda}_{T,i} + \mathbf{I}_M)^{-1} \mathbf{Q}_{T,i}^T \mathbf{r}_{i,j}}{\sqrt{\det(2\omega \mathbf{\Lambda}_{T,i} + \mathbf{I}_M)}}. \quad (\text{A1.17})$$

Então, considerando

$$\int_0^{\infty} \frac{\partial}{\partial \omega} f_{Ti,j}(\omega) d\omega = f_{Ti,j}(\infty) - f_{Ti,j}(0) \quad (A1.18)$$

e que $f_{Ti,j}(\infty) = 0$, obtêm-se

$$p(i, j) = M \mathbf{q}_{T,i}^T \mathbf{H}_i \mathbf{Q}_{T,i}^T \mathbf{r}_{i,j} \quad (A1.19)$$

onde os elementos da matriz diagonal \mathbf{H}_i são dados por

$$h_i(l, l) = \int_0^{\infty} \frac{1}{(1 + 2\omega \lambda_{i,l}) \sqrt{\prod_{k=1}^M (1 + 2\omega \lambda_{i,k})}} d\omega. \quad (A1.20)$$

Na expressão anterior, $\lambda_{i,k}$, para $k = 1, \dots, M$, são os autovalores da matriz $\mathbf{R}_{T,i}$. As integrais de (A1.20) podem ser resolvidas seguindo o procedimento apresentado no Capítulo 2. Assim, obtêm-se

$$h_i(l, l) \cong \frac{1}{2\lambda_{i,l} \sqrt{a_i}} \left[\sum_{q=1}^{M/2} A_{i,l,q} \ln(\lambda'_{i,q}) + B_{i,l} \ln(\lambda_{i,l}) \right] \quad (A1.21)$$

onde

$$a_i = \prod_{k=1}^M \lambda_{i,k} \quad (A1.22)$$

$$A_{i,l,q} = \frac{\lambda'_{i,q} \lambda_{i,l}}{\lambda'_{i,q} - \lambda_{i,l}} \prod_{\substack{j=1 \\ j \neq q}}^{M/2} \frac{\lambda'_{i,q} \lambda'_{i,j}}{\lambda'_{i,q} - \lambda'_{i,j}} \quad (A1.23)$$

e

$$B_{i,l} = \prod_{q=1}^{M/2} \frac{\lambda_{i,l} \lambda'_{i,q}}{\lambda_{i,l} - \lambda'_{i,q}}. \quad (A1.24)$$

APÊNDICE 2
DETERMINAÇÃO DA MATRIZ S

Neste apêndice, um novo procedimento para determinar os elementos da matriz S é discutido. De acordo com a abordagem proposta, os elementos da diagonal principal de S são calculados sem utilizar aproximações, as quais conduzem, muitas vezes, a imprecisões no processo de modelagem. Nessa abordagem, os elementos fora da diagonal principal são calculados utilizando o Princípio da Média (PM). Assim, os resultados obtidos com o método proposto são mais precisos quando comparado com aqueles obtidos considerando outros métodos disponíveis na literatura.

Para calcular os elementos da diagonal de S , um procedimento similar ao apresentado no Apêndice 1 é considerado. Assim, utilizando a definição formal do valor esperado de processos conjuntamente gaussianos, tem-se

$$s(i, i) = \frac{M^2}{\sqrt{(2\pi)^M \det(\mathbf{R}_{T,i})}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{x_{T,i}^2}{(\mathbf{x}_{T,i}^T \mathbf{x}_{T,i})^2} e^{-\mathbf{x}_{T,i}^T \mathbf{R}_{T,i}^{-1} \mathbf{x}_{T,i}/2} d\mathbf{x}_{T,i}. \tag{A2.1}$$

Aqui também é definida uma função auxiliar, denotada por $g_i(\omega)$, escrita como

$$g_i(\omega) = \frac{M^2}{\sqrt{(2\pi)^M \det(\mathbf{R}_{T,i})}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{x_{T,i}^2}{(\mathbf{x}_{T,i}^T \mathbf{x}_{T,i})^2} e^{-\mathbf{x}_{T,i}^T \mathbf{M}_i^{-1}(\omega) \mathbf{x}_{T,i}/2} d\mathbf{x}_{T,i} \tag{A2.2}$$

com

$$\mathbf{M}_i(\omega) = (2\omega \mathbf{I}_M + \mathbf{R}_{T,i}^{-1})^{-1}.$$

Note que

$$s(i, i) = g_i(0). \tag{A2.3}$$

Agora, derivando parcialmente duas vezes (A2.2) com respeito a ω , o termo $(\mathbf{x}_{T,i}^T \mathbf{x}_{T,i})^2$ do denominador de $g_i(\omega)$ é eliminado, obtendo-se

$$\frac{\partial^2 g_i(\omega)}{\partial \omega^2} = \frac{M^2 \sqrt{\det[\mathbf{M}_i(\omega)]}}{\sqrt{\det(\mathbf{R}_{T,i})}} \beta_i(\omega) \quad (\text{A2.4})$$

com

$$\beta_i(\omega) = \frac{1}{\sqrt{(2\pi)^M \det[\mathbf{M}_i(\omega)]}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} x_{T,i}^2 e^{-\mathbf{x}_{T,i}^T \mathbf{M}_i^{-1}(\omega) \mathbf{x}_{T,i}/2} d\mathbf{x}_{T,i}. \quad (\text{A2.5})$$

O fator $\beta_i(\omega)$ representa a variância do sinal $x_{T,i}(n)$, portanto, é igual ao (1,1)-ésimo elemento da matriz $\mathbf{M}_i(\omega)$. Dessa forma,

$$\frac{\partial^2 g_i(\omega)}{\partial \omega^2} = \frac{M^2 [(2\omega \mathbf{R}_{T,i} + \mathbf{I}_M)^{-1} \mathbf{R}_{T,i}]_{1,1}}{\sqrt{\det(2\omega \mathbf{R}_{T,i} + \mathbf{I}_M)}}. \quad (\text{A2.6})$$

Considerando a autodecomposição de $\mathbf{R}_{T,i}$, obtém-se

$$\frac{\partial^2 g_i(\omega)}{\partial \omega^2} = \frac{M^2 \mathbf{q}_{T,i}^T (2\omega \mathbf{\Lambda}_{T,i} + \mathbf{I}_M)^{-1} \mathbf{q}_{T,i}}{\sqrt{\det(2\omega \mathbf{\Lambda}_{T,i} + \mathbf{I}_M)}}. \quad (\text{A2.7})$$

Assim, os elementos da diagonal de \mathbf{S} podem ser calculados utilizando

$$s(i,i) = M^2 \mathbf{q}_{T,i}^T \mathbf{U}_i \mathbf{q}_{T,i}^T \quad (\text{A2.8})$$

onde \mathbf{U}_i é uma matriz diagonal auxiliar com elementos dados por

$$u_i(l,l) = \int_0^{\infty} \int_{\omega_1=\omega_2}^{\infty} \frac{\lambda_{i,l}}{(1+2\omega_1 \lambda_{i,l}) \sqrt{\prod_{k=1}^M (1+2\omega_1 \lambda_{i,k})}} d\omega_1 d\omega_2. \quad (\text{A2.9})$$

Para determinar a expressão anterior, pares de autovalores $\lambda_{i,k}$ de $\mathbf{R}_{T,i}$ são substituídos por sua média geométrica com multiplicidade dois. Assim, considerando que as raízes são todas reais e diferentes, a expansão em frações parciais de (A2.9) permite obter uma solução fechada para a integral. Assim,

$$u_i(l, l) \approx \frac{-1}{4\sqrt{a_i}} \left\{ \sum_{q=1}^{M/2} \frac{A_{i,l,q}}{\lambda'_{i,q}} [1 + \ln(2\lambda'_{i,q})] + \frac{B_{i,l}}{\lambda_{i,l}} [1 + \ln(2\lambda_{i,l})] \right\}. \quad (\text{A2.10})$$

Os elementos fora da diagonal principal de \mathbf{S} são calculados utilizando primeiro o PM através de

$$s(i, j) = E \left[\frac{x_{T,i} x_{T,j}}{\hat{\sigma}_i^2 \hat{\sigma}_j^2} \right] \approx E \left[\frac{1}{\hat{\sigma}_i^2 \hat{\sigma}_j^2} \right] E[x_{T,i} x_{T,j}] \quad (\text{A2.11})$$

e em seguida, considerando que as variâncias estimadas são não-correlacionadas umas com as outras, resultando em

$$s(i, j) = E \left[\frac{x_{T,i} x_{T,j}}{\hat{\sigma}_i^2 \hat{\sigma}_j^2} \right] \approx E \left[\frac{1}{\hat{\sigma}_i^2} \right] E[x_{T,i} x_{T,j}] E \left[\frac{1}{\hat{\sigma}_j^2} \right]. \quad (\text{A2.12})$$

Finalmente, seguindo o procedimento apresentado em [24], obtém-se

$$E \left[\frac{1}{\hat{\sigma}_i^2} \right] = \frac{M}{2\sqrt{a_i}} \sum_{q=1}^{M/2} C_{i,q} \ln(\lambda'_{i,q}), \quad i = 1, 2, \dots, N \quad (\text{A2.13})$$

com

$$C_{i,q} = \prod_{\substack{j=1 \\ j \neq q}}^{M/2} \frac{\lambda'_{i,q} \lambda'_{i,j}}{\lambda'_{i,j} - \lambda'_{i,q}}. \quad (\text{A2.14})$$

APÊNDICE 3
CÁLCULO EXATO DAS MATRIZES DE COVARIÂNCIA NORMALIZADAS

Neste apêndice, uma metodologia de cálculo para as matrizes de covariância normalizadas \mathbf{R}_x^N , definida como

$$\mathbf{R}_x^N = E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] \quad (\text{A3.1})$$

é apresentada. Esse procedimento é similar ao apresentado no Capítulo 2, com a diferença que aqui é considerando um vetor de sinais gaussianos complexos $\mathbf{x}(n)$, com matriz de covariância \mathbf{R} .

Para determinar o valor esperado em (A3.1), é definida uma função matricial auxiliar $\mathbf{F}_C(\omega)$, cujos elementos são dados pela seguinte expressão:

$$f_{C_{i,j}}(\omega) = \frac{1}{\pi^N \det(\mathbf{R})} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \frac{x_i x_j}{\mathbf{x}^H \mathbf{x}} e^{-\mathbf{x}^H \mathbf{L}_C^{-1} \mathbf{x}} d\mathbf{x}}_{N \text{ integrais}} \quad (\text{A3.2})$$

com $\mathbf{L}_C^{-1} = \mathbf{R}^{-1} + \omega \mathbf{I}$. Note que

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] = \mathbf{F}_C(0). \quad (\text{A3.3})$$

Diferenciando (A3.2) com respeito a ω e utilizando as propriedades da função densidade de probabilidade gaussiana complexa, obtém-se

$$f_{C_{i,j}}(0) = \int_0^{\infty} \frac{\{\mathbf{R}[\mathbf{I} + \omega \mathbf{R}]^{-1}\}_{i,j}}{\det(\mathbf{I} + \omega \mathbf{R})} d\omega. \quad (\text{A3.4})$$

Agora, utilizando a decomposição $\mathbf{R} = \mathbf{Q}_C \mathbf{\Lambda}_C \mathbf{Q}_C^T$, onde \mathbf{Q}_C é a matriz de autovetores de \mathbf{R} e $\mathbf{\Lambda}_C$ é uma matriz diagonal contendo os autovalores λ_i^C , para $i=1,2,\dots,N$, de \mathbf{R} , tem-se

$$E \left[\frac{\mathbf{x}(n) \mathbf{x}^H(n)}{\mathbf{x}^H(n) \mathbf{x}(n)} \right] = \mathbf{Q}_C \mathbf{H}_C \mathbf{Q}_C^T. \quad (\text{A3.5})$$

Assim, o valor esperado (A3.1) é determinado a partir dos N elementos da matriz diagonal \mathbf{H}_C , dados por

$$h_C(k, k) = \int_0^\infty \frac{\lambda_k^C}{(1 + \omega \lambda_k^C) \det(\mathbf{I} + \omega \mathbf{\Lambda}_C)} d\omega. \quad (\text{A3.6})$$

A integral (A3.6) pode ser então calculada utilizando expansão em frações parciais. Como exemplo, considere-se o caso em que todos os autovalores são diferentes, situação comumente encontrada na prática, resultando na seguinte expressão para os elementos de \mathbf{H}_C :

$$h_C(k, k) = A_{1,k}^C \lambda_k^C + A_{2,k}^C \ln(\lambda_k^C) + \sum_{\substack{i=1 \\ i \neq k}}^N B_{i,k}^C \ln(\lambda_i^C) \quad (\text{A3.7})$$

com

$$A_{1,k}^C = \frac{(\lambda_k^C)^{N-2}}{\prod_{\substack{i=1 \\ i \neq k}}^N (\lambda_k^C - \lambda_i^C)}, \quad (\text{A3.8})$$

$$B_{i,k}^C = \frac{\lambda_k^C (\lambda_i^C)^{N-1}}{(\lambda_i^C - \lambda_k^C) \prod_{\substack{j=1 \\ j \neq i}}^N (\lambda_i^C - \lambda_j^C)} \quad (\text{A3.9})$$

e

$$A_{2,k}^C = -A_{1,k}^C \lambda_k^C \sum_{\substack{i=1 \\ i \neq k}}^N \frac{\lambda_i^C}{\lambda_k^C - \lambda_i^C}. \quad (\text{A3.10})$$

APÊNDICE 4
VERIFICAÇÃO DO AUTOVALOR DOMINANTE

Neste apêndice, verifica-se a condição de autovalor dominante da matriz de covariância amostral normalizada quando ocorre o comportamento não-balanceado descrito no Capítulo 5. Para tal, considera-se que as matrizes de covariância têm grande dispersão de autovalores quando $\Delta \rightarrow 0$. Nesse caso, as seguintes relações são válidas:

$$\lim_{\Delta \rightarrow 0} \frac{\lambda_i}{\lambda_j} = 0, \quad j < i \quad (\text{A4.1})$$

e

$$\lim_{\Delta \rightarrow 0} \frac{\lambda_i}{\lambda_j} = \infty, \quad i < j. \quad (\text{A4.2})$$

Os autovalores da matriz de covariância amostral normalizada são dados por (A3.7). Para considerar o autovalor dominante, devemos fazer $k = 1$ em (A3.7), obtendo

$$h_C(1,1) = A_{1,1}^C \lambda_1^C + A_{2,1}^C \ln(\lambda_1^C) + \sum_{i=2}^N B_{i,1}^C \ln(\lambda_i^C). \quad (\text{A4.3})$$

Analisa-se a seguir cada um dos termos de (A4.3) utilizando (A4.1) e (A4.2):

i) Considerando o coeficiente $A_{1,1}^C$, temos

$$\lim_{\Delta \rightarrow 0} A_{1,1}^C = \lim_{\Delta \rightarrow 0} \frac{(\lambda_1^C)^{N-2}}{\prod_{l \neq 1} (\lambda_1^C - \lambda_l^C)} = \lim_{\Delta \rightarrow 0} \frac{(\lambda_1^C)^{-1}}{\prod_{l \neq 1} \left(1 - \frac{\lambda_l^C}{\lambda_1^C}\right)} = (\lambda_1^C)^{-1} \quad (\text{A4.4})$$

o qual resulta no $\lim_{\Delta \rightarrow 0} A_{1,1}^C \lambda_1^C = 1$.

ii) Agora, para o coeficiente $A_{2,1}^{\mathbb{C}}$, temos

$$\lim_{\Delta \rightarrow 0} A_{2,1}^{\mathbb{C}} = - \lim_{\Delta \rightarrow 0} \lambda_1^{\mathbb{C}} A_{1,1}^{\mathbb{C}} \sum_{j \neq 1} \frac{\lambda_j^{\mathbb{C}}}{\lambda_j^{\mathbb{C}} - \lambda_1^{\mathbb{C}}} = \lim_{\Delta \rightarrow 0} \sum_{j \neq 1} \frac{1}{1 - \frac{\lambda_1^{\mathbb{C}}}{\lambda_j^{\mathbb{C}}}} = 0. \quad (\text{A4.5})$$

Em conseqüência, $\lim_{\Delta \rightarrow 0} A_{2,1}^{\mathbb{C}} \ln(\lambda_1^{\mathbb{C}}) = 0$.

iii) Finalmente, para o coeficiente $B_{i,1}^{\mathbb{C}}$, obtém-se

$$\lim_{\Delta \rightarrow 0} B_{i,1}^{\mathbb{C}} = \lim_{\Delta \rightarrow 0} \frac{\lambda_1^{\mathbb{C}} (\lambda_i^{\mathbb{C}})^{N-1}}{(\lambda_i^{\mathbb{C}} - \lambda_1^{\mathbb{C}}) \prod_{l \neq i} (\lambda_i^{\mathbb{C}} - \lambda_l^{\mathbb{C}})} = \lim_{\Delta \rightarrow 0} \frac{1}{\left(\frac{\lambda_i^{\mathbb{C}}}{\lambda_1^{\mathbb{C}}} - 1 \right) \prod_{l \neq i} \left(1 - \frac{\lambda_l^{\mathbb{C}}}{\lambda_i^{\mathbb{C}}} \right)} = 0, \quad i \neq 1 \quad (\text{A4.6})$$

resultando em $\lim_{\Delta \rightarrow 0} \sum_{i=2}^N B_{i,1}^{\mathbb{C}} \ln(\lambda_i^{\mathbb{C}}) = 0$. Agora, usando (i), (ii) e (iii) em

(A4.3), tem-se $\lim_{\Delta \rightarrow 0} h_{\mathbb{C}}(1,1) = 1$.

Então, considerando os outros autovalores, i.e., $h_{\mathbb{C}}(k,k)$ para $k \neq 1$, e utilizando novamente (A4.1) e (A4.2), obtém-se

$$\lim_{\Delta \rightarrow 0} A_{1,k}^{\mathbb{C}} \lambda_k^{\mathbb{C}} = \lim_{\Delta \rightarrow 0} \frac{(\lambda_k^{\mathbb{C}})^{N-1}}{\prod_{l \neq k} (\lambda_k^{\mathbb{C}} - \lambda_l^{\mathbb{C}})} = \lim_{\Delta \rightarrow 0} \frac{1}{\prod_{l \neq k} \left(1 - \frac{\lambda_l^{\mathbb{C}}}{\lambda_k^{\mathbb{C}}} \right)} = 0, \quad (\text{A4.7})$$

$$\lim_{\Delta \rightarrow 0} A_{2,k}^{\mathbb{C}} = - \lim_{\Delta \rightarrow 0} \lambda_k^{\mathbb{C}} A_{1,k}^{\mathbb{C}} \sum_{j \neq k} \frac{\lambda_j^{\mathbb{C}}}{\lambda_j^{\mathbb{C}} - \lambda_k^{\mathbb{C}}} = - \lim_{\Delta \rightarrow 0} \lambda_k^{\mathbb{C}} A_{1,k}^{\mathbb{C}} \sum_{j \neq k} \frac{1}{1 - \frac{\lambda_k^{\mathbb{C}}}{\lambda_j^{\mathbb{C}}}} = 0 \quad (\text{A4.8})$$

e

$$\lim_{\Delta \rightarrow 0} B_{j,k}^{\mathbb{C}} = \lim_{\Delta \rightarrow 0} \frac{\lambda_k^{\mathbb{C}} (\lambda_j^{\mathbb{C}})^{N-1}}{(\lambda_j^{\mathbb{C}} - \lambda_k^{\mathbb{C}}) \prod_{l \neq j} (\lambda_j^{\mathbb{C}} - \lambda_l^{\mathbb{C}})} = \lim_{\Delta \rightarrow 0} \frac{1}{\left(\frac{\lambda_j^{\mathbb{C}}}{\lambda_k^{\mathbb{C}}} - 1 \right) \prod_{l \neq j} \left(1 - \frac{\lambda_l^{\mathbb{C}}}{\lambda_j^{\mathbb{C}}} \right)} = 0, \quad j \neq k$$

(A4.9)

o que leva a $\lim_{\Delta \rightarrow 0} h_{\mathbb{C}}(k, k) = 0$ para $k \neq 1$.