

Eduardo Vinicius Kuhn

**MODELAGEM ESTOCÁSTICA DO ALGORITMO NLMS:
REVISÃO E APRIMORAMENTOS**

Florianópolis
2012

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

Eduardo Vinicius Kuhn

**MODELAGEM ESTOCÁSTICA DO ALGORITMO NLMS:
REVISÃO E APRIMORAMENTOS**

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina para a obtenção do Grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Rui Seara

Florianópolis
2012

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática da Biblioteca Universitária da UFSC

Kuhn, Eduardo Vinicius
Modelagem estocástica do algoritmo NLMS [dissertação] :
Revisão e aprimoramentos / Eduardo Vinicius Kuhn ;
orientador, Rui Seara - Florianópolis, SC, 2012.
129 p. ; 21cm

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

Inclui referências

1. Engenharia Elétrica. 2. Filtragem adaptativa.
3. Modelagem estocástica. 4. Algoritmo NLMS. I. Seara, Rui.
II. Universidade Federal de Santa Catarina. Programa de
Pós-Graduação em Engenharia Elétrica. III. Título.

MODELAGEM ESTOCÁSTICA DO ALGORITMO NLMS: REVISÃO E APRIMORAMENTOS

Eduardo Vinicius Kuhn

Esta Dissertação foi julgada adequada para obtenção do Título de Mestre 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. Patrick Kuo-Peng, Dr.

Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Prof. Rui Seara, Dr.

Presidente

Prof. Eduardo Luiz Ortiz Batista, Dr.

Prof. Hans Helmut Zürn, Ph.D.

Dedico este trabalho aos meus pais
Sérgio e Tânia, e ao meu irmão
Fernando.

AGRADECIMENTOS

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

À minha família pelo incentivo, apoio, dedicação e compreensão ao longo dos anos. Em especial, ao meu pai, minha mãe, meu irmão, aos meus avós Odilo e Ana, a minha madrinha Luciane e meu padrinho Adelar, minha tia Dirce, e meus tios Arlindo e Delacir.

À Jaqueline, companheira de todas as horas, pelo seu apoio e dedicação mesmo nos momentos mais difíceis.

Aos meus amigos e colegas, Marcos Matsuo, Eduardo Beck e Fabio Perez pelas valiosas conversas e pelo apoio no decorrer dos últimos dois anos.

Ao Prof. Rui Seara pela amizade, incentivo, motivação, apoio e dedicação, fundamentais para o desenvolvimento deste trabalho de pesquisa e para minha formação acadêmica e profissional.

Ao Prof. Eduardo Luiz Ortiz Batista, vulgo Dudu, pelas suas valiosas contribuições durante todo o desenvolvimento do presente trabalho, pelas inúmeras conversas sobre filtragem adaptativa e outros tópicos.

Ao Prof. Hans Helmut Zürn, pelos conhecimentos transmitidos durante o primeiro ano do curso e também suas contribuições para o aprimoramento deste trabalho.

Aos meus amigos Elton e Walter, pelo grande apoio à realização deste trabalho, seja manifestando suas opiniões, realizando favores, oferecendo conselhos e conversando sobre os mais diversos assuntos.

Não menos importante, a todos os amigos do LINSE pela ótima convivência diária. Para registrar, um muito obrigado ao Augusto, Bernardo, Calisto, Ciro, Douglas, Feijão, Francisco, Guilherme, Javier, João, José, Mariane, Neco, Nonato, Sandra, Sidnei e Simone.

Ao CNPq, pelo apoio financeiro, à UFSC e ao LINSE por toda infraestrutura disponibilizada para a realização deste trabalho.

Estendo ainda os agradecimentos a todos que de alguma forma contribuíram para o desenvolvimento e conclusão desta dissertação.

RESUMO

Este trabalho de pesquisa trata da modelagem estocástica de algoritmos adaptativos com sinal de entrada normalizado. Particularmente, visa se obter um modelo estocástico mais preciso para o algoritmo NLMS (*normalized least-mean-square*) do que os até então disponíveis na literatura. O modelo aqui proposto considera um problema de identificação de sistema com planta estacionária, apresenta solução analítica e contempla diferentes tipos de sinais de entrada gaussianos (complexos, reais, correlacionados e não correlacionados). A partir das expressões que descrevem o comportamento médio do algoritmo, são derivadas expressões para o valor do erro quadrático médio em excesso em regime permanente e o desajuste. Tais expressões possibilitam uma melhor compreensão de como e quais parâmetros afetam o desempenho do algoritmo em regime permanente. Além disso, considerando sinal de entrada branco, relações que permitem ajustar (através do passo de adaptação) a velocidade de convergência e o erro em regime permanente dos algoritmos NLMS e LMS são apresentadas. Resultados de simulação para diferentes cenários de operação são mostrados, atestando a precisão do modelo proposto frente a outros modelos da literatura.

Palavras-chave: Algoritmo NLMS, filtragem adaptativa, modelagem estocástica.

ABSTRACT

This research work focuses on the stochastic modeling of adaptive algorithms with normalized input data. Specifically, the aim is to obtain a more accurate stochastic model for the normalized least mean square (NLMS) algorithm than those available so far in the literature. The stochastic model proposed here considers a system identification problem with stationary plant, presents analytical solution, and covers different types of Gaussian input data (complex, real, correlated, and uncorrelated). From expressions that describe the algorithm behavior, expressions for the steady state value of the excess mean-square error and the misadjustment are derived. Such expressions allow a better understanding of how and which parameters affect the steady state performance of the algorithm. In addition, for white input signal, relations that permit to adjust (via step size) the rate of convergence and the steady state error of the NLMS and LMS algorithms are presented. Simulation results for different operation scenarios are shown, attesting the accuracy of the proposed model in comparison with other models from the literature.

Keywords: NLMS algorithm, adaptive filtering, stochastic modeling.

LISTA DE FIGURAS

Figura 1.1: Diagrama de blocos de um problema de predição (adaptado de [11]).	24
Figura 1.2: Diagrama de blocos de um filtro adaptativo (adaptado de [3]).	26
Figura 1.3: Diagrama de blocos de um filtro adaptativo considerando uma estrutura transversal de filtragem (adaptado de [1] e [11]).	28
Figura 1.4: Diagrama em blocos de um problema de identificação de sistema (adaptado de [1], [2] e [11]).	37
Figura 2.1: Dispersão de autovalores de \mathbf{R} (linha sólida cinza) e \mathbf{R}^N (linha sólida escura).	63
Figura 2.2: Superfície de desempenho de um filtro adaptativo de segunda ordem. (a) Algoritmo LMS. (b) Algoritmo NLMS.	64
Figura 3.1: Exemplo 1. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.	69
Figura 3.2: Exemplo 1. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.	70
Figura 3.3: Exemplo 2. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando uma mesma velocidade de convergência.	71
Figura 3.4: Exemplo 2. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando um mesmo valor em regime permanente.	72
Figura 3.5: Exemplo 3. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.	73
Figura 3.6: Exemplo 3. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.	74
Figura 3.7: Exemplo 4. Comportamento médio de quatro coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.	76

Figura 3.8: Exemplo 4. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.	77
Figura 3.9: Exemplo 5. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [61]. (Linhas escuras traços-pontos) modelo de [66]. (Linhas escuras tracejadas) modelo proposto.	78
Figura 3.10: Exemplo 5. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [61]. (Linha escura traço-ponto) modelo de [66]. (Linha escura tracejada) modelo proposto.	79
Figura 3.11: Exemplo 6. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando uma mesma velocidade de convergência.	81
Figura 3.12: Exemplo 6. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando um mesmo valor em regime permanente.	81
Figura 3.13: Exemplo 7. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [57]. (Linhas escuras traços-pontos) modelo de [60]. (Linhas escuras tracejadas) modelo proposto (praticamente sobreposto com [60]).	82
Figura 3.14: Exemplo 7. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [57]. (Linha escura traço-ponto) modelo de [60]. (Linha escura tracejada) modelo proposto (praticamente sobreposto com [60]).	83
Figura 3.15: Exemplo 8. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [57]. (Linhas escuras traços-pontos) modelo de [61]. (Linhas escuras tracejadas) modelo proposto.	85
Figura 3.16: Exemplo 8. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [57]. (Linha escura traço-ponto) modelo de [61]. (Linha escura tracejada) modelo proposto.	86

Figura 3.17: Exemplo 9. EQM em excesso em regime permanente considerando diferentes valores de μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.....	88
Figura 3.18: Exemplo 9. Desajuste em função do parâmetro μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.....	88
Figura 3.19: Exemplo 10. EQM em excesso em regime permanente considerando diferentes valores de μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.....	89
Figura 3.20: Exemplo 10. Desajuste em função do passo de adaptação. (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.....	89

SUMÁRIO

CAPÍTULO 1 - INTRODUÇÃO	21
1.1 Teoria de estimação linear	23
1.2 Fundamentos de filtragem adaptativa	25
1.2.1 Estruturas de filtragem	26
1.2.2 Algoritmos adaptativos.....	29
1.2.3 Métricas de desempenho	34
1.2.4 Problema de identificação de sistema.....	36
1.3 Objetivos do trabalho	37
1.4 Estrutura do trabalho	38
CAPÍTULO 2 - MODELAGEM DO ALGORITMO NLMS	39
2.1 Introdução	39
2.2 Modelagem estocástica de algoritmos adaptativos	39
2.3 Modelos apresentados na literatura	40
2.4 Modelo proposto	44
2.4.1 Comportamento médio do vetor de coeficientes	45
2.4.2 Curva de aprendizagem	48
2.4.3 Matriz de covariância do vetor de erro nos coeficientes ..	50
2.5 Análise do modelo proposto	54
2.5.1 EQM em excesso em regime permanente	54
2.5.2 Desajuste	58
2.5.3 Relacionando os algoritmos NLMS e LMS	58
2.5.4 Efeito da normalização no desempenho do algoritmo.....	60
2.6 Conclusões	65
CAPÍTULO 3 - RESULTADOS DE SIMULAÇÃO	67
3.1 Introdução	67
3.2 Ambiente de operação	67
3.3 Sinal de entrada gaussiano complexo	68
3.3.1 Exemplo 1.....	68
3.3.2 Exemplo 2.....	70
3.3.3 Exemplo 3.....	71
3.3.4 Exemplo 4.....	74
3.4 Sinal de entrada gaussiano real	77
3.4.1 Exemplo 5.....	78
3.4.2 Exemplo 6.....	80
3.4.3 Exemplo 7.....	82
3.4.4 Exemplo 8.....	84
3.5 Desempenho do algoritmo em regime permanente	86

3.5.1 Exemplo 9	86
3.5.2 Exemplo 10	87
3.6 Conclusões	90
CAPÍTULO 4 - CONSIDERAÇÕES FINAIS.....	91
4.1 Introdução	91
4.2 Conclusões e comentários finais	91
4.3 Sugestões para trabalhos futuros	93
APÊNDICE A - CÁLCULO DE H.....	95
A.1 Sinais de entrada gaussianos complexos	96
A.1.1 Sinais de entrada correlacionados	98
A.1.2 Sinais de entrada não-correlacionados	99
A.2 Sinais de entrada gaussianos reais.....	99
A.2.1 Sinais de entrada correlacionados	100
A.2.2 Sinais de entrada não-correlacionados	102
APÊNDICE B - CÁLCULO DE T E P.....	103
B.1 Sinais de entrada gaussianos complexos	104
B.1.1 Sinais de entrada correlacionados.....	105
B.1.2 Sinais de entrada não-correlacionados.....	106
B.2 Sinais de entrada gaussianos reais.....	107
B.2.1 Sinais de entrada correlacionados.....	108
B.2.2 Sinais de entrada não-correlacionados.....	109
APÊNDICE C - CÁLCULO DE S.....	111
C.1 Sinais de entrada gaussianos complexos	111
C.1.1 Sinais de entrada correlacionados.....	112
C.1.2 Sinais de entrada não-correlacionados.....	112
C.2 Sinais de entrada gaussianos reais.....	113
C.2.1 Sinais de entrada correlacionados.....	113
C.2.2 Sinais de entrada não-correlacionados.....	113
APÊNDICE D - ANÁLISE DE [71].....	115
APÊNDICE E - SOLUÇÃO APROXIMADA PARA (2.32)	119
REFERÊNCIAS.....	121

INTRODUÇÃO

Nas últimas décadas, a filtragem adaptativa vem se consolidando como um tópico de pesquisa cada vez mais relevante para a área de processamento digital de sinais. Tal relevância é corroborada pela popularização dos processadores digitais de sinais (*digital signal processors* – DSPs) de alto desempenho. Os DSPs possibilitam a implementação de diferentes estruturas de filtragem como também diferentes algoritmos adaptativos para as mais diversas aplicações [1], [2], tornando assim sedutoras as aplicações resultantes devido à sua alta confiabilidade, robustez, precisão, flexibilidade e baixo custo [3].

O uso de filtragem adaptativa tem se mostrado como uma solução interessante, especialmente, nos casos em que não se tem conhecimento *a priori* das estatísticas dos sinais envolvidos no processo e/ou quando os sinais envolvidos variam no tempo. Tais características permitem o uso de filtros adaptativos em inúmeras aplicações práticas, dentre as quais se destacam: cancelamento de eco acústico ou de rede (envolvendo voz ou dados) [4]–[9], equalização de canal [1], [10], [11], cancelamento adaptativo de ruído [2], [4], [8], [9], controle ativo de ruído [1], [12], conformação de feixe em arranjo de antenas [1], [2], [10], [11] e codificação preditiva [1], [2], [11], [13]. De modo geral, em função da forma como os sinais envolvidos são caracterizados ou extraídos do ambiente de operação [3], todas as aplicações até então discutidas na literatura podem ser divididas em quatro classes distintas (cada qual com suas peculiaridades) denominadas: identificação de sistema, modelagem de sistema inverso, predição e cancelamento de interferências [1], [2], [11].

Devido ao seu vasto campo de aplicação, as pesquisas em filtragem adaptativa vêm despertando grande interesse e apresentando inúmeros desafios teóricos que persistem até o presente momento [10]. Dentre os principais tópicos de pesquisa na área, pode se destacar o desenvolvimento de novas estruturas de filtragem e de novos algoritmos adaptativos, a determinação de parâmetros ótimos para os algoritmos adaptativos e a modelagem estocástica de algoritmos adaptativos.

A modelagem estocástica de algoritmos adaptativos visa determinar expressões analíticas, isto é, um modelo matemático que descreva da melhor forma possível (mais fielmente) o comportamento do algoritmo sob estudo. Assim, para uma dada condição de operação, o modelo poderá prever (mesmo que de forma aproximada) o desempenho do algoritmo. Através dessa modelagem, uma análise mais detalhada do comportamento do algoritmo poderá ainda revelar possíveis situações indesejadas (anômalas), permitindo assim se propor modificações visando melhorar seu desempenho frente ao problema observado ou a alguma condição de operação específica. Além disso, a partir das expressões que descrevem o comportamento do algoritmo, podem ser derivadas também expressões para o seu comportamento em regime permanente, dessa forma, possibilitando o estabelecimento de relações de causa e efeito entre os parâmetros que controlam o algoritmo e algumas métricas usualmente utilizadas para avaliar seu desempenho.

Neste contexto, o presente trabalho tem como foco o desenvolvimento de um modelo estocástico para um consagrado algoritmo adaptativo, denominado algoritmo NLMS (*normalized least mean square*), considerando um problema de identificação de sistema com planta estacionária. Para tal, busca se determinar um modelo mais preciso do que os até então apresentados na literatura (considerando o menor número possível de hipóteses simplificativas), com solução analítica¹ e capaz de operar com diferentes tipos de sinais de entrada gaussianos (complexos, reais, correlacionados e não correlacionados). Particularmente, o desenvolvimento aqui apresentado visa estender a proposta de modelagem discutida em [14], a qual é válida apenas para sinais de entrada reais, utilizando uma abordagem similar à apresentada em [15]–[24].

Na sequência, no intuito de fornecer uma base teórica para o desenvolvimento do presente trabalho de pesquisa, alguns conceitos básicos relacionados à filtragem adaptativa são tratados. Primeiramente, a teoria de estimação linear é brevemente discutida, seguida dos fundamentos de filtragem adaptativa englobando as estruturas de filtragem, os algoritmos adaptativos [especificamente, o algoritmo LMS (*least mean square*) e o algoritmo NLMS] e as métricas de desempenho, visando um problema de identificação de sistema. Finalmente, os objetivos e a organização do manuscrito são apresentados.

¹ Cabe ressaltar que os termos solução analítica e formulação fechada são utilizados indistintamente ao longo do trabalho para indicar expressões cuja solução não requer o uso de técnicas numéricas.

1.1 Teoria de estimação linear

O tópico estimação trata da inferência de valores desconhecidos ou aleatórios a partir de um conjunto de dados (observações) relacionados com uma dada informação de interesse. Por exemplo, o processo de extrair determinada característica de um dado conjunto de observações pode ser visto como um problema de estimação. Da mesma forma, quando se infere sobre a entrada de um determinado sistema a partir das amostras obtidas na saída do mesmo, as quais geralmente são corrompidas por ruído, recai-se também em um problema de estimação [25]. O dispositivo responsável por extrair a informação de interesse, a partir de dados inerentemente ruidosos, é denominado estimador (ou filtro). Como tal definição é bastante ampla, a teoria de estimação (filtragem) encontra aplicações em diversos campos, a saber: sistemas de radar e/ou sonar, processamento de fala, análise de imagem, biomedicina, comunicações, sistemas de controle, sismologia e engenharia financeira [11], [26].

Os estímulos iniciais para o desenvolvimento da teoria de estimação linear foram suscitados por estudos astronômicos atribuídos a Galileu Galilei em 1632, em que se buscava minimizar diferentes funções de erro (funções custo). Contudo, a origem da teoria de estimação linear propriamente dita é creditada a Gauss que, com a idade de 18 anos em 1795, concebeu o método de mínimos quadrados para o estudo do movimento dos corpos celestes. No início do século XIX, estabeleceu-se uma grande controvérsia acerca da autoria do método dos mínimos quadrados devido ao fato de Gauss não ter publicado seu trabalho até então. Ao invés disso, de forma independente, tal método foi publicado por Legendre em 1805 [11].

Na área de estimação linear em processos estocásticos, trabalhos pioneiros foram apresentados, independentemente, por Kolmogorov [27] e Wiener [28] no final da década de 1930 e início da década de 1940, respectivamente. Kolmogorov, motivado pelo trabalho de Wold [29], propôs uma solução para um problema de predição linear em tempo discreto envolvendo processos estacionários. Por outro lado, Wiener formulou o problema de predição linear em tempo contínuo, derivando uma solução explícita para o estimador ótimo conhecida atualmente como equação de Wiener Hopf². Para tal, Wiener considerou um problema de estimação de um processo corrompido por ruído

² Tal equação, como originalmente formulada, envolve a solução de uma integral e estabelece o filtro (estimador) linear ótimo sujeito à restrição de causalidade.

aditivo, o qual basicamente pode ser visto como um problema de filtragem [10], [11].

Posteriormente, em 1947, Levinson reformulou o problema de filtragem de Wiener em tempo discreto (veja Figura 1.1) [11]. Nesse caso, a equação de Wiener Hopf assume a seguinte forma matricial [1], [3], [11], [30]:

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

onde \mathbf{w}_o representa o vetor de coeficientes ótimo que minimiza o erro de estimação $e(n)$ no sentido quadrático médio (considerando uma estrutura transversal de filtragem), \mathbf{R} é a matriz de autocorrelação do sinal de entrada $x(n)$ e \mathbf{p} , o vetor de correlação cruzada entre o sinal de entrada e a resposta desejada $d(n)$ [11].

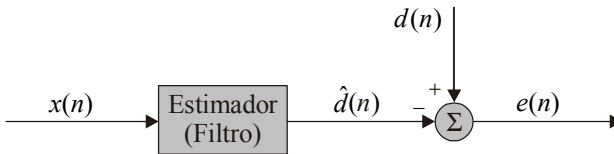


Figura 1.1: Diagrama de blocos de um problema de predição (adaptado de [11]).

Visto que o sinal de entrada é assumido estacionário no sentido amplo [25], a matriz de autocorrelação \mathbf{R} assume a forma de uma matriz Toeplitz [31]. Assim, considerando que \mathbf{R} é não-singular [31]-[33], pré-multiplicando ambos os lados de (1.1) por \mathbf{R}^{-1} , a solução de Wiener pode ser reescrita como [1], [2], [10], [11], [30]

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p}. \quad (1.2)$$

A partir de (1.2), observa-se que a solução de Wiener requer o conhecimento das características estatísticas dos sinais envolvidos, isto é, a matriz \mathbf{R} e o vetor \mathbf{p} . Entretanto, na prática, tais informações geralmente não são conhecidas *a priori*. Além disso, verifica-se que (1.2) envolve o cálculo da inversa de uma matriz cuja solução na maioria das vezes não pode ser obtida analiticamente, o que implica uma elevada complexidade computacional à solução. Dessa forma, visando contornar esses problemas para tornar a solução viável do ponto de vista de implementação prática, abordagens alternativas a (1.2) são mandatórias.

1.2 Fundamentos de filtragem adaptativa

No contexto de processamento de sinais, os filtros são dispositivos capazes de modificar os componentes de um sinal, seja removendo ou conformando os componentes indesejados de forma a facilitar a extração da informação de interesse [3]. Assim, conhecidas as características do sinal a ser filtrado (sinal de entrada), as especificações da filtragem requerida e a estrutura de filtragem a ser utilizada, é possível projetar um filtro com coeficientes fixos (invariante no tempo) para realizar a tarefa [2]. Contudo, como mencionado anteriormente, em muitas aplicações as informações e especificações requeridas na etapa de projeto não são completamente conhecidas *a priori* e/ou suas características variam no tempo. Nessas situações, o uso das técnicas convencionais de síntese de filtros com coeficientes fixos não podem ser mais utilizadas com sucesso [1]-[3], [11].

Diante das limitações dos filtros convencionais, visando propiciar um melhor desempenho e sem a necessidade de intervenção externa do projetista, o uso de um filtro com coeficientes adaptativos (variantes no tempo) torna-se essencial (veja Seção 1.2.1). Tais sistemas, denominados filtros adaptativos, possuem a capacidade de se autoprojetar ou se adaptar ao ambiente no qual estão inseridos, usando para isso um algoritmo recursivo que modifica sua resposta durante a operação de modo a atender os requisitos desejados [2], [11]. Portanto, baseado em algum critério de desempenho pré-definido, no sinal de entrada do sistema e na resposta desejada (sinal de referência), o filtro adaptativo ajusta seus coeficientes durante a operação [3], aprendendo e se adaptando às características do ambiente em questão [2].

Considerando a característica variante no tempo dos coeficientes de um filtro adaptativo, o diagrama de blocos mostrado na Figura 1.1 pode ser então reformulado conforme ilustrado na Figura 1.2. A partir desta última, observa-se que todo filtro adaptativo tem a ele associado uma estrutura de filtragem e um algoritmo adaptativo que, através de um mecanismo de controle, ajusta recursivamente os coeficientes do filtro de forma a minimizar uma determinada função custo, usualmente o erro quadrático médio (EQM). Com base em tal função custo e em alguns parâmetros de inicialização, o algoritmo adaptativo faz com que os coeficientes do filtro converjam iterativamente para a solução que minimiza a função custo considerada. Essa convergência pode ocorrer tanto em um ambiente estacionário quanto não-estacionário, contanto que, neste último, as estatísticas dos sinais envolvidos variem de forma

relativamente lenta ou pouco frequente, possibilitando ao filtro adaptativo exercer sua capacidade de rastreamento [1], [2], [9]-[11].

Em certas aplicações, para facilitar a manipulação matemática, o sinal de entrada e o sinal desejado são comumente representados utilizando a forma complexa [30], isto é, os valores das amostras dos sinais envolvidos são números complexos. Isso ocorre, por exemplo, na equalização de um canal de transmissão, onde técnicas de modulação de sinal como o PSK (*phase shift keying*) ou o QAM (*quadrature amplitude modulation*) são utilizadas [1], [11], [30]. Situações semelhantes podem ser encontradas em sistemas de radar e sonar, em arranjo de antenas [10], [11], ou na implementação de filtros adaptativos no domínio transformado (e/ou no domínio da frequência como introduzido por Narayan *et al.* [34]). Neste último, dependendo da transformação utilizada, o filtro adaptativo pode operar internamente com sinais complexos, mesmo quando os sinais externos são reais [1]. Assim, visando tratar o caso mais geral, no desenvolvimento deste trabalho, assume-se o uso de sinais complexos. Salienta-se que, a partir da forma complexa, a correspondente forma real do filtro adaptativo pode ser diretamente obtida. Consequentemente, as estruturas de filtragem e os algoritmos adaptativos representados na forma complexa correspondem ao caso mais geral, incluindo a representação na forma real como um caso particular [11]. Nesse contexto, considerando a forma complexa, a seguir as estruturas de filtragem e os algoritmos adaptativos são descritos.

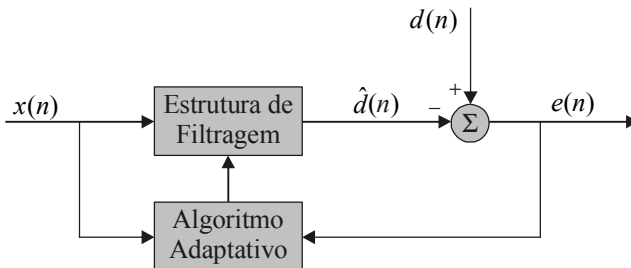


Figura 1.2: Diagrama de blocos de um filtro adaptativo (adaptado de [3]).

1.2.1 Estruturas de filtragem

Conforme mencionado anteriormente, todo filtro adaptativo tem a ele associado uma estrutura de filtragem cujos coeficientes são ajustados iterativamente por um algoritmo recursivo. Como consequência, devido à contínua variação de seus coeficientes, tem-se que um filtro adaptativo

não é, a rigor, um sistema linear uma vez que não satisfaz o princípio da superposição durante sua adaptação. Apesar disso, os filtros adaptativos são comumente classificados como lineares ou não-lineares dependendo da estrutura de filtragem utilizada, isto é, desconsiderando sua característica adaptativa [35]. Portanto, considerando coeficientes fixos, se a relação de entrada e saída obedece ao princípio da superposição, o filtro adaptativo é dito linear, caso contrário, é dito não-linear [11].

Embora a maioria das aplicações sejam restritas ao uso de filtros lineares, os filtros adaptativos não-lineares são indispensáveis em aplicações onde os fenômenos físicos a serem modelados são fortemente não-lineares [1]. Dentre essas aplicações, destacam-se a equalização de sistemas de comunicação por satélite, o processamento de sinais dependente da percepção e o estudo das distorções harmônicas em alto faltantes [35]. No entanto, as estruturas de filtragem não-lineares estão fora do escopo do presente trabalho de pesquisa (para detalhes veja [35]).

Com respeito aos filtros adaptativos lineares, as diversas estruturas de filtragem utilizadas podem ser classificadas em dois grandes grupos, dependendo da característica da resposta ao impulso, a saber: estruturas com resposta ao impulso infinita (*infinite impulse response* - IIR) e estruturas com resposta ao impulso finita (*finite impulse response* - FIR) [2], [3], [10], [11]. As estruturas IIR têm como principal característica a presença de caminhos de realimentação que acarretam problemas de instabilidade na estrutura, a menos que cuidados especiais sejam tomados com relação à alocação dos pólos da função de transferência do filtro adaptativo. Além disso, em uma estrutura IIR adaptativa, a função custo geralmente possui mínimos locais, o que pode resultar na convergência do filtro para um desses mínimos locais ao invés do mínimo global (como desejado) [1], [3], [11]. Para contornar tais problemas, as estruturas FIR (filtro transversal, preditor *lattice* e arranjo sistólico) têm se mostrado atrativas do ponto de vista prático, devido à inerente estabilidade e ao fato de que a respectiva função custo é geralmente convexa e tem um único mínimo [11]. Dentre tais estruturas, em função da sua versatilidade e facilidade de implementação, o filtro transversal (comumente denominado filtro de linha de retardo ou apenas filtro FIR) vem sendo a estrutura mais utilizada em implementações práticas de filtros adaptativos [1], [11], [30], [35]. Essa estrutura, introduzida por H. E. Kallmann em 1940 [36], tem emergido como uma ferramenta essencial para o processamento digital de sinais em uma vasta gama de aplicações, incluindo as que envolvem a filtragem adaptativa [1], [11].

A Figura 1.3 ilustra o diagrama de blocos de um filtro adaptativo considerando uma estrutura transversal de filtragem (destacada pela região pontilhada). Consta-se que o sinal de saída do filtro adaptativo, nesse caso a estimativa do sinal desejado $\hat{d}(n)$, é gerado como uma combinação linear das amostras atrasadas do sinal de entrada com os coeficientes do filtro adaptativo. Dessa forma, a relação de entrada e saída pode ser expressa como [1]-[3], [11], [30]

$$\begin{aligned}\hat{d}(n) &= \sum_{k=0}^{M-1} w_k^*(n)x(n-k) \\ &= \mathbf{w}^H(n)\mathbf{x}(n)\end{aligned}\quad (1.3)$$

onde $\mathbf{w}(n)=[w_0(n) \ \cdots \ w_{M-1}(n)]^T$ denota o vetor de coeficientes do filtro adaptativo, $\mathbf{x}(n)=[x(n) \ \cdots \ x(n-M+1)]^T$, o vetor com as amostras mais recentes do sinal de entrada (vetor do sinal de entrada), M , a ordem do filtro, $*$, o complexo-conjugado, e $(\cdot)^H$, o complexo-conjugado transposto (hermitiano). Ainda referente à Figura 1.3, verifica-se que o sinal de erro (também denominado erro de estimação) pode ser escrito como

$$\begin{aligned}e(n) &= d(n) - \hat{d}(n) \\ &= d(n) - \mathbf{w}^H(n)\mathbf{x}(n).\end{aligned}\quad (1.4)$$

Note que, a partir de (1.4), assumindo que os coeficientes do filtro adaptativo são fixos, a solução de Wiener pode ser obtida através da

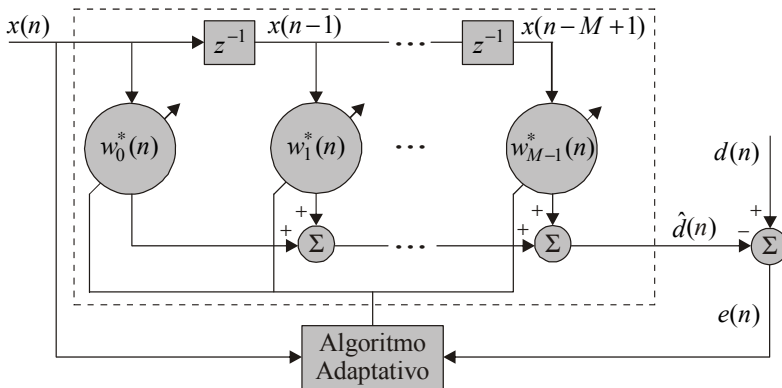


Figura 1.3: Diagrama de blocos de um filtro adaptativo considerando uma estrutura transversal de filtragem (adaptado de [1] e [11]).

minimização do EQM. De maneira similar, a maioria dos algoritmos adaptativos disponíveis na literatura podem ser obtidos considerando a minimização de alguma função custo envolvendo (1.4).

1.2.2 Algoritmos adaptativos

Como mencionado, um filtro adaptativo é caracterizado basicamente por dois blocos básicos: uma estrutura de filtragem, cujos coeficientes são atualizados conforme a necessidade da aplicação; e um algoritmo adaptativo que, baseado no sinal de entrada, no sinal desejado e no sinal de erro, adapta os coeficientes da estrutura de filtragem. Logo, o algoritmo adaptativo é responsável por ajustar os coeficientes do filtro adaptativo visando atender um dado critério de desempenho pré-estabelecido (por exemplo, a minimização do EQM). Esse algoritmo é geralmente determinado a partir de uma função custo (envolvendo, por exemplo, o sinal de erro), visando fazer com que (para uma condição inicial arbitrária) $\mathbf{w}(n)$ convirja para a solução que minimiza a função custo considerada [1], [3], [10].

Um método iterativo de minimização que vem sendo comumente utilizado devido à sua simplicidade é denominado método da descida mais íngreme (*steepest descent*) [1], [11]. A partir de tal método, a solução de Wiener [veja (1.2)] pode ser obtida recursivamente através da seguinte expressão [1], [3], [9], [11]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu \nabla J(n) \quad (1.5)$$

onde μ é o passo de adaptação que controla a velocidade de convergência do algoritmo,

$$\nabla J(n) \triangleq \frac{\partial J(n)}{\partial \mathbf{w}^*(n)} \quad (1.6)$$

representa o vetor gradiente da função custo $J(n)$, a qual é definida como [1], [10], [11]

$$J(n) \triangleq E[|e(n)|^2] \quad (1.7)$$

com $E(\cdot)$ e $|\cdot|$ denotando, respectivamente, o valor esperado e o módulo. Assim, substituindo (1.4) em (1.7) e assumindo que o vetor de sinal de entrada e o sinal desejado são conjuntamente estacionários, a função custo (em um dado instante n) pode ser reescrita como

$$J(n) = E[|d(n)|^2] - \mathbf{p}^H \mathbf{w}(n) - \mathbf{w}^H(n) \mathbf{p} + \mathbf{w}^H(n) \mathbf{R} \mathbf{w}(n) \quad (1.8)$$

onde $\mathbf{R} \triangleq E[\mathbf{x}(n)\mathbf{x}^H(n)]$ e $\mathbf{p} \triangleq E[\mathbf{x}(n)d^*(n)]$. Finalmente, como indicado em (1.6), derivando (1.8) em relação a $\mathbf{w}^*(n)$ e substituindo a

expressão resultante em (1.5), tem-se [1], [3], [9], [11], [30]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu[\mathbf{p} - \mathbf{R}\mathbf{w}(n)] \quad (1.9)$$

como solução recursiva para (1.2).

A partir de (1.2) e (1.9), constata-se que tanto a solução recursiva obtida pelo método da descida mais íngreme quanto à solução dada por (1.2) requerem o conhecimento *a priori* das estatísticas dos sinais envolvidos; informações que na prática não estão prontamente disponíveis [1]. Dessa forma, objetivando implementações práticas, versões simplificadas do método da descida mais íngreme são usualmente utilizadas, as quais dão origem a muitos dos algoritmos adaptativos disponíveis atualmente na literatura (como, por exemplo, os algoritmos da família LMS) [10]. Tais algoritmos, pertencentes à classe dos algoritmos de gradiente estocástico³, não requerem o conhecimento *a priori* das estatísticas dos sinais envolvidos, pois fazem uso de estimativas instantâneas do gradiente (obtidas a partir de dados instantâneos). Logo, devido principalmente ao uso de valores instantâneos dos sinais envolvidos, pode-se dizer que esses algoritmos incorporam mecanismos de aprendizagem que permitem rastrear variações nas características dos sinais e/ou variações na dinâmica do processo em que o filtro adaptativo está inserido [10], [11].

Como principais vantagens dos algoritmos baseados no gradiente estocástico, pode-se citar a baixa complexidade computacional e a elevada robustez numérica [1], [10], [11], [30]; características que têm motivado o uso de algoritmos tais como o LMS [1], [9]-[11], [37], [38], NLMS [1], [3], [10], [11], [39]-[41], TD-LMS (*transform-domain* LMS) [1], [3], [11], [34], algoritmos de passo variável (VSS-LMS) [42]-[45], algoritmos proporcionais (PNLMS) [46]-[50], dentre outros, nas mais diversas aplicações. Além disso, a cada dia novos algoritmos são propostos na literatura visando tratar condições específicas de operação ou reduzir as limitações de algoritmos já consagrados. No que se refere aos algoritmos baseados no gradiente estocástico, os algoritmos LMS e NLMS (discutidos a seguir) vêm tendo certo destaque no decorrer dos últimos anos [10].

³ O termo “gradiente estocástico” vem sendo utilizado para distinguir o algoritmo LMS [veja (1.16)] do método da descida mais íngreme [veja (1.9)]. Note que, o método da descida mais íngreme faz uso de um valor determinístico do gradiente para o cálculo recursivo do filtro de Wiener, enquanto que os algoritmos baseados no gradiente estocástico utilizam estimativas instantâneas do gradiente, o que implica que o vetor com a direção de atualização está sujeito a flutuações aleatórias denominadas ruído de gradiente [10], [11].

1.2.2.1 Algoritmo LMS

O algoritmo LMS foi originalmente proposto por Widrow e Hoff em 1960 no contexto de um estudo sobre reconhecimento de padrões (comumente referido na literatura como *Adaline*) [37]. Posteriormente, em 1975, tal algoritmo foi estendido para lidar com o caso de dados complexos [51], exibindo um formato diferente do correntemente apresentado na literatura devido à forma como a matriz de autocorrelação⁴ é definida. Em muitos aspectos, considera-se o algoritmo LMS como sendo um marco na teoria de filtragem adaptativa moderna. Desde a sua criação, esse algoritmo vem sendo estudado sob diferentes ângulos, sempre mostrando notável resiliência. Nas teorias de estimação e filtragem adaptativa, poucos algoritmos foram utilizados com tanto sucesso em tantas áreas distintas [10]. Atualmente, o algoritmo LMS é considerado por muitos como o mais utilizado em aplicações envolvendo filtragem adaptativa [1], [3], [10], [11], [30], sendo essa popularidade devida, principalmente, à sua robustez numérica e à baixa complexidade computacional; características bastante atrativas em aplicações práticas.

No algoritmo LMS, a adaptação do vetor de coeficientes do filtro se dá na direção do negativo do gradiente instantâneo do módulo ao quadrado do sinal de erro [com respeito à $\mathbf{w}^*(n)$]. Assim, devido ao uso da estimativa instantânea (ruidosa) do vetor gradiente, o algoritmo percorre a superfície de desempenho em uma trajetória irregular, cujo curso médio é o caminho da descida mais íngreme (ou seja, obtém-se na média a convergência para a solução de Wiener [3], [10], [11]). Portanto, em contraste com o método da descida mais íngreme [veja (1.5)-(1.9)], o algoritmo LMS utiliza uma aproximação estocástica do vetor gradiente $\nabla \hat{J}(n)$, obtida a partir de dados instantâneos disponíveis no sistema. Dessa forma,

$$\nabla \hat{J}(n) = \frac{\partial \hat{J}(n)}{\partial \mathbf{w}^*(n)} \quad (1.10)$$

com

$$\hat{J}(n) = |e(n)|^2. \quad (1.11)$$

⁴ Widrow e outros autores basearam suas derivações na seguinte definição para a matriz de autocorrelação do sinal de entrada: $\mathbf{R} = E[\mathbf{x}^*(n)\mathbf{x}^T(n)]$. Entretanto, atualmente, tem-se adotado $\mathbf{R} = E[\mathbf{x}(n)\mathbf{x}^H(n)]$ [11].

Então, substituindo (1.11) em (1.10) e utilizando a regra da cadeia na derivada, tem-se

$$\nabla \hat{J}(n) = e(n) \frac{\partial e^*(n)}{\partial \mathbf{w}^*(n)} + e^*(n) \frac{\partial e(n)}{\partial \mathbf{w}^*(n)}. \quad (1.12)$$

Considerando agora (1.4), as derivadas⁵ parciais presentes em (1.12) são obtidas como

$$\begin{aligned} \frac{\partial e^*(n)}{\partial \mathbf{w}^*(n)} &= \frac{\partial}{\partial \mathbf{w}^*(n)} [d^*(n) - \mathbf{x}^H(n) \mathbf{w}(n)] \\ &= 0 \end{aligned} \quad (1.13)$$

e

$$\begin{aligned} \frac{\partial e(n)}{\partial \mathbf{w}^*(n)} &= \frac{\partial}{\partial \mathbf{w}^*(n)} [d(n) - \mathbf{w}^H(n) \mathbf{x}(n)] \\ &= -\mathbf{x}(n). \end{aligned} \quad (1.14)$$

Logo, substituindo (1.13) e (1.14) em (1.12), a estimativa do vetor gradiente é

$$\nabla \hat{J}(n) = -e^*(n) \mathbf{x}(n). \quad (1.15)$$

Finalmente, a partir de (1.5) e (1.15), a equação de atualização dos coeficientes do algoritmo LMS, considerando a forma complexa⁶, pode ser escrita como [1], [3], [11], [30]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \mathbf{x}(n) e^*(n) \quad (1.16)$$

sendo o sinal de erro obtido através de (1.4). Salienta-se que o desenvolvimento conduzido aqui na derivação do algoritmo LMS se baseia na concepção clássica introduzida em [9], [37] e [51], a qual é motivada pela escassez de informações sobre as estatísticas dos sinais envolvidos. No entanto, outra abordagem largamente usada na derivação do algoritmo LMS se dá através de um processo de otimização com restrição, considerando o princípio do mínimo distúrbio [10].

No algoritmo LMS, a velocidade de convergência é afetada diretamente pelas variações de amplitude do sinal de entrada, isto é, quando o nível de potência do sinal de entrada é alto, o algoritmo

⁵ As derivadas utilizadas são: $\partial(\mathbf{a}^H \boldsymbol{\beta}) / \partial \mathbf{a}^* = \boldsymbol{\beta}$; $\partial(\boldsymbol{\beta}^H \mathbf{a}) / \partial \mathbf{a}^* = \mathbf{0}$ [3], [11].

⁶ No caso de dados complexos, o fator 2 que multiplica o passo de adaptação não se faz presente na equação de atualização do vetor de coeficientes devido à forma como é definida a derivada complexa [3], [10], [11]. Geralmente, no caso de dados reais o fator 2 é incorporado no passo de adaptação do algoritmo.

amplifica o ruído de gradiente podendo acarretar certa instabilidade no sistema. Por outro lado, quando o nível de potência do sinal de entrada é baixo, a convergência torna-se lenta [10], [11]. Além disso, visando reduzir o efeito do ruído de gradiente, muitas vezes o passo de adaptação é mantido pequeno de forma que o algoritmo convirja mais lentamente [11]. Outra desvantagem do algoritmo LMS é que sua velocidade de convergência também é afetada (tornando-se mais lenta) com o aumento da dispersão dos autovalores da matriz de autocorrelação do sinal de entrada [1], [10], [11]. Assim, visando contornar tais problemas, diversas estratégias vêm sendo propostas na literatura, resultando em algoritmos mais aprimorados (tais como o algoritmo NLMS) [24].

1.2.2.2 Algoritmo NLMS

O algoritmo NLMS foi originalmente proposto por Kaczmarz [39] como um algoritmo de projeções em 1937 [52]-[54]. No entanto, na literatura corrente, essa descoberta é atribuída a Nagumo e Noda [40] e/ou Albert e Gardner [41] que, de forma independente, apresentaram o mesmo resultado em 1967. A terminologia atual de algoritmo LMS normalizado (NLMS) foi cunhada por Bitmead e Anderson [55] em 1980, em função da única distinção entre o algoritmo LMS e NLMS residir na normalização do passo de adaptação pela norma euclidiana ao quadrado do vetor de sinal de entrada [11], [14]. Tal procedimento reduz a sensibilidade do algoritmo a variações na potência do sinal de entrada [54], [56], tornando assim o NLMS atrativo para aplicações como o cancelamento adaptativo de ruído e o cancelamento de eco de rede [5], [10], [54]. A sua robustez também pode ser verificada para sinais altamente correlacionados, nos quais o efeito da normalização se reflete através da redução da dispersão de autovalores da matriz de autocorrelação do sinal de entrada [53].

A equação de adaptação do algoritmo NLMS é dada por [11]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{\mathbf{x}(n)e^*(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \quad (1.17)$$

sendo o sinal de erro calculado com o auxílio de (1.4). Aqui, vale ressaltar que diversas abordagens podem ser utilizadas para se obter a equação de atualização dos coeficientes do algoritmo NLMS, podendo-se citar como principais: a solução de um problema de otimização com restrição (utilizando o princípio do mínimo distúrbio) [10], [11], [53]; a aproximação estocástica do método de Newton [10]; a

minimização do erro *a posteriori* [1]; ou ainda como um caso particular do algoritmo de projeções afins (*affine projection algorithm* - AP), considerando apenas um vetor de dados [54].

Em implementações práticas, visando evitar divisão por zero ou *overflow*, é usual a adição de uma pequena constante positiva ε (denominada parâmetro de regularização) ao denominador de (1.17) [10], [55]-[58]. Tais situações ocorrem quando a potência do sinal de entrada é muito pequena, como, por exemplo, em períodos de silêncio no processamento de um sinal de fala. Assim, (1.17) é reescrita como

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu \frac{\mathbf{x}(n)e^*(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \quad (1.18)$$

sendo agora denominado algoritmo ε -NLMS [10]. Apesar de o parâmetro de regularização ser escolhido independentemente do passo de adaptação, é sabido que ele afeta o erro em regime permanente, devendo assim ser cuidadosamente escolhido [59] (como regra geral, adota-se $\varepsilon \ll E[\mathbf{x}^H(n)\mathbf{x}(n)]$ [60]). Portanto, para uma dada condição inicial $\mathbf{w}(0)$, a expressão (1.18) adapta iterativamente os coeficientes da estrutura de filtragem de acordo com o algoritmo ε -NLMS.

1.2.3 Métricas de desempenho

A escolha de uma estrutura de filtragem e de um algoritmo adaptativo a serem adotados em um sistema é, em geral, uma tarefa complexa que requer o maior número possível de informações sobre a aplicação em questão. Assim, visando auxiliar o projetista nessas escolhas, tendo em vista a vasta gama de estruturas de filtragem e algoritmos adaptativos disponíveis na literatura, as seguintes métricas de desempenho usualmente consideradas na análise e projeto de filtros adaptativos podem ser utilizadas:

- *Velocidade de convergência*: representa o número de iterações necessárias para que o algoritmo, perante sinais estacionários, convirja o mais próximo possível da solução ótima. Consequentemente, uma alta velocidade de convergência permite ao filtro adaptativo se ajustar rapidamente a um ambiente desconhecido [11].
- *Desajuste*: é um parâmetro adimensional que fornece uma medida quantitativa do desvio introduzido pelo algoritmo adaptativo em relação ao EQM mínimo J_{\min} [11]. Tal métrica é definida como [2], [9]-[11]

$$\begin{aligned} \mathcal{M} &\triangleq \frac{J(\infty) - J_{\min}}{J_{\min}} \\ &\triangleq \frac{J_{\text{ex}}(\infty)}{J_{\min}} \end{aligned} \quad (1.19)$$

onde $J(\infty)$ corresponde ao EQM em regime permanente, dado por

$$J(\infty) \triangleq \lim_{n \rightarrow \infty} E[|e(n)|^2] \quad (1.20)$$

e $J_{\text{ex}}(\infty)$ denota o EQM em excesso em regime permanente [10], [38].

- *Rastreabilidade*: corresponde à capacidade do algoritmo reagir às mudanças no ambiente em que o filtro adaptativo está inserido, ou seja, representa a capacidade do algoritmo de acompanhar as variações nas estatísticas dos sinais envolvidos no processo [10], [38].
- *Robustez*: refere-se à capacidade de um algoritmo adaptativo operar satisfatoriamente com sinais mal condicionados, ou seja, à sua capacidade de exibir baixa sensibilidade a pequenos distúrbios internos ou externos ao sistema [10], [11].
- *Requisitos computacionais*: mensurados geralmente em números de operações requeridas para realizar uma iteração, isto é, levando em conta a quantidade de adições, multiplicações, divisões, comparações, e acessos à memória, dentre outras operações [11].
- *Estrutura*: se refere à maneira como a informação veicula dentro do algoritmo. Nesse quesito, certas estruturas apresentam vantagens devido, principalmente, a sua forma modular, simplicidade de implementação e robustez numérica [10], [11].
- *Estabilidade numérica*: trata de uma característica inerente ao modo como as equações são determinadas e resolvidas pelo algoritmo, o qual é considerado numericamente instável quando os erros decorrentes dos efeitos de quantização se acumulam levando-o a um comportamento anômalo, e numericamente estável (robusto), quando ele é pouco sensível a variações no comprimento da palavra utilizada em sua implementação digital [11].

Como o uso da filtragem adaptativa em aplicações práticas é bem diversificado, a solução desenvolvida para uma dada aplicação pode não ser diretamente aplicável em outra. Assim, para se ter sucesso no projeto de dispositivos contendo filtros adaptativos, é necessário compreender o ambiente físico de operação do sistema, relacionando-o com a realidade da aplicação de interesse. Afeito a isso, pode-se dizer que as métricas usualmente empregadas podem não ser úteis a determinadas aplicações práticas, devendo o projetista encontrar outras formas de mensurar a eficácia da solução desenvolvida, criando ainda medidas alternativas quando necessário.

1.2.4 Problema de identificação de sistema

Na classe de aplicações de identificação de sistema (também denominada modelagem de sistema [1]), busca-se através de um filtro adaptativo obter a melhor estimativa possível da resposta ao impulso do sistema desconhecido (planta) [2], [11]. No caso de sistemas de controle, o filtro adaptativo tem por objetivo estimar os parâmetros ou o estado do sistema e então, usando tais informações, projetar um controlador eficiente [1], [2]. De forma similar, em sistemas de cancelamento de eco de rede, proveniente de aplicações envolvendo telefonia, busca-se modelar o caminho de eco (canal de transmissão), visando gerar uma estimativa do eco presente na linha de maneira a cancelá-lo por subtração no domínio elétrico [1], [2], [4].

Na Figura 1.4, é apresentado um diagrama de blocos ilustrando um problema de identificação de sistema. Observa-se que, nesse tipo de configuração, tanto o sistema a ser identificado quanto o filtro adaptativo são excitados com um mesmo sinal de entrada. O sinal de saída da planta é corrompido por um ruído aditivo de medição $z(n)$, resultando no sinal desejado. O ruído de medição (proveniente de diversos fatores) corresponde à parcela que não pode ser modelada pelo filtro adaptativo. Dessa forma, assumindo convergência, conclui-se que o sinal de erro tem como seu principal componente o ruído de medição [2], [11].

Diante do até então apresentado, no desenvolvimento do presente trabalho de pesquisa assume-se que tanto o sistema desconhecido quanto o filtro adaptativo (que é adaptado pelo algoritmo NLMS) são representados por um filtro FIR de mesma ordem, denotados w_0 e $w(n)$, respectivamente, e que o ruído de medição é independente de qualquer outro sinal do sistema. Nesse contexto, são apresentados a seguir os objetivos deste trabalho de pesquisa.

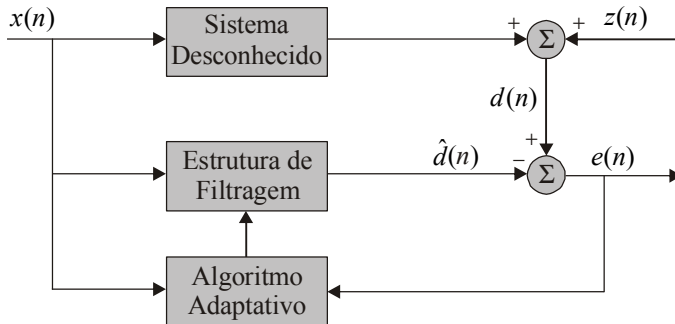


Figura 1.4: Diagrama em blocos de um problema de identificação de sistema (adaptado de [1], [2] e [11]).

1.3 Objetivos do trabalho

Em resumo, os principais objetivos deste trabalho são:

- Desenvolver um modelo estocástico para o algoritmo NLMS, considerando o menor número possível de hipóteses simplificativas, com formulação fechada e hábil a operar tanto com dados reais quanto complexos.
- Determinar soluções analíticas para o cálculo das matrizes de autocorrelação normalizadas [veja (2.1)], válidas para dados de entrada gaussianos reais, complexos, correlacionados e não-correlacionados.
- Derivar expressões fechadas que relacionem algumas métricas de desempenho com os parâmetros de controle do algoritmo, obtendo assim expressões que descrevem adequadamente o comportamento do algoritmo em regime permanente.
- Relacionar os algoritmos LMS e NLMS através do passo de adaptação (para o caso de sinais de entrada não-correlacionados).
- Realizar uma revisão bibliográfica, visando catalogar os modelos estocásticos (mais relevantes) disponíveis na literatura para o algoritmo NLMS.
- Comparar o modelo proposto com os demais modelos apresentados na literatura para o algoritmo NLMS, visando avaliar suas precisões.
- Fornecer novas ferramentas para a análise de algoritmos adaptativos com sinal de entrada normalizado.

1.4 Estrutura do trabalho

Este trabalho está organizado em quatro capítulos como segue. No Capítulo 1, é apresentada uma breve discussão sobre os fundamentos de filtragem adaptativa e os objetivos desta dissertação. No Capítulo 2, inicialmente, são abordados aspectos filosóficos referentes ao processo de modelagem de algoritmos adaptativos, seguida de uma revisão bibliográfica dos trabalhos mais relevantes encontrados na literatura referentes à modelagem estocástica do algoritmo NLMS. Ainda no Capítulo 2, é proposto um modelo estocástico para o algoritmo NLMS, considerando um problema de identificação de sistemas com planta estacionária, juntamente com expressões que descrevem o comportamento do algoritmo em regime permanente. O Capítulo 3 traz os resultados de simulação, incluindo resultados comparativos entre o modelo proposto e diversos modelos disponíveis na literatura para o algoritmo NLMS. Finalmente, no Capítulo 4, são expostas as considerações finais destacando as conclusões, comentários finais e sugestões para trabalhos futuros.

MODELAGEM DO ALGORITMO NLMS

2.1 Introdução

Neste capítulo, considerando sinais de entrada gaussianos, é derivado um modelo estocástico para o algoritmo NLMS operando em um problema de identificação de sistema com planta estacionária. O modelo proposto contempla o caso mais geral do algoritmo (dados complexos), sendo o caso de dados reais uma situação particular deste último. Nesse contexto, são apresentadas expressões que descrevem o comportamento médio do vetor de coeficientes, a matriz de covariância do vetor de erro nos coeficientes e a curva de aprendizagem do algoritmo. Com base nesses resultados, são derivadas expressões analíticas para o erro quadrático médio em excesso em regime permanente e o desajuste. Tais figuras de mérito nos permitem inferir sobre o funcionamento do algoritmo adaptativo, propiciando assim uma melhor compreensão de como e quais parâmetros afetam o seu desempenho. Adicionalmente, o efeito da normalização na superfície de desempenho é analisado, sendo ainda algumas relações entre os algoritmos NLMS e LMS estabelecidas.

2.2 Modelagem estocástica de algoritmos adaptativos

O funcionamento adequado de um filtro adaptativo depende diretamente da escolha apropriada dos parâmetros que controlam o algoritmo. Tal escolha é geralmente um processo dispendioso que requer muitas simulações. No entanto, o uso de um modelo matemático que descreva o comportamento médio do algoritmo pode ser um agente facilitador ao projetista. Esses modelos, quando alimentados com dados referentes às características da aplicação desejada e dos sinais envolvidos, têm a capacidade de prever (mesmo que de forma aproximada) o comportamento do sistema em questão e, com isso, facilitar, dentre outros aspectos, a escolha dos parâmetros do algoritmo [17].

A modelagem de algoritmos adaptativos não é uma tarefa trivial, requerendo um vasto conhecimento de diversas ferramentas

matemáticas e apresentando ainda grandes desafios teóricos para serem superados. Apesar dos esforços até então realizados, não existe uma teoria unificada de análise para todos os algoritmos encontrados na literatura. Assim, os diversos algoritmos adaptativos devem ser tratados separadamente levando em consideração suas peculiaridades, bem como o tipo do sinal de entrada (real ou complexo) e suas características estatísticas (correlacionado ou não-correlacionado) [10]. Além disso, devido à característica recursiva dos algoritmos adaptativos, mesmo aqueles relativamente mais simples (tal como o LMS) apresentam certa complexidade matemática em sua análise estocástica [11].

Visando então tornar possível o tratamento matemático, hipóteses simplificativas são eventualmente utilizadas na modelagem de algoritmos adaptativos. Essas hipóteses estão em geral relacionadas a condições específicas de operação, tais como passo de adaptação, ordem do filtro, características do sinal de entrada e aproximações estatísticas consideradas. Geralmente, o uso de um maior número de hipóteses simplificativas implica um modelo mais restrito, isto é, com limitada aplicabilidade prática. Por outro lado, o uso de poucas aproximações torna o modelo mais preciso e menos limitado, porém invariavelmente aumenta a complexidade matemática em sua derivação [18]. Além do mais, é lógico se admitir que a qualidade do modelo esteja diretamente relacionada à validade das hipóteses utilizadas em sua derivação, isto é, a conjectura estabelecida na hipótese deve ser consistente com o comportamento efetivo do algoritmo. Nesse contexto, fica evidente que existe um forte compromisso entre complexidade matemática, quantidade e validade das hipóteses simplificativas utilizadas durante o processo de modelagem do algoritmo, e precisão do modelo estocástico obtido.

2.3 Modelos apresentados na literatura

Em relação à modelagem do algoritmo NLMS, o maior problema reside no cálculo de valores esperados do seguinte tipo [17], [21], [56], [61]:

$$\mathbf{R}^N = \mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right]. \quad (2.1)$$

Na literatura, tais valores esperados são conhecidos por momentos normalizados [62] (ou matrizes de autocorrelação normalizadas) devido à presença do fator $\mathbf{x}^H(n)\mathbf{x}(n)$ em seu denominador. Com respeito ao cálculo desses momentos, o trabalho desenvolvido em [14] tem tido

grande destaque na literatura. Lá, uma técnica que não requer aproximações é proposta; porém, soluções analíticas são obtidas apenas para algumas situações particulares e somente para o caso de sinais de entrada gaussianos reais [15]-[18], [21], [23], [24].

Em um posterior trabalho [60], o mesmo autor de [14] desenvolveu um modelo para o algoritmo ε -NLMS. No entanto, mesmo para o caso de sinais de entrada brancos, uma solução fechada para o cálculo dos momentos normalizados não pôde ser obtida. Nesse caso, como fator complicador, além da integral hiperelíptica¹ como em [14], tem-se a presença de uma função não-racional no numerador da integral, referente ao parâmetro de regularização. Tal parâmetro inviabiliza a determinação de soluções analíticas, sendo necessário, então, recorrer a técnicas de integração numérica [63], o que torna a solução computacionalmente complexa e de difícil interpretação teórica.

Seguindo esta linha cronológica, mesmo sabendo que algumas falhas foram apontadas por Morgan [64], o trabalho de Tarrab e Feuer [65] também é de grande relevância. Lá, importantes relações entre os algoritmos LMS e NLMS são estabelecidas para o caso de sinais de entrada reais não-correlacionados, as quais são aqui estendidas para o caso de dados complexos (veja Seção 2.5). No que se refere a sinais de entrada correlacionados, apenas um caso muito particular é analisado em [52]. No entanto, tal solução recai no cálculo de uma série hipergeométrica infinita que não possui solução analítica.

Em 1993, Rupp [62] publicou um trabalho no qual fez uso da técnica proposta em [14] para desenvolver o modelo dos algoritmos LMS e NLMS para sinais de entrada obtidos por um processo aleatório esféricamente invariante² (*spherically invariant random process - SIRP*). Contudo, para sinais não-gaussianos, o termo

$$m_u = E \left[\frac{1}{\mathbf{u}^T(n)\mathbf{u}(n)} \right], \quad (2.2)$$

requerido no cálculo do momento de segunda ordem do vetor de erro nos coeficientes, não possui solução analítica (devendo ser obtido

¹ Uma integral abeliana, também denominada hiperelíptica, tem a seguinte forma: $\int_0^x [R(t)]^{-1/2} dt$, em que $R(t)$ é um polinômio de grau maior do que quatro [15]-[18], [21], [84], [87].

² É um processo estocástico estacionário no qual as funções de densidade de probabilidade conjunta podem ser calculadas a partir de uma função de densidade de probabilidade unidimensional, sendo o processo gaussiano um caso particular [62].

eventualmente por simulação numérica). Referente à (2.2), $\mathbf{u}(n)$ representa um processo aleatório não-correlacionado obtido através da relação $\mathbf{u}(n) = \mathbf{L}^{-1}\mathbf{x}(n)$, sendo $\mathbf{x}(n)$ o processo correlacionado e \mathbf{L} uma matriz triangular inferior proveniente da decomposição de Cholesky da matriz de autocorrelação do sinal de entrada. Ainda em 1993, Slock [52] sugeriu o uso de um modelo mais simples para representar o sinal de entrada, possibilitando assim a sua decomposição em três parcelas independentes. Através dessa abordagem, soluções analíticas são apresentadas somente para algumas condições específicas semelhantes a [14]. Logo, a questão de sinais de entrada correlacionados e/ou complexos permanece ainda um problema em aberto na literatura.

Diante das dificuldades previamente mencionadas e visando à obtenção de soluções analíticas para os valores esperados normalizados, algumas aproximações são comumente consideradas, as quais resultam quase sempre em modelos imprecisos, principalmente, quando o sinal de entrada é correlacionado. Por exemplo, em [11] e [57], é utilizada a aproximação, denominada aqui princípio da média modificado³ (PMM), dada por

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] \cong \frac{E[\mathbf{x}(n)\mathbf{x}^H(n)]}{E[\mathbf{x}^H(n)\mathbf{x}(n)]} = \frac{\mathbf{R}}{M\sigma_x^2} \quad (2.3)$$

onde σ_x^2 representa a variância do sinal de entrada. Essa aproximação leva a resultados satisfatórios somente quando o sinal de entrada é branco e o número de coeficientes é grande (tipicamente, considera-se $M > 30$).

Em [66], o princípio da média⁴ (PM), definido em [67], é utilizado em conjunto com a hipótese da independência, permitindo o

³ Um valor esperado dado por $E[\alpha(n)/\beta(n)]$ pode ser aproximado por $E[\alpha(n)]/E[\beta(n)]$, visto que, geralmente, $\alpha(n)$ e $\beta(n)$ representam estatísticas de segunda ordem do sinal de entrada e, dessa forma, tendem a variar lentamente em função de n [8].

⁴ Sejam $\alpha(n)$ e $\beta(n)$ dois processos conjuntamente estacionários, com $\beta(n)$ variando lentamente em relação a $\alpha(n)$, então, pode-se assumir que as variáveis aleatórias são quase independentes. Assim, considerando que as funções $f(\cdot)$ e $g(\cdot)$ são funções contínuas e limitadas, tem-se $E\{f[\alpha(n)]g[\beta(n)]\} \cong E\{f[\alpha(n)]\}E\{g[\beta(n)]\}$ [67].

uso da função densidade de probabilidade chi-quadrada⁵ para representar $\mathbf{x}^H(n)\mathbf{x}(n)$, o que resulta em

$$E\left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)}\right] \cong E\left[\frac{1}{\mathbf{x}^H(n)\mathbf{x}(n)}\right]E[\mathbf{x}(n)\mathbf{x}^H(n)] = \frac{\mathbf{R}}{(M-2)\sigma_x^2}. \quad (2.4)$$

Contudo, apesar de essa estratégia simplificar notavelmente a matemática envolvida, ela é apropriada somente para sinais de entrada não-correlacionados e filtros de ordem elevada, gerando, assim, um modelo bastante limitado [16], [19].

Em [61], Barrault *et al.* têm utilizado uma abordagem baseada em equações diferenciais estocásticas (*stochastic differential equation* - SDE) para derivar o modelo do algoritmo NLMS. Tal abordagem foi inicialmente desenvolvida por Kushner [52] e introduzida na análise de algoritmos recursivos em [68]. Entretanto, o uso dessa estratégia limita a utilização do modelo a passos de adaptação infinitesimais [52], requerendo também o cálculo dos referidos valores esperados normalizados. Para tal, em [61], foram também utilizadas aproximações similares a (2.4). Assim, conclui-se que o modelo resultante é de pouca serventia em casos práticos, pois foi derivado para condições de operação muito particulares.

Visando à obtenção de um modelo mais realista para sinais gaussianos reais correlacionados, Lobato *et al.* [16] empregaram o PM associado à técnica proposta em [14] no cálculo de \mathbf{R}^N . Entretanto, como em [14], [60] e [62], tal estratégia recai na solução de integrais hiperelípticas de alta ordem. Nesse ponto, visando à obtenção de uma solução analítica, [16] propõe uma aproximação, a qual foi posteriormente aprimorada em [15] e [22], com o objetivo de eliminar o operador raiz quadrada presente no denominador da integral hiperelíptica. Dessa forma, permitindo a obtenção de soluções analíticas para as referidas integrais. O modelo resultante é mais preciso do que outros que também possuem solução analítica, porém ainda restrito as condições de aplicação do PM. Posteriormente, uma estratégia semelhante é utilizada por Kolodziej *et al.* [17], [18], [21] para obter uma formulação fechada para o trabalho de Rupp [62]. Nesses trabalhos

⁵ Uma variável aleatória ψ tem distribuição de probabilidade chi-quadrada χ_K^2 com K graus de liberdade se $\psi = \sum_{k=1}^K |\phi(k)|^2$, onde $\phi(k)$ é branco e tem distribuição gaussiana e variância unitária [11].

[17], [18], [21], os momentos normalizados são obtidos sem o uso do PM, resultando em um modelo ainda mais preciso, mas limitado a dados reais obtidos através de um processo aleatório SIRP. Aqui, cabe ressaltar que, como em [62], o modelo obtido também requer a solução de (2.2) que, exceto para dados gaussianos, não possui uma solução fechada.

Recentemente, Chan e Zhou em 2009 [69] propuseram um modelo estocástico para o algoritmo ε -NLMS operando com sinal de entrada gaussiano real correlacionado. Tal estudo ainda se baseia na técnica desenvolvida em [14] e [60]. Portanto, devido à presença do parâmetro de regularização, sua solução para os momentos normalizados também fica vinculada ao uso de técnicas de integração numérica [63].

Para o caso de sinais de entrada complexos e correlacionados muito pouco tem sido desenvolvido para o algoritmo NLMS em termos de modelagem, podendo-se citar apenas dois recentes trabalhos de Al-Naffouri *et al.* [70] e [71]. Entretanto, os resultados apresentados em tais trabalhos são no mínimo questionáveis (duvidosos). Por exemplo, em [70], os referidos autores afirmam ter uma solução fechada para calcular os requeridos momentos normalizados; contudo, devido à presença da integral exponencial generalizada, referente ao parâmetro de regularização, tal afirmação não procede. Referente à [71], é possível mostrar que a expressão fornecida para o cálculo de \mathbf{R}^N diverge quando utilizada em filtros de ordem superior a dois (veja Apêndice D).

Portanto, baseado nos trabalhos discutidos até o presente momento, pode-se considerar, em nosso entendimento, que a modelagem do algoritmo NLMS operando com dados gaussianos complexos, correlacionados ou não, permanece ainda um problema em aberto. Nesse contexto, o presente trabalho visa à obtenção de um modelo estocástico para o algoritmo NLMS que possua solução analítica, opere tanto com dados complexos quanto reais e apresente maior precisão do que os modelos até então disponíveis na literatura.

2.4 Modelo proposto

Nesta seção, são derivadas expressões que descrevem o comportamento médio do vetor de coeficientes, a curva de aprendizagem e a matriz de covariância do vetor de erro nos coeficientes. Ao mesmo tempo, são estabelecidas as hipóteses simplificativas utilizadas durante o processo de modelagem do algoritmo NLMS.

2.4.1 Comportamento médio do vetor de coeficientes

Considerando um problema de identificação de sistema (veja Figura 1.4), a partir de (1.4), pode-se expressar o sinal de erro como

$$e(n) = \mathbf{w}_o^H \mathbf{x}(n) - \mathbf{w}^H(n) \mathbf{x}(n) + z(n) \quad (2.5)$$

e, seu complexo-conjugado, por

$$e^*(n) = \mathbf{x}^H(n) \mathbf{w}_o - \mathbf{x}^H(n) \mathbf{w}(n) + z^*(n) \quad (2.6)$$

onde $z(n)$ representa o ruído de medição e \mathbf{w}_o corresponde ao vetor de coeficientes ótimo que, por questão de simplicidade, tem a mesma dimensão de $\mathbf{w}(n)$.

Substituindo (2.6) em (1.18) e tomando o valor esperado de ambos os lados da expressão resultante, obtém-se

$$E[\mathbf{w}(n+1)] = E[\mathbf{w}(n)] + \underbrace{\mu E \left[\frac{\mathbf{x}(n) \mathbf{x}^H(n)}{\varepsilon + \mathbf{x}^H(n) \mathbf{x}(n)} \right]}_A \mathbf{w}_o - \underbrace{\mu E \left[\frac{\mathbf{x}(n) \mathbf{x}^H(n) \mathbf{w}(n)}{\varepsilon + \mathbf{x}^H(n) \mathbf{x}(n)} \right]}_B + \underbrace{\mu E \left[\frac{\mathbf{x}(n) z^*(n)}{\varepsilon + \mathbf{x}^H(n) \mathbf{x}(n)} \right]}_C. \quad (2.7)$$

Para determinar os valores esperados dados pelos termos A, B e C em (2.7) é necessário, inicialmente, estabelecer as seguintes hipóteses simplificativas:

H1) Assume-se que ε é uma constante positiva de valor muito pequeno [2], [10], [11], [54] se comparada à potência do sinal de entrada, não tendo efeito significativo no comportamento médio do algoritmo [55], [56], [58], [60], podendo então ser desprezada. Vale ressaltar que a presença do parâmetro de regularização leva a uma integral que não tem solução analítica [60], [69], inviabilizando assim a obtenção de um modelo com solução fechada.

H2) O ruído de medição $z(n)$ é independente de qualquer outro sinal envolvido no processo, tendo média nula e variância σ_z^2 [10], [11]. Tal hipótese é comumente utilizada na modelagem de algoritmos adaptativos [1], [2], [11], [30], [56].

H3) Considerando a condição de adaptação lenta (passo de adaptação pequeno), pode-se supor que o vetor de coeficientes

$\mathbf{w}(n)$ é estatisticamente independente do vetor do sinal de entrada $\mathbf{x}(n)$ [1], [9], [11], [38]. Tal hipótese é conhecida na literatura por Teoria da Independência⁶ (TI) [1], [10], [11] e, na maioria das situações, não é válida quando uma estrutura transversal de filtragem é utilizada, uma vez que $M-1$ elementos da memória do filtro são idênticos em iterações consecutivas [10], [11], [52], [72]. Contudo, apesar de esforços terem sido realizados na obtenção de modelos que não utilizem a TI [56], [72]-[74], nenhuma solução viável tem sido ainda encontrada. Assim, visando tornar a derivação matematicamente tratável, considera-se aqui o uso da TI como uma hipótese admissível.

Assim, utilizando a Hipótese H1) no termo A de (2.7), obtém-se

$$\mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \cong \mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] = \mathbf{R}^N \quad (2.8)$$

onde \mathbf{R}^N é a matriz de autocorrelação normalizada [62]. Através da Hipótese H2) é possível mostrar que o termo C de (2.7) é nulo, isto é,

$$\mathbb{E} \left[\frac{\mathbf{x}(n)z^*(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] = \mathbb{E} \left[\frac{\mathbf{x}(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \mathbb{E}[z^*(n)] = 0 \quad (2.9)$$

decorrente do fato de que $z(n)$ é independente do sinal de entrada e tem média nula [11]. Para determinar o termo B de (2.7) é utilizada a Hipótese H3), resultando em

$$\mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \cong \mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \mathbb{E}[\mathbf{w}(n)] \quad (2.10)$$

onde, empregando a Hipótese H1) ou equivalentemente o resultado obtido em (2.8), tem-se

$$\mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \cong \mathbf{R}^N \mathbb{E}[\mathbf{w}(n)]. \quad (2.11)$$

Por fim, com base nos resultados (2.8), (2.9) e (2.11), pode-se reescrever (2.7) como

$$\mathbb{E}[\mathbf{w}(n+1)] = \mathbb{E}[\mathbf{w}(n)] - \mu \mathbf{R}^N \mathbb{E}[\mathbf{w}(n)] + \mu \mathbf{R}^N \mathbf{w}_o. \quad (2.12)$$

⁶ A Teoria da Independência foi originalmente proposta por Senne em 1968, mas se popularizou através da análise estatística do algoritmo LMS [38].

Dessa forma, para uma dada condição inicial, o comportamento médio do vetor de coeficientes é completamente caracterizado se a matriz de autocorrelação normalizada e o vetor de coeficientes ótimo \mathbf{w}_0 forem conhecidos.

O procedimento para determinar \mathbf{R}^N é bastante complicado, pois envolve o cálculo de uma integral de alta ordem (caso complexo) ou uma integral hiperelíptica (caso real) [15]-[18], [21]. Além disso, sabendo que a função densidade de probabilidade gaussiana é definida de forma diferente para dados reais ou complexos [25], duas soluções distintas para \mathbf{R}^N são obtidas. Tal característica justifica o motivo de os modelos de algoritmos adaptativos serem tratados separadamente dependendo do tipo do sinal de entrada. Em outras palavras, a diferença em (2.12) frente ao tipo de sinal de entrada (complexo ou real) está na forma como a matriz de autocorrelação normalizada é determinada. Nesse contexto, o presente trabalho busca generalizar [14] apresentando um procedimento exato para o cálculo da matriz de autocorrelação normalizada para o caso de dados complexos e aproximado, porém bastante preciso, para o caso de dados reais. Vale ressaltar que, para este último cenário, a solução aqui descrita pode ser vista como um caso particular de [18] e [21], e tem como objetivo tornar este documento auto-contido no que se refere à modelagem do algoritmo NLMS para sinais de entrada gaussianos.

Com vistas aos resultados apresentados no Apêndice A, a matriz de autocorrelação normalizada pode ser expressa como

$$\mathbf{R}^N = \mathbf{Q}\mathbf{H}\mathbf{Q}^H \quad (2.13)$$

onde \mathbf{Q} é a matriz de autovetores oriunda da decomposição

$$\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^H \quad (2.14)$$

e \mathbf{H} é uma matriz diagonal com dimensão $M \times M$ cujos elementos são determinados em função das características do sinal de entrada. Particularmente, para o caso de sinais de entrada complexos e correlacionados, os elementos de \mathbf{H} são obtidos através de

$$h_{i,i}^{\mathbb{C}} = -\frac{1}{G^{\mathbb{C}}} \left[\frac{A_{1,i}^{\mathbb{C}}}{\omega_i^{\mathbb{C}}} + \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^{\mathbb{C}} \log \left(\frac{\omega_l^{\mathbb{C}}}{\omega_i^{\mathbb{C}}} \right) \right] \quad (2.15)$$

com (\mathbb{C} denotando sinal de entrada complexo) $\omega_k^{\mathbb{C}}$, $G^{\mathbb{C}}$, $A_{1,i}^{\mathbb{C}}$ e $B_{l,i}^{\mathbb{C}}$ dados, respectivamente, por (A.13), (A.14), (A.16) e (A.17). Entretanto,

quando o vetor de entrada $\mathbf{x}(n)$ é correlacionado e composto por variáveis aleatórias reais, os elementos de \mathbf{H} (denotados $h_{i,i}^{\mathbb{R}}$) devem ser determinados através de (A.32). Finalmente, para o caso de sinais de entrada não-correlacionados (complexos ou reais), \mathbf{H} pode ser reescrita como

$$\mathbf{H} = \frac{\mathbf{I}}{M}. \quad (2.16)$$

Note que, o resultado proposto para o caso de sinais de entrada não-correlacionados é muito mais simples do que as soluções apresentadas em [57], [61] e [66], e está em linha com [14] e [65]. Portanto, dependendo das características do sinal de entrada, uma das soluções analíticas apresentadas para o cálculo da matriz de autocorrelação normalizada pode ser utilizada [veja (2.15), (A.32) e (2.16)]. Assim, para uma dada condição inicial, conhecida a solução de \mathbf{R}^N , o comportamento médio do vetor de coeficientes pode ser determinado através de (2.12).

2.4.2 Curva de aprendizagem

A curva de aprendizagem descreve a evolução do erro quadrático médio de um dado algoritmo adaptativo. Essa métrica fornece informações referentes ao comportamento do algoritmo em regime transitório e em regime permanente, sendo comumente adotada como medida de desempenho na comparação de algoritmos adaptativos. Conforme definido em (1.7), o EQM pode ser reescrito como [11]

$$J(n) = E[e(n)e^*(n)] \quad (2.17)$$

sendo essa expressão designada curva de aprendizado devido à sua dependência em n [2], [9]-[11]. Para determiná-la, inicialmente, define-se o vetor de erro nos coeficientes como [38]

$$\mathbf{v}(n) = \mathbf{w}(n) - \mathbf{w}_0. \quad (2.18)$$

Logo, usando (2.18) em (2.5) e (2.6), tem-se

$$e(n) = -\mathbf{v}^H(n)\mathbf{x}(n) + z(n) \quad (2.19)$$

e

$$e^*(n) = -\mathbf{x}^H(n)\mathbf{v}(n) + z^*(n). \quad (2.20)$$

Determinando o produto $e(n)e^*(n)$, tomando o valor esperado de ambos os lados da expressão resultante e utilizando (2.17), obtém-se

$$J(n) = E\{-\mathbf{v}^H(n)\mathbf{x}(n) + z(n)\}[-\mathbf{x}^H(n)\mathbf{v}(n) + z^*(n)]. \quad (2.21)$$

Manipulando (2.21), considerando as características de $z(n)$ (veja Hipótese H2), tem-se

$$J(n) = E[z(n)z^*(n)] + E[\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)]. \quad (2.22)$$

A partir do segundo termo do lado direito de (2.22), dado que $E[\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)]$ é um escalar, utilizando as propriedades do traço, a Hipótese H3 (veja Seção 2.4.1) e (2.14), obtém-se

$$E[\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)] = \text{tr}[\Lambda \mathbf{Q}^H \mathbf{K}(n) \mathbf{Q}] \quad (2.23)$$

onde $\text{tr}(\cdot)$ denota o operador traço de uma matriz, e

$$\mathbf{K}(n) = E[\mathbf{v}(n)\mathbf{v}^H(n)] \quad (2.24)$$

a matriz de covariância do vetor de erro nos coeficientes. Agora, definindo a matriz de covariância do vetor de erro nos coeficientes (com os eixos transformados) como

$$\mathbf{K}'(n) = \mathbf{Q}^H E[\mathbf{v}(n)\mathbf{v}^H(n)] \mathbf{Q} \quad (2.25)$$

e usando então o fato de que Λ é uma matriz diagonal, tem-se [1], [2], [30], [75]

$$E[\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)] = \boldsymbol{\lambda}^T \mathbf{k}'(n) \quad (2.26)$$

onde $\boldsymbol{\lambda}$ é um vetor contendo os autovalores de \mathbf{R} e $\mathbf{k}'(n)$, um vetor com os elementos da diagonal de $\mathbf{K}'(n)$. Note que, como a matriz \mathbf{R} é hermitiana⁷, os elementos de $\boldsymbol{\lambda}$ são reais [2], [10], [26], [33]. Além disso, como $\mathbf{K}(n)$ também é uma matriz hermitiana e $\mathbf{K}'(n)$ é obtida através de uma transformação de congruência⁸, pode-se garantir que os elementos de $\mathbf{k}'(n)$ também são reais [32], [33].

Observa-se que a primeira parcela do lado direito de (2.22) é função apenas da variância do ruído de medição. Tal termo representa o EQM mínimo da função custo e é obtido quando $\mathbf{w}(n)$ assume o valor de \mathbf{w}_0 . Assim,

$$J_{\min} = E[z(n)z^*(n)] = \sigma_z^2. \quad (2.27)$$

A partir de (2.26), a segunda parcela do lado direito de (2.22) pode ser reescrita como

⁷ Uma matriz Φ é dita hermitiana quando $\Phi = \Phi^H$ [11], [33].

⁸ Transformações de congruência preservam a forma hermitiana de uma matriz [32].

$$J_{\text{ex}}(n) = \lambda^T \mathbf{k}'(n). \quad (2.28)$$

Esse termo representa o EQM em excesso introduzido pelo algoritmo adaptativo devido à flutuação dos coeficientes em relação ao ponto ótimo [2], [9]-[11], [38]. Logo, através de (2.27) e (2.28), pode-se expressar a curva de aprendizagem do algoritmo NLMS como

$$J(n) = J_{\text{min}} + J_{\text{ex}}(n) \quad (2.29)$$

cuja forma é comumente obtida na modelagem de algoritmos adaptativos da família LMS [1], [2], [9]-[11], [30], [75].

Portanto, a curva de aprendizagem do algoritmo NLMS é completamente caracterizada se a potência do ruído de medição, os autovalores da matriz de autocorrelação do sinal de entrada e o vetor $\mathbf{k}'(n)$, a cada instante n , são conhecidos. Nesse ponto, cabe salientar que, garantindo a convergência dos elementos da diagonal da matriz de covariância do vetor de erro nos coeficientes, garante-se, por consequência, que o algoritmo NLMS converge [1], [2], [30].

2.4.3 Matriz de covariância do vetor de erro nos coeficientes

Para determinar uma recursão para $\mathbf{k}'(n)$, subtrai-se \mathbf{w}_o de ambos os lados de (1.18), substitui-se (2.20) na expressão resultante, determina-se o produto externo de $\mathbf{v}(n+1)$ [isto é, $\mathbf{v}(n+1)\mathbf{v}^H(n+1)$], toma-se o valor esperado de ambos os lados da expressão resultante, e levando-se em conta que os termos que contém $z(n)$ são nulos devido à natureza do ruído de medição (veja Hipótese H2), obtém-se

$$\begin{aligned} E[\mathbf{v}(n+1)\mathbf{v}^H(n+1)] &= E[\mathbf{v}(n)\mathbf{v}^H(n)] - \mu E \left[\frac{\mathbf{v}(n)\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \\ &\quad - \mu E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)\mathbf{v}^H(n)}{\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)} \right] \\ &\quad + \mu^2 E \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n) |z(n)|^2}{[\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\} \\ &\quad + \mu^2 E \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)}{[\varepsilon + \mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\}. \end{aligned} \quad (2.30)$$

Agora, fazendo uso das Hipóteses H1)-H3) (veja Seção 2.4.1), de (2.24) e (2.27), os requeridos valores esperados podem ser reescritos como

$$\begin{aligned}
 \mathbf{K}(n+1) &= \mathbf{K}(n) - \mu \mathbf{K}(n) \mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] \\
 &\quad - \mu \mathbb{E} \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] \mathbf{K}(n) \\
 &\quad + \underbrace{\mu^2 \mathbb{E} \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n)}{[\mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\}}_D J_{\min} \\
 &\quad + \underbrace{\mu^2 \mathbb{E} \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{v}(n)\mathbf{v}^H(n)\mathbf{x}(n)\mathbf{x}^H(n)}{[\mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\}}_E.
 \end{aligned} \tag{2.31}$$

Assumindo que o sinal de entrada é gaussiano, o último termo do lado direito de (2.31) representa um momento de quarta ordem⁹ (normalizado e ponderado). A solução desse termo para sinais de entrada reais ou complexos difere também de uma constante [veja (2.33)]; dessa forma, a solução aqui proposta leva em consideração tal diferença.

Diante dos resultados apresentados nos Apêndices A-C, pré- e pós-multiplicando (2.31) por \mathbf{Q}^H e \mathbf{Q} , respectivamente, e tomando os elementos da diagonal principal da expressão resultante, obtém-se a seguinte recursão¹⁰ para $\mathbf{k}'(n)$ [17], [18], [21], [61], [62]:

$$\mathbf{k}'(n+1) = \mathbf{B}\mathbf{k}'(n) + g\mathbf{s} \tag{2.32}$$

com

$$\mathbf{B} = \mathbf{I} - 2\mu\mathbf{H} + \mu^2(\zeta\mathbf{T} + \mathbf{P}) = \begin{cases} 1 - 2\mu h_{i,i} + (\zeta + 1)\mu^2 t_{i,i} & i = j \\ \mu^2 p_{i,j} & i \neq j \end{cases} \tag{2.33}$$

e

$$g = \mu^2 J_{\min}. \tag{2.34}$$

⁹ A fatoração de um momento de quarta ordem de uma variável gaussiana é dada por: $\mathbb{E}[\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{K}(n)\mathbf{x}(n)\mathbf{x}^H(n)] = \zeta\mathbf{R}\mathbf{K}(n)\mathbf{R} + \mathbf{R}\text{tr}[\mathbf{R}\mathbf{K}(n)]$, com $\zeta = 1$ para dados complexos e $\zeta = 2$ para dados reais [2], [10], [25], [26].

¹⁰ Esta abordagem de transformar a recursão de $\mathbf{K}(n)$ em uma equação de diferenças vetorial $\mathbf{k}'(n)$ foi apresentada na literatura por Fisher em 1983 [75], sendo largamente utilizada por diversos autores [2], [14], [17], [18], [21], [52], [60]-[62], [65].

As matrizes \mathbf{H} , \mathbf{T} e \mathbf{P} , e o vetor \mathbf{s} são determinados de forma diferente dependendo do tipo do sinal de entrada (real ou complexo e correlacionado ou não-correlacionado). Além disso, a constante ζ também é modificada levando em consideração tal característica, ou seja, para dados complexos faz-se $\zeta = 1$ e para dados reais $\zeta = 2$.

A partir de (2.32), observa-se que a presença do termo $\mathbf{g}\mathbf{s}$ implica que a recursão para $\mathbf{k}'(n)$ nunca convergirá para zero, isto é, os coeficientes do filtro adaptativo estarão sempre flutuando em torno de seus valores ótimos [1], [2], [9]-[11], [14], [38], [75]-[78]. Tal comportamento é esperado quando algoritmos adaptativos são utilizados em ambientes ruidosos. Entretanto, quando a solução aproximada $\tilde{\mathbf{k}}(n) \equiv \text{diag}\{\mathbf{Q}^H \mathbf{E}[\mathbf{v}(n)] \mathbf{E}[\mathbf{v}^H(n)] \mathbf{Q}\}$ é adotada, $\tilde{\mathbf{k}}(n)$ converge para zero contrariando o comportamento efetivo do algoritmo (veja Apêndice E). Essa aproximação vem sendo considerada na modelagem de outros algoritmos adaptativos [19], [20], [23], [79], [80], porém, mesmo para adaptação lenta, não se consegue capturar completamente o comportamento do algoritmo, principalmente, no que tange a solução em regime permanente, devendo tal aproximação ser então evitada.

Uma condição inicial para $\mathbf{k}'(n)$ pode ser obtida substituindo (2.18) em (2.25), fazendo $n=0$ na expressão resultante e assumindo que o vetor de coeficientes do algoritmo adaptativo é inicializado com zero [isto é, $\mathbf{w}(0) = \mathbf{0}$]. Assim,

$$\mathbf{k}'(0) = \text{diag}(\mathbf{Q}^H \mathbf{w}_o \mathbf{w}_o^H \mathbf{Q}) \quad (2.35)$$

onde $\text{diag}(\cdot)$ denota o operador diagonal [33]. Dessa forma, para uma dada condição inicial $\mathbf{k}'(0)$, a evolução de $\mathbf{k}'(n)$ pode ser predita se as matrizes \mathbf{H} , \mathbf{T} e \mathbf{P} , e o vetor \mathbf{s} forem conhecidos. Nesse contexto, o formato adotado para a recursão da matriz de covariância do vetor de erro nos coeficientes [representado pela equação de diferenças vetorial (2.32)] apresenta certa flexibilidade, pois permite acomodar a maioria dos modelos estocásticos do algoritmo NLMS encontrados na literatura [14]-[18], [21], [52], [57], [60]-[62], [65], [66], bastando apenas modificar a forma como \mathbf{H} , \mathbf{T} , \mathbf{P} e \mathbf{s} são calculados.

No modelo proposto, a matriz \mathbf{H} é determinada de forma idêntica à apresentada na derivação do comportamento médio do vetor de coeficientes (veja Seção 2.4.1). Enquanto, para o caso de sinais de entrada complexos e correlacionados, os elementos da matriz diagonal \mathbf{T} e da diagonal de \mathbf{P} são dados por (veja Apêndice B)

$$t_{i,i}^C = p_{i,i}^C = \frac{1}{G^C} \left(-\frac{A_{1,i}^C}{2\omega_i^C} + \sum_{\substack{l=1 \\ l \neq i}}^M \frac{B_{l,i}^C}{(\omega_l^C - \omega_i^C)} \left\{ \omega_l^C \left[1 + \log \left(\frac{\omega_l^C}{\omega_i^C} \right) \right] - \omega_i^C \right\} \right) \quad (2.36)$$

os elementos fora da diagonal da matriz simétrica \mathbf{P} por

$$p_{i,j}^C = \frac{1}{G^C} \left(\frac{A_{1,i}^C}{\omega_j^C - \omega_i^C} \left\{ \log(-\omega_i^C) + \frac{2\omega_j^C [1 - \log(-\omega_j^C)]}{\omega_j^C - \omega_i^C} \right\} + B_{j,i}^C \left\{ \frac{2\omega_i^C [1 - \log(-\omega_i^C)]}{\omega_j^C - \omega_i^C} - \log(-\omega_j^C) \right\} + \sum_{\substack{l=1 \\ l \neq i,j}}^M B_{l,i}^C \left\{ \frac{\omega_l^C [1 - \log(-\omega_l^C)]}{(\omega_l^C - \omega_j^C)} + \frac{(\omega_l^C + \omega_j^C - 2\omega_i^C)\omega_i^C [1 - \log(-\omega_i^C)]}{(\omega_j^C - \omega_i^C)^2} + \frac{(\omega_l^C - \omega_i^C)(\omega_l^C + \omega_i^C - 2\omega_j^C)\omega_j^C [1 - \log(-\omega_j^C)]}{(\omega_j^C - \omega_i^C)^2 (\omega_l^C - \omega_j^C)} \right\} \right) \quad (2.37)$$

e os elementos de \mathbf{s} por (veja Apêndice C)

$$s_i^C = \frac{-1}{G^C} \left(A_{1,i}^C \log(-\omega_i^C) - \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^C \{ \omega_l^C [1 - \log(-\omega_l^C)] - \omega_i^C [1 - \log(-\omega_i^C)] \} \right) \quad (2.38)$$

com ω_k^C , G^C , $A_{1,i}^C$ e $B_{l,i}^C$ obtidos, respectivamente, a partir de (A.13), (A.14), (A.16) e (A.17). Entretanto, para o caso de sinais de entrada reais e correlacionados, os elementos de \mathbf{T} , \mathbf{P} e \mathbf{s} devem ser determinados através de (B.24), (B.25) e (C.14), respectivamente. Finalmente, para o caso de sinais de entrada não-correlacionados, as requeridas soluções podem ser simplificadas para (veja Apêndices B-C)

$$\mathbf{T} = \frac{\mathbf{I}}{M(M + \zeta)}, \quad (2.39)$$

$$\mathbf{P} = \frac{\mathbf{1}\mathbf{1}^T}{M(M + \zeta)} \quad (2.40)$$

e

$$\mathbf{s} = \frac{\mathbf{1}}{M(M - \zeta)\sigma_x^2}. \quad (2.41)$$

onde $\mathbf{1}$ denota um vetor com elementos unitários de dimensão $M \times 1$. Portanto, através de (2.29) e (2.32), conhecidas as características do sinal de entrada e a condição inicial para $\mathbf{k}'(n)$ [dada por (2.35)], é possível prever o comportamento dos elementos da diagonal da matriz de covariância do vetor de erro nos coeficientes e a curva de aprendizagem do algoritmo NLMS. Dessa forma, torna-se possível estabelecer algumas relações entre as métricas de desempenho e os parâmetros que controlam o algoritmo, com objetivo de adequá-lo da melhor forma possível ao sistema adaptativo sob projeto.

2.5 Análise do modelo proposto

Nesta seção, são derivadas expressões que relacionam os parâmetros do algoritmo NLMS a algumas métricas de desempenho utilizadas na análise de algoritmos adaptativos, possibilitando: melhorar a compreensão de como tais parâmetros afetam o comportamento do algoritmo; comparar os algoritmos LMS e NLMS através da velocidade de convergência e/ou erro em regime permanente; e explicar o efeito da normalização na velocidade de convergência do algoritmo NLMS.

2.5.1 EQM em excesso em regime permanente

Utilizando a definição do EQM em excesso [dada por (2.28)] e fazendo $n \rightarrow \infty$ em ambos os lados dessa expressão, tem-se [1], [10], [11], [30]

$$J_{\text{ex}}(\infty) = \lambda^T \mathbf{k}'(\infty) \quad (2.42)$$

evidenciando que o EQM em excesso em regime permanente depende diretamente dos elementos de $\mathbf{k}'(\infty)$, implicando que um estudo mais aprofundado de (2.32) será necessário.

Sabe-se que a solução para a equação de diferenças vetorial de $\mathbf{k}'(n)$ é dada por

$$\mathbf{k}'(n) = \mathbf{B}^n \mathbf{k}'(0) + g \sum_{m=0}^{n-1} \mathbf{B}^m \mathbf{s} \quad (2.43)$$

podendo ser encontrada iterativamente através de (2.32) [2], [14], [60], [61], [75]. Logo, fazendo $n \rightarrow \infty$ e assumindo convergência (garantida quando todos os autovalores de \mathbf{B} têm módulo menor do que a unidade), o termo função de $\mathbf{k}'(0)$ tende para zero. Assim, a partir do

segundo termo do lado direito de (2.43), dado que [33]

$$\sum_{m=0}^{\infty} \mathbf{B}^m = (\mathbf{I} - \mathbf{B})^{-1} \quad (2.44)$$

para $|\lambda(\mathbf{B})| < 1$, obtém-se¹¹ [1], [2], [30], [33], [75], [81]

$$\mathbf{k}'(\infty) = g(\mathbf{I} - \mathbf{B})^{-1} \mathbf{s}. \quad (2.45)$$

Note que, para $g \rightarrow 0$ (equivalentemente, $\mu \rightarrow 0$ ou $J_{\min} \rightarrow 0$) o EQM em excesso em regime permanente introduzido pelo algoritmo adaptativo é desprezível, ou seja, nessa condição as flutuações em torno do ponto ótimo são insignificantes.

Agora, utilizando a relação [2]

$$\sum_{m=0}^{n-1} \mathbf{B}^m = (\mathbf{I} - \mathbf{B})^{-1} - \mathbf{B}^n (\mathbf{I} - \mathbf{B})^{-1} \quad (2.46)$$

(2.43) pode ser alternativamente reescrita como [2], [61], [75]

$$\mathbf{k}'(n) = \mathbf{B}^n [\mathbf{k}'(0) - \mathbf{k}'(\infty)] + \mathbf{k}'(\infty). \quad (2.47)$$

A parcela $\mathbf{B}^n [\mathbf{k}'(0) - \mathbf{k}'(\infty)]$ é responsável (principalmente) pelo comportamento do algoritmo em regime transitório, tendendo a zero quando $n \rightarrow \infty$. Por sua vez, o termo $\mathbf{k}'(\infty)$ representa a solução em regime permanente e não está presente quando utilizamos a solução aproximada dada por $\tilde{\mathbf{k}}(\infty)$ (veja Apêndice E).

Analisando (2.45) e (2.47), constata-se que \mathbf{B} e \mathbf{s} afetam o comportamento do modelo da curva de aprendizagem tanto em regime transitório quanto em regime permanente. Dessa forma, quanto mais fiéis (precisas) forem suas estimativas em relação ao desempenho efetivo do algoritmo, melhor será a predição obtida através do modelo estocástico. Para ilustrar tal comentário, considere que as aproximações utilizadas durante o processo de modelagem não reflitam o comportamento efetivo do algoritmo, seja por modificar de maneira inconsistente a magnitude dos elementos de \mathbf{B} ou de \mathbf{s} . Então, é de se esperar que a convergência do modelo seja mais rápida ou mais lenta com respeito à simulação, assim resultando em um descasamento entre o comportamento teórico e prático e, por consequência, em um modelo

¹¹ Esse mesmo resultado pode ser obtido fazendo $n \rightarrow \infty$ em ambos os lados de (2.32), assumindo convergência e notando que $\lim_{n \rightarrow \infty} \mathbf{k}'(n+1) = \lim_{n \rightarrow \infty} \mathbf{k}'(n) = \mathbf{k}'(\infty)$ [1], [24], [30], [57], [61].

impreciso (veja [16], [57], [61], [66]). Nesse quesito, devido ao reduzido número de hipóteses simplificadoras utilizadas, acredita-se que o modelo proposto consiga capturar com maior precisão o comportamento do algoritmo em questão.

A partir de (2.45), observa-se que a solução de $\mathbf{k}'(\infty)$ envolve o cálculo da inversa de uma matriz. Entretanto, para o caso mais geral (sinais de entrada correlacionados), tal inversa não possui solução analítica. Assim, com o objetivo de evitar o uso de técnicas numéricas, as quais tornam a solução dispendiosa, e visando a obtenção de uma formulação fechada, uma solução aproximada para tal inversa é aqui proposta. Para tal, inicialmente, considere a seguinte relação [33]:

$$(\mathbf{M} - \mathbf{N})^{-1} = \mathbf{M}^{-1}[\mathbf{I} + \mathbf{N}(\mathbf{M}^{-1}\mathbf{N})^{-1}\mathbf{M}^{-1}] \quad (2.48)$$

como um caso particular do lema de inversão de matrizes [11], [26], [32], [33]. Agora, denotando $\mathbf{N} = \varphi\mathbf{N}'$, (2.48) pode ser reescrita como

$$(\mathbf{M} - \varphi\mathbf{N}')^{-1} = \mathbf{M}^{-1}[\mathbf{I} + \varphi\mathbf{N}'(\mathbf{M}^{-1}\mathbf{N}')^{-1}\mathbf{M}^{-1}]. \quad (2.49)$$

Então, assumindo que $\varphi \ll 1$, tem-se

$$(\mathbf{M} - \varphi\mathbf{N}')^{-1} \cong \mathbf{M}^{-1}(\mathbf{I} + \varphi\mathbf{N}'\mathbf{M}^{-1}) \quad (2.50)$$

como solução aproximada para a inversa de (2.45).

Do mesmo modo, substituindo (2.33) e (2.34) em (2.45), e considerando passo de adaptação pequeno [tendo em vista (2.50)], a solução de $\mathbf{k}'(\infty)$ pode ser aproximada para

$$\mathbf{k}'(\infty) \cong \frac{\mu}{2} J_{\min} \mathbf{A}^{-1} \left(\mathbf{I} + \frac{\mu}{2} \mathbf{P} \mathbf{A}^{-1} \right) \mathbf{s} \quad (2.51)$$

onde \mathbf{A} é uma matriz diagonal definida como

$$\mathbf{A} = \left(\mathbf{H} - \frac{\mu\zeta}{2} \mathbf{T} \right). \quad (2.52)$$

Por fim, substituindo (2.51) em (2.42), o EQM em excesso em regime permanente é obtido de forma analítica através de

$$J_{\text{ex}}(\infty) \cong \frac{\mu}{2} J_{\min} \boldsymbol{\lambda}^T \mathbf{A}^{-1} \left(\mathbf{I} + \frac{\mu}{2} \mathbf{P} \mathbf{A}^{-1} \right) \mathbf{s}. \quad (2.53)$$

É importante enfatizar que a expressão (2.53) é válida tanto para sinais de entrada correlacionados quanto não-correlacionados. Porém, para este último caso, uma solução para a inversa presente em (2.45) pode ser obtida de forma exata. Dessa forma, recomenda-se o uso de (2.53) apenas para o caso de sinais de entrada correlacionados.

Considerando que o sinal de entrada é não-correlacionado, substituindo (2.16), (2.39) e (2.40) em (2.33), a matriz \mathbf{B} pode ser reescrita como

$$\mathbf{B} = \left[\mathbf{I} - \frac{2\mu}{M} \mathbf{I} + \frac{\mu^2}{M(M+\zeta)} (\zeta \mathbf{I} + \mathbf{1}\mathbf{1}^T) \right]. \quad (2.54)$$

Então, utilizando (2.41), (2.54) e a identidade de Sherman-Morrison¹² para obter a matriz inversa, (2.45) resulta em

$$\mathbf{k}'(\infty) = \left[\frac{\mu}{(2-\mu)} \frac{1}{(M-\zeta)} \frac{J_{\min}}{\sigma_x^2} \right] \mathbf{1}. \quad (2.55)$$

Finalmente, substituindo (2.55) em (2.42) e escrevendo $\lambda = \sigma_x^2 \mathbf{1}$, obtém-se

$$J_{\text{ex}}(\infty) = \frac{\mu}{(2-\mu)} \frac{M}{(M-\zeta)} J_{\min}. \quad (2.56)$$

Portanto, considerando que o sinal de entrada seja não-correlacionado, (2.56) determina o valor do EQM em excesso em regime permanente. Observe que a solução aqui apresentada, especificamente para dados de entrada reais, está alinhada com [10], [52] e [65].

O resultado dado em (2.56) pode ser obtido de maneira mais simples utilizando uma equação de diferenças escalar para $\mathbf{k}'(n)$, ou seja, utilizando a variância do vetor de erro nos coeficientes (definida como $E[\mathbf{v}^H(n+1)\mathbf{v}(n+1)]$). No entanto, devido à forma como os valores esperados encontrados são calculados, o modelo resultante fica restrito a sinais de entrada não-correlacionados [14], [78].

Analisando-se (2.56) para valores determinados de passo de adaptação, ordem do filtro e EQM mínimo, é possível constatar que o EQM em excesso em regime permanente para o algoritmo NLMS operando com dados reais será maior comparativamente ao caso de dados complexos. Além disso, comparando tal expressão com sua versão equivalente para o algoritmo LMS, reproduzida aqui como [10]

$$J_{\text{ex}}^{\text{LMS}}(\infty) = \frac{\mu M \sigma_x^2}{2 - \mu \sigma_x^2 (M + \zeta)} J_{\min} \quad (2.57)$$

¹² Tal identidade é um caso particular do lema de inversão de matrizes (identidade de Woodbury [11]), estabelecendo que $(\Xi + \mathbf{v}\mathbf{v}^T)^{-1} = \Xi^{-1} - (\mathbf{1} + \mathbf{v}^T \Xi^{-1} \mathbf{v})^{-1} \Xi^{-1} \mathbf{v}\mathbf{v}^T \Xi^{-1}$, onde Ξ denota uma matriz com dimensão $M \times M$ e \mathbf{v} , um vetor com dimensão $M \times 1$ [2], [26], [33].

percebe-se que esta última é função da potência do sinal de entrada. Portanto, comparando (2.56) com (2.57), pode-se inferir que o algoritmo NLMS é menos sensível a variações da potência do sinal de entrada.

2.5.2 Desajuste

O desajuste é um parâmetro adimensional que fornece uma medida quantitativa de quão próximo os coeficientes do filtro adaptativo estão em relação ao ótimo no sentido do EQM. Para o algoritmo NLMS, tal métrica, definida em (1.19), é obtida a partir de (2.42), (2.45) e (2.27) como [1], [30]

$$\mathcal{M} = \mu^2 \boldsymbol{\lambda}^T (\mathbf{I} - \mathbf{B})^{-1} \mathbf{s}. \quad (2.58)$$

Entretanto, novamente, a solução envolve a inversa da matriz $(\mathbf{I} - \mathbf{B})$. Tal inversa, para o caso de sinais de entrada correlacionados, não possui solução analítica (veja Seção 2.5.1). Assim, visando à obtenção de uma formulação fechada, recorre-se novamente a aproximação definida em (2.50). Dessa forma, a partir de (2.53), obtém-se

$$\mathcal{M} \cong \frac{\mu}{2} \boldsymbol{\lambda}^T \mathbf{A}^{-1} \left(\mathbf{I} + \frac{\mu}{2} \mathbf{P} \mathbf{A}^{-1} \right) \mathbf{s}. \quad (2.59)$$

Essa expressão, apesar de válida para sinais de entrada não-correlacionados, é recomendada apenas para o caso de sinais de entrada correlacionados. Particularmente, para o caso de sinais de entrada não-correlacionados, a seguinte solução é proposta:

$$\mathcal{M} = \frac{\mu}{(2 - \mu)} \frac{M}{(M - \zeta)}. \quad (2.60)$$

A partir de (2.59) e (2.60), constata-se que tais expressões não são funções convexas de μ , como ocorre quando o algoritmo opera em um ambiente não-estacionário [24], [38], [82]. Além do mais, observa-se que o desajuste é uma função crescente com respeito ao passo de adaptação. Dessa forma, considerando uma condição de adaptação lenta (passo de adaptação pequeno), o desajuste tende a ser pequeno (insignificante).

2.5.3 Relacionando os algoritmos NLMS e LMS

Analisando (1.16) e (1.17), é possível perceber que existe uma relação muito estreita entre os algoritmos LMS e NLMS [54], isto é, através de uma mudança de variáveis, o algoritmo NLMS pode ser escrito em função do algoritmo LMS [10]. Tal abordagem, intuitivamente, indica que possivelmente existam expressões que

relacionem o desempenho desses dois algoritmos. Aqui, especificamente para o caso de sinais de entrada não-correlacionados, algumas dessas relações são apresentadas.

Considerando sinais de entrada não-correlacionados, substituindo (2.16) em (2.12), a recursão para o comportamento médio do vetor de coeficientes do algoritmo NLMS pode ser escrita como

$$E[\mathbf{w}(n+1)] = \left(1 - \frac{\mu}{M}\right) E[\mathbf{w}(n)] + \frac{\mu}{M} \mathbf{w}_o. \quad (2.61)$$

Por outro lado, para o algoritmo LMS, a equação que descreve o comportamento médio do vetor de coeficientes é dada por [1], [2], [9], [30], [38]

$$E[\mathbf{w}(n+1)] = (1 - \mu_{\text{LMS}} \sigma_x^2) E[\mathbf{w}(n)] + \mu_{\text{LMS}} \sigma_x^2 \mathbf{w}_o. \quad (2.62)$$

Agora, por inspeção em (2.61) e (2.62), é possível identificar que a taxa de convergência desses algoritmos pode ser tornada aproximadamente igual fazendo-se

$$\mu = M \sigma_x^2 \mu_{\text{LMS}}. \quad (2.63)$$

Portanto, quando o sinal de entrada é não-correlacionado, os algoritmos apresentam a mesma velocidade de convergência se os valores de passo de adaptação forem adequadamente ajustados [veja (2.63)] [14], [66]. De maneira análoga, considerando algumas aproximações, esse mesmo resultado pode ser obtido restringindo o modelo de $\mathbf{k}'(n)$ dos algoritmos LMS e NLMS para apresentarem o mesmo comportamento [14].

Outra medida de comparação entre algoritmos adaptativos se dá através da velocidade de convergência. Assim, visando tornar essa comparação justa, todos os algoritmos sob análise devem ser ajustados para apresentarem um mesmo valor para o desajuste ou o EQM em excesso em regime permanente [64], [83]. Nesse contexto, a partir de (1.19) e (2.57), é possível mostrar que o desajuste do algoritmo LMS é dado por [1], [10], [30]

$$\mathcal{M}^{\text{LMS}} = \frac{\mu_{\text{LMS}} M \sigma_x^2}{2 - \mu_{\text{LMS}} \sigma_x^2 (M + \zeta)}. \quad (2.64)$$

Dessa forma, comparando (2.60) com (2.64), as seguintes relações entre o passo de adaptação dos algoritmos LMS e NLMS são obtidas

$$\mu = \frac{\mu_{\text{LMS}} \sigma_x^2 (M - \zeta)}{1 - \mu_{\text{LMS}} \zeta \sigma_x^2} \quad (2.65)$$

$$\mu_{\text{LMS}} = \frac{\mu}{\sigma_x^2 [M - \zeta(1 - \mu)]}. \quad (2.66)$$

Portanto, utilizando (2.65) ou (2.66), é possível ajustar os algoritmos LMS e NLMS para um mesmo valor de desajuste (ou EQM em excesso em regime permanente). Salienta-se ainda que, para o caso de sinais de entrada correlacionados, relações equivalentes a (2.63), (2.65) e (2.66) não puderam ser obtidas devido à forma como os elementos de \mathbf{H} , \mathbf{T} , \mathbf{P} e \mathbf{s} são calculados nesse tipo de situação.

2.5.4 Efeito da normalização no desempenho do algoritmo

É sabido que a taxa (ou velocidade) de convergência do algoritmo LMS sofre grande influência do grau de condicionamento da matriz de autocorrelação do sinal de entrada [1], [11], [54]. Esse parâmetro, denominado dispersão de autovalores, é definido como

$$\chi(\mathbf{R}) = \frac{\lambda_{\max}}{\lambda_{\min}} \quad (2.67)$$

onde λ_{\max} e λ_{\min} são, respectivamente, o maior e menor autovalor de \mathbf{R} [2], [11], [53]. Tal medida influi diretamente no formato da superfície de desempenho do algoritmo adaptativo, tornando-a mais ou menos excêntrica e, dessa forma, resultando em uma maior ou menor velocidade de convergência [1], [11]. Como regra geral, para os algoritmos baseados no gradiente estocástico, a velocidade de convergência diminui quando a dispersão dos autovalores de \mathbf{R} aumenta. Por outro lado, à medida que $\chi(\mathbf{R}) \rightarrow 1$ (sinal de entrada branco), as curvas de níveis da superfície de desempenho passam a apresentar formas próximas às circulares, tornando a convergência mais homogênea (rápida) [30]. Dessa forma, é evidente que a razão dada por (2.67) deve ser mantida o mais próximo possível da unidade.

Visando possibilitar um estudo sobre o efeito da normalização na matriz de autocorrelação do sinal de entrada, juntamente com a interpretação geométrica da superfície de desempenho, um filtro adaptativo de segunda ordem é aqui considerado. Nesse cenário, o sinal de entrada tem matriz de autocorrelação definida como

$$\mathbf{R} = \sigma_x^2 \begin{bmatrix} 1 & -\alpha \\ -\alpha & 1 \end{bmatrix} \quad (2.68)$$

onde α corresponde ao coeficiente de correlação de um processo auto-regressivo de primeira ordem AR(1). Tal processo é representado por [2], [11]

$$x(n) = -\alpha x(n-1) + \sqrt{(1-\alpha^2)}v(n) \quad (2.69)$$

onde $v(n)$ denota um ruído branco com variância unitária.

Os autovalores de (2.68), para $0 < \alpha < 1$, são dados por [1]

$$\lambda_1 = 1 - \alpha \quad (2.70)$$

e

$$\lambda_2 = 1 + \alpha \quad (2.71)$$

sendo a matriz de autovetores definida como

$$\mathbf{Q} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (2.72)$$

Logo, substituindo (2.70) e (2.71) em (2.67), a dispersão de autovalores de \mathbf{R} é

$$\chi(\mathbf{R}) = \frac{1 + \alpha}{1 - \alpha}. \quad (2.73)$$

De (2.73), verifica-se que conforme o coeficiente de correlação é aumentado ($\alpha \rightarrow 1$), maior a dispersão de autovalores da matriz de autocorrelação e, conseqüentemente, menor a velocidade de convergência do algoritmo adaptativo.

Para estudar o efeito da normalização na velocidade de convergência do algoritmo NLMS, (2.12) é reescrita em função do vetor de erro nos coeficientes (ou seja, realiza-se uma translação nos eixos do sistema de coordenadas [9]). Assim, subtraindo \mathbf{w}_0 de ambos os lados de (1.18), substituindo (2.20) na expressão resultante, tomando o valor esperado em ambos os lados e utilizando as Hipóteses H1)-H3) (veja Seção 2.4.1), tem-se

$$E[\mathbf{v}(n+1)] = (\mathbf{I} - \mu \mathbf{R}^N) E[\mathbf{v}(n)]. \quad (2.74)$$

Usando (2.13) em (2.74) e denotando o vetor de erro nos coeficientes (com os eixos rotacionados em função dos autovetores da matriz de autocorrelação [1], [9]) como

$$\mathbf{v}'(n) = \mathbf{Q}^H E[\mathbf{v}(n)] \quad (2.75)$$

obtem-se

$$\mathbf{v}'(n+1) = (\mathbf{I} - \mu \mathbf{H}) \mathbf{v}'(n). \quad (2.76)$$

Assim, constata-se que o momento de primeira ordem do vetor de erro nos coeficientes do algoritmo NLMS converge se o módulo dos autovalores de $(\mathbf{I} - \mu \mathbf{H})$ for menor do que a unidade. Note que, fazendo

$\mathbf{H} = \mathbf{\Lambda}$, (2.76) resulta no momento de primeira ordem do vetor de erro nos coeficientes do algoritmo LMS [1], [2], [9], [11], [30], [38].

Agora, considerando que o sinal de entrada é complexo, com matriz de autocorrelação dada por (2.68), os elementos de \mathbf{H} podem ser calculados por

$$h_{1,1}^C = \frac{\lambda_1}{(\lambda_2 - \lambda_1)^2} \left\{ \lambda_1 - \lambda_2 \left[1 + \log \left(\frac{\lambda_1}{\lambda_2} \right) \right] \right\} \quad (2.77)$$

e

$$h_{2,2}^C = \frac{\lambda_2}{(\lambda_2 - \lambda_1)^2} \left\{ \lambda_2 - \lambda_1 \left[1 - \log \left(\frac{\lambda_1}{\lambda_2} \right) \right] \right\}. \quad (2.78)$$

Então, assumindo $\lambda_2 > \lambda_1$, substituindo (2.77) e (2.78) em (2.67), e considerando (2.70), (2.71) e (2.73), a dispersão de autovalores de \mathbf{R}^N pode ser escrita como

$$\chi(\mathbf{R}^N) = \chi(\mathbf{R})\Gamma(\alpha) \quad (2.79)$$

com

$$\Gamma(\alpha) = \frac{(1 + \alpha) - (1 - \alpha) \{1 + \log[\chi(\mathbf{R})]\}}{(1 - \alpha) - (1 + \alpha) \{1 - \log[\chi(\mathbf{R})]\}}. \quad (2.80)$$

Analisando (2.80), para $0 < \alpha < 1$, constata-se que tal função é positiva e decrescente em relação ao parâmetro α . Adicionalmente, como uma matriz de autocorrelação é sempre não negativa definida e quase sempre definida positiva [11], pode-se inferir que a dispersão de autovalores de \mathbf{R} é uma função não negativa. Portanto, a partir de (2.79) e (2.80), torna-se evidente que $\chi(\mathbf{R}^N)$ assume um valor não negativo, menor ou igual a $\chi(\mathbf{R})$.

A Figura 2.1 ilustra as curvas de dispersão de autovalores obtida através de (2.73) e (2.79) para diferentes valores de α . Tais curvas mostram que quanto mais correlacionado for o sinal de entrada, menor será a dispersão de autovalores de \mathbf{R}^N se comparada à \mathbf{R} . Tal característica, justifica, parcialmente, o motivo de o algoritmo NLMS apresentar uma velocidade de convergência maior comparativamente ao algoritmo LMS, sobretudo em se tratando de sinais de entrada correlacionados. Por outro lado, quando o sinal de entrada possui fraca correlação entre suas amostras, a diferença entre $\chi(\mathbf{R})$ e $\chi(\mathbf{R}^N)$ é insignificante, tendo-se dessa forma pouca vantagem no uso do algoritmo NLMS em relação ao LMS.

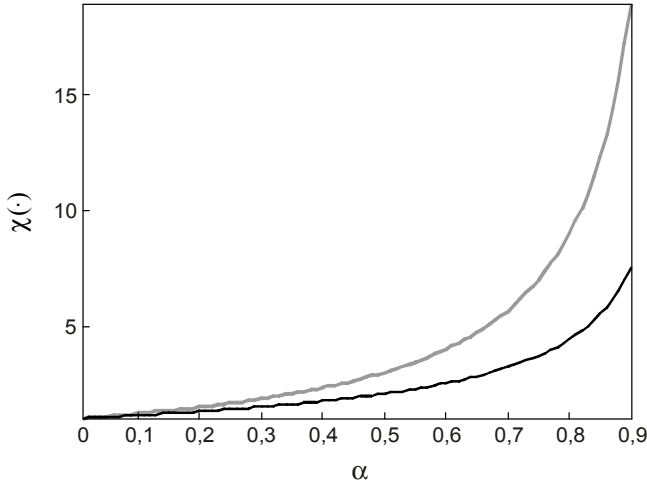


Figura 2.1: Dispersão de autovalores de \mathbf{R} (linha sólida cinza) e \mathbf{R}^N (linha sólida escura).

Analisando o caso de sinais de entrada reais, a seguinte relação entre a dispersão dos autovalores de \mathbf{R}^N e \mathbf{R} é obtida [53]:

$$\chi(\mathbf{R}^N) = \sqrt{\chi(\mathbf{R})}. \quad (2.81)$$

Novamente, verifica-se que a matriz de autocorrelação normalizada apresenta uma dispersão de autovalores menor do que a da matriz de autocorrelação do sinal de entrada; sendo essa redução mais significativa do que a obtida para o caso de sinais de entrada complexos.

Na literatura, buscando justificar a velocidade convergência superior do algoritmo NLMS (comparativamente ao LMS), têm sido encontradas algumas abordagens visando generalizar o estudo sobre o efeito da normalização na redução da dispersão dos autovalores de \mathbf{R} . Por exemplo, em [53] é proposta uma conjectura mostrando que a dispersão de autovalores da matriz de autocorrelação normalizada é sempre menor ou igual a sua versão não normalizada. Entretanto, tal conjectura, carece de uma prova matemática formal demonstrando sua validade. Em [62], faz-se uso da desigualdade de Cauchy-Schwarz-Buniakowsky [84] para obter uma expressão aproximada para (2.81), válida independentemente da ordem do filtro adaptativo. Contudo, tal abordagem é eficiente somente quando a dispersão de autovalores de \mathbf{R} é pequena, apresentando então um resultado muito distante do comportamento efetivo na maioria dos casos práticos.

As aproximações comumente utilizadas na modelagem de algoritmos normalizados [16], [57], [61], [66] (veja Seção 2.3), apesar de simples, não conseguem capturar o efeito da normalização nas matrizes de autocorrelação, visto que, nesses casos, os autovalores de \mathbf{R} são somente escalados em função da ordem do filtro e da variância

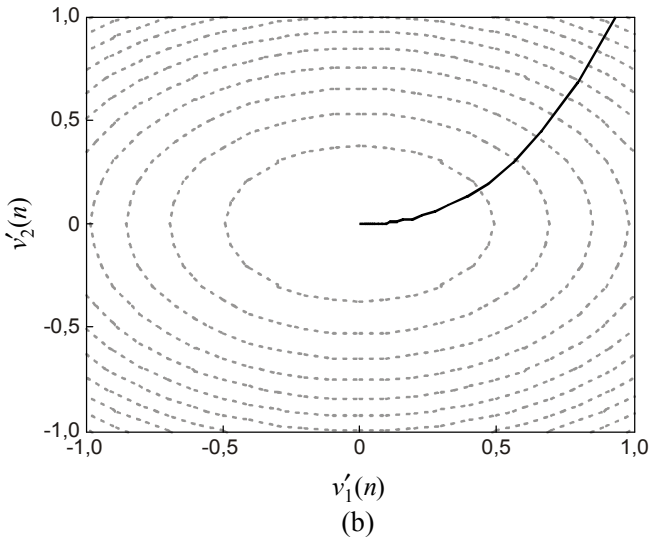
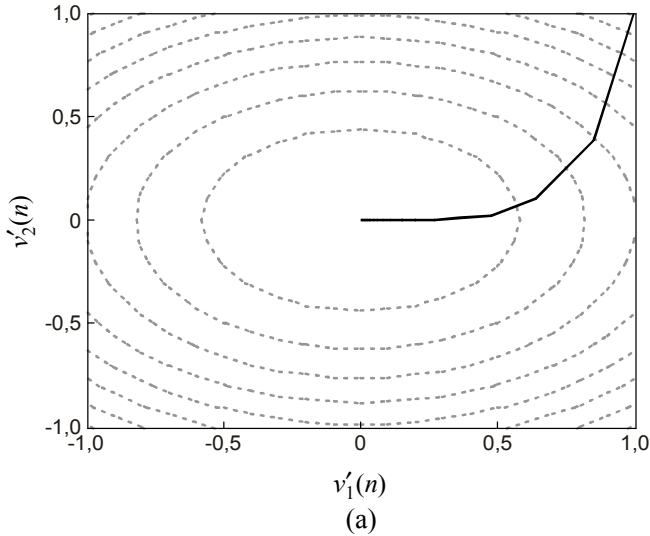


Figura 2.2: Superfície de desempenho de um filtro adaptativo de segunda ordem. (a) Algoritmo LMS. (b) Algoritmo NLMS.

do sinal de entrada. Tal característica, em nossa opinião, é uma das principais vantagens do algoritmo NLMS sobre os algoritmos não-normalizados e, quando desconsiderada, acarreta em um modelo impreciso.

Sabe-se que o efeito da normalização do sinal de entrada se reflete no formato da superfície de desempenho. Assim, utilizando alguns resultados mostrados anteriormente (veja Seção 2.4), conclusões podem ser obtidas acerca do formato (aproximado) da superfície de desempenho dos algoritmos LMS e NLMS. Como exemplo, considere o vetor de coeficientes ótimo dado por $\mathbf{w}_o = [1,9 \quad -0,3]^T$, o passo de adaptação $\mu = 0,5$, o coeficiente de correlação $\alpha = 0,5$ e o erro mínimo $J_{\min} = 10^{-4}$. Baseado nesse cenário, a Figura 2.2 apresenta as curvas de níveis correspondentes à superfície de desempenho juntamente com a trajetória de convergência dos coeficientes para os algoritmos LMS [Figura 2.2(a)] e NLMS [Figura 2.2(b)]. Para o algoritmo LMS, observa-se uma deformação das elipses na direção do autovalor de maior magnitude, o que resulta em uma degradação na velocidade de convergência desse algoritmo. Para esse mesmo cenário, a Figura 2.2(b) ilustra o formato das curvas de níveis da superfície de desempenho do algoritmo NLMS associada com a curva de convergência dos coeficientes. Comparando-a com a Figura 2.2(a), percebe-se que a operação de normalização modifica substancialmente a geometria da superfície de desempenho, levando a uma redução na excentricidade dessas curvas de níveis. Dessa forma, o algoritmo normalizado tende a apresentar uma característica de convergência superior à do algoritmo LMS.

No intuito de verificar a precisão do modelo analítico descrevendo o comportamento médio do algoritmo NLMS, bem como das expressões que regem seu desempenho no regime permanente, no próximo capítulo o modelo proposto é avaliado sob diferentes cenários de operação.

2.6 Conclusões

Neste capítulo, assumindo um número reduzido de hipóteses simplificativas, foi derivado um modelo estocástico para o algoritmo NLMS operando em um ambiente estacionário. Tal modelo contempla o caso de sinais de entrada gaussianos reais e complexos, como também correlacionados e não-correlacionados. Para o caso de sinais de entrada complexos, a abordagem utilizada permite que os momentos

normalizados sejam calculados de forma exata. Para o caso de sinais de entrada reais, uma solução aproximada restrita a filtros adaptativos de ordem par foi também obtida. A partir das expressões que descrevem o comportamento médio do algoritmo, foram derivadas expressões que relacionam algumas métricas de desempenho (em regime permanente) com os parâmetros de controle do algoritmo NLMS. Finalmente, para o caso de sinal de entrada não-correlacionado, foram obtidas expressões que relacionam a velocidade de convergência e o EQM em excesso em regime permanente dos algoritmos NLMS e LMS através do passo de adaptação.

RESULTADOS DE SIMULAÇÃO

3.1 Introdução

Neste capítulo, com o objetivo de verificar a precisão do modelo estocástico proposto para o algoritmo NLMS, são apresentados resultados de simulação considerando um problema de identificação de sistema. A escolha dos parâmetros utilizados nos exemplos visa representar, de forma clara e concisa, suas vantagens e limitações. Os resultados obtidos são organizados em três seções, a primeira tratando o caso de sinal de entrada complexo, a segunda considerando sinal de entrada real e, por fim, são verificadas as expressões que descrevem o desempenho do algoritmo em regime permanente. Quando cabível, o comportamento médio previsto pelo modelo proposto é aqui confrontado com simulações numéricas de Monte Carlo (MC) e com outros modelos teóricos encontrados na literatura.

3.2 Ambiente de operação

Como ambiente de operação, considera-se um problema de identificação de sistema com planta estacionária. A planta (sistema a ser identificado) utilizada é normalizada através da seguinte relação

$$\mathbf{w}_o = \frac{\mathbf{w}_o}{\sqrt{\mathbf{w}_o^H \mathbf{w}_o}} \quad (3.1)$$

de forma que \mathbf{w}_o possua norma unitária [17], [18], [21], [24], [57], [66]. As ordens do sistema a ser identificado \mathbf{w}_o e do filtro adaptativo são assumidas iguais, sendo os coeficientes do filtro adaptativo inicializados com zero, isto é, $\mathbf{w}(0) = \mathbf{0}$. Nesse contexto, para todos os exemplos, o comportamento médio do algoritmo é estimado através de simulações de MC considerando 200 realizações independentes.

O sinal de entrada do sistema em questão é oriundo de um processo auto-regressivo (*autoregressive* - AR) de segunda ordem, cuja equação de diferenças é dada por [2], [11]

$$x(n) = -a_1 x(n-1) - a_2 x(n-2) + v(n) \quad (3.2)$$

onde a_1 e a_2 são os coeficientes do processo AR(2) e $v(n)$, um ruído branco gaussiano cuja variância σ_v^2 é ajustada através da seguinte relação:

$$\sigma_v^2 = \left(\frac{1-a_2}{1+a_2} \right) [(1+a_2)^2 - a_1^2] \quad (3.3)$$

de modo que $x(n)$ tenha variância unitária. Para o caso de dados complexos, a variância¹ σ_v^2 é dividida igualmente entre a parte real e imaginária de $v(n)$.

O ruído de medição inerente ao ambiente de operação do filtro adaptativo tem variância $\sigma_z^2 = 10^{-4}$, implicando uma razão sinal-ruído (*signal-to-noise ratio* - SNR) de 40 dB (visto que, $\sigma_x^2 = 1$). Tal valor representa o EQM mínimo alcançado pelo algoritmo adaptativo quando $\mathbf{w}(n) = \mathbf{w}_0$. Baseado nesse cenário de operação, diversos exemplos são apresentados a seguir visando avaliar a precisão do modelo proposto.

3.3 Sinal de entrada gaussiano complexo

Esta seção trata especificamente de exemplos envolvendo sinais de entrada gaussianos complexos. Primeiramente, o caso de sinal de entrada branco é avaliado, juntamente com as respectivas soluções para os momentos normalizados [veja (2.16), (2.39), (2.40) e (2.41)]. Em seguida, nesse mesmo contexto, aborda-se o caso de sinal de entrada correlacionado, sendo as soluções para os momentos normalizados agora dadas por (2.15), (2.36), (2.37) e (2.38).

3.3.1 Exemplo 1

Neste exemplo, considera-se sinal de entrada branco, passo de adaptação $\mu = 0,1$ e um filtro adaptativo de ordem $M = 5$. A planta a ser identificada é oriunda do modelo de um canal de comunicações (canal de desvanecimento Rayleigh² [10]) e possui o seguinte vetor de coeficientes:

¹ A variância de uma variável aleatória complexa ϕ é definida como $\sigma_\phi^2 = E[|\phi - E(\phi)|^2]$. Logo, assumindo média nula, tem-se $\sigma_\phi^2 = \sigma_{\phi,\mathbb{R}}^2 + \sigma_{\phi,\mathbb{C}}^2$ [10].

² Este tipo de canal é comumente utilizado na modelagem de ambientes de comunicações sem fio (*wireless*), pois ele consegue capturar satisfatoriamente os efeitos das reflexões e atenuações que o sinal sofre quando viaja do transmissor para o receptor [10].

$$\mathbf{w}_0 = [0,2913 - j0,1786 \quad -0,2940 + j0,1212 \quad 0,3651 - j0,4814 \quad -0,2472 - j0,5627 \quad 0,0501 + j0,1918]^T. \quad (3.4)$$

A Figura 3.1 ilustra o comportamento médio das partes real [Figura 3.1(a)] e imaginária [Figura 3.1 (b)] dos coeficientes do filtro

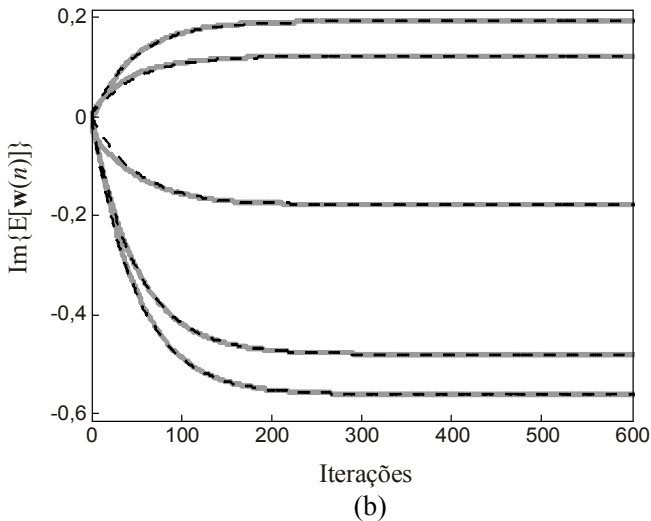
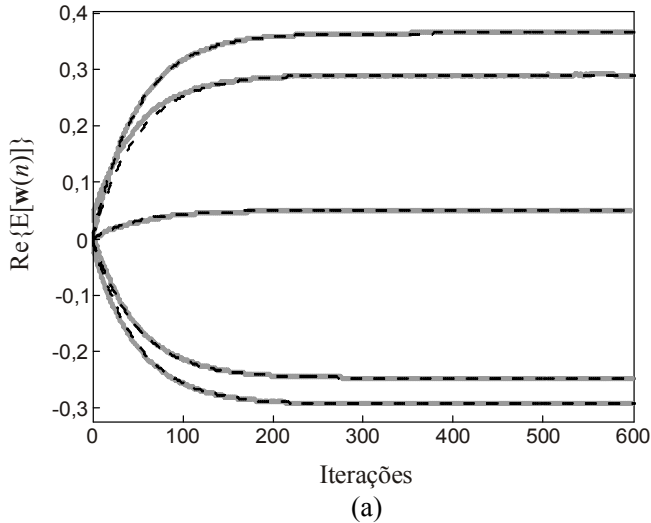


Figura 3.1: Exemplo 1. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.

adaptativo estimado via simulação MC e obtido através do modelo proposto. Consta-se que a precisão do modelo proposto é muito boa tanto para o regime transitório quanto para o regime permanente.

Na Figura 3.2 é mostrada a curva de aprendizagem teórica (predita pelo modelo proposto) e a curva de aprendizagem estimada (obtida através de simulações MC) do algoritmo adaptativo NLMS. Novamente, observe a excelente capacidade de predição obtida para o caso de sinal de entrada não-correlacionado (branco). Tal precisão se deve ao reduzido número de hipóteses simplificadoras utilizadas ao longo do processo de modelagem do algoritmo NLMS e também a forma como o cálculo dos momentos normalizados é efetuado.

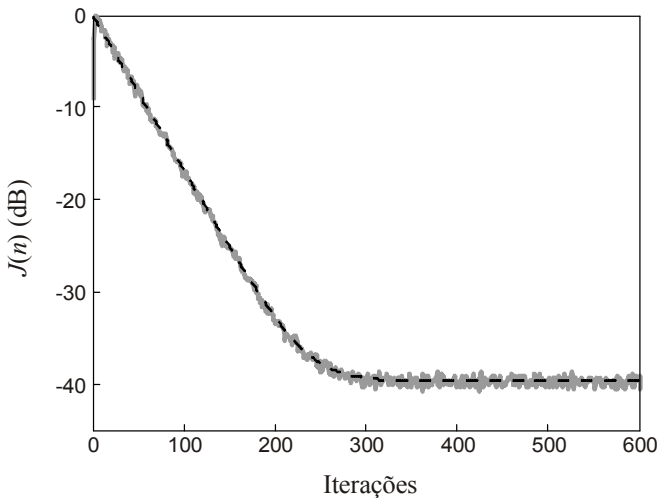


Figura 3.2: Exemplo 1. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.

3.3.2 Exemplo 2

Visando comparar o desempenho dos algoritmos LMS e NLMS, isto é, verificar as relações (2.63) e (2.66), assumem-se os mesmos parâmetros utilizados no exemplo anterior. De (2.63), verifica-se que o passo de adaptação do algoritmo LMS deve ser ajustado para $\mu_{\text{LMS}} = 0,02$, de forma a proporcionar a mesma velocidade de convergência do algoritmo NLMS. Em contrapartida, para alcançar o mesmo desajuste (EQM em excesso em regime permanente), de (2.66) obtém-se $\mu_{\text{LMS}} = 0,0244$.

Na Figura 3.3, os algoritmos LMS e NLMS foram ajustados [utilizando (2.63)] para uma mesma velocidade de convergência. Os resultados apresentados confirmam que a velocidade de convergência desses algoritmos é igual para valores de passo de adaptação adequadamente escolhidos, corroborando com o fato de que o algoritmo NLMS tende a apresentar um EQM em excesso em regime permanente (desajuste) maior do o algoritmo LMS. A Figura 3.4 ilustra o desempenho dos algoritmos LMS e NLMS considerando o mesmo desajuste. Dessa forma, para um dado valor de desajuste, mostra-se que o algoritmo NLMS apresenta uma velocidade de convergência inferior quando comparada ao algoritmo LMS.

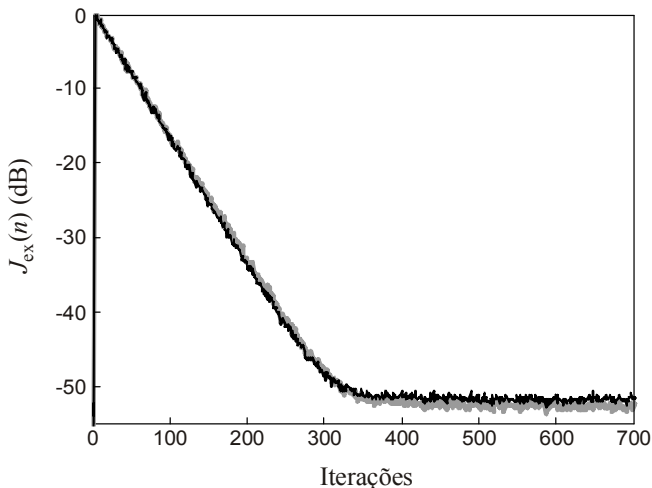


Figura 3.3: Exemplo 2. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando uma mesma velocidade de convergência.

3.3.3 Exemplo 3

Neste exemplo, considera-se um filtro adaptativo de ordem $M = 8$, passo de adaptação $\mu = 0,1$ e sinal de entrada correlacionado. Os coeficientes do processo AR(2) [veja (3.2)] são iguais a $a_1 = -0,55$ e $a_2 = 0,9$, resultando em uma dispersão de autovalores na matriz de autocorrelação do sinal de entrada de $\chi(\mathbf{R})=115$. O sistema a ser identificado é obtido como no Exemplo 1 e é dado por (3.5).

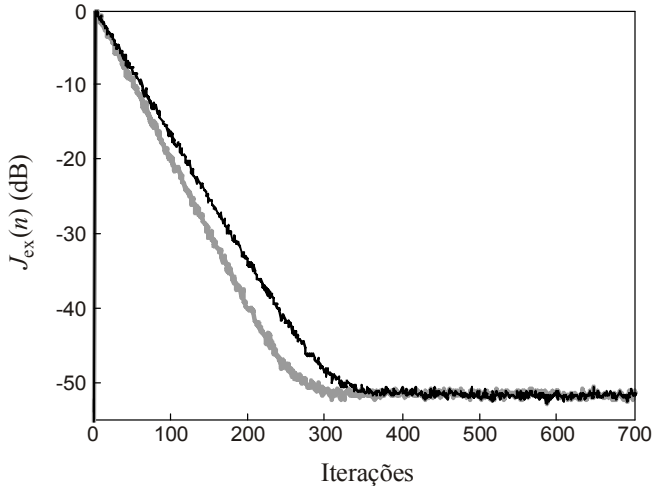


Figura 3.4: Exemplo 2. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando um mesmo valor em regime permanente.

$$\begin{aligned} \mathbf{w}_o = & [-0,0997 + j0,0436 \quad -0,0515 + j0,2498 \quad -0,5326 + j0,0192 \\ & -0,3876 - j0,2292 \quad 0,3035 + j0,1351 \quad -0,2151 - j0,1889 \quad (3.5) \\ & 0,0206 - j0,1298 \quad 0,1921 + j0,4357]^T \end{aligned}$$

A Figura 3.5 mostra o comportamento médio das partes real [Figura 3.5(a)] e imaginária [Figura 3.5(b)] dos coeficientes do filtro adaptativo. Constata-se que o modelo proposto apresenta uma excelente capacidade de predição, capturando fielmente a evolução dos coeficientes adaptativos tanto em regime transitório quanto em regime permanente. Salienta-se que, é em regime transitório que os modelos tendem a apresentar maior dificuldade em prever o comportamento do algoritmo, exibindo então certo descasamento entre os resultados de simulação MC e os obtidos através do modelo.

A Figura 3.6 apresenta os resultados obtidos para a curva de aprendizagem do algoritmo NLMS. Verifica-se que o modelo proposto está em concordância com a simulação MC para um sinal de entrada correlacionado. Vale lembrar que, sinais de entrada correlacionados representam o principal obstáculo à modelagem de algoritmos adaptativos. Assim, visando facilitar a sua derivação, muitas vezes são utilizadas aproximações que originam modelos pouco realistas quando comparados à simulação numérica do algoritmo.

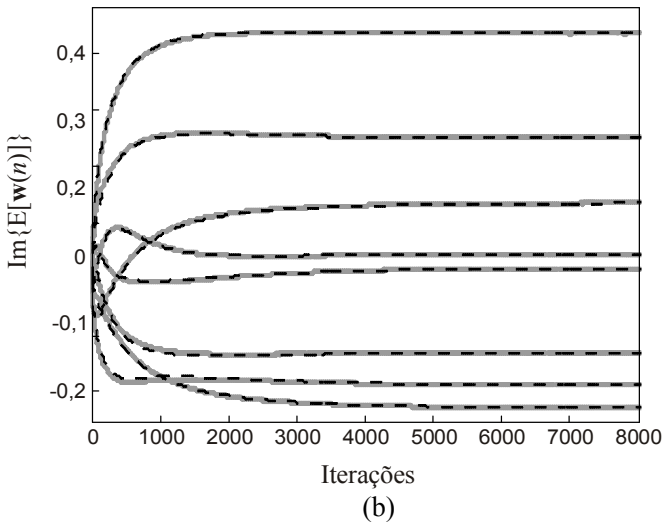
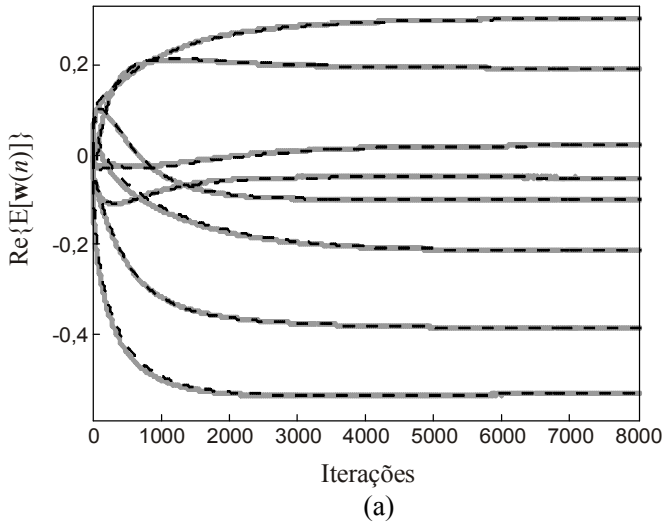


Figura 3.5: Exemplo 3. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.

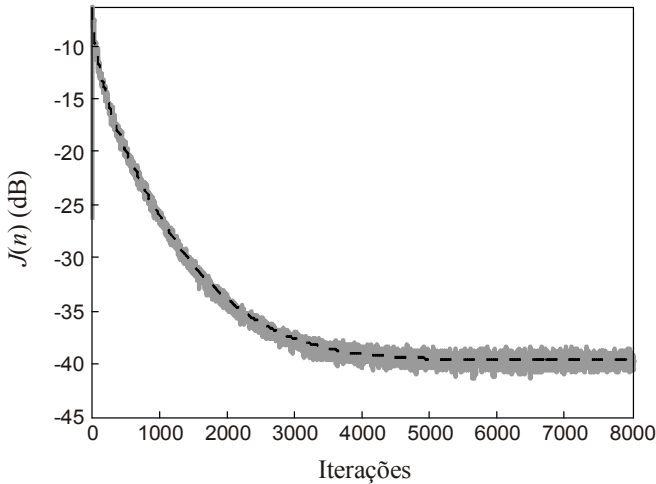


Figura 3.6: Exemplo 3. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.

3.3.4 Exemplo 4

Com o objetivo de avaliar a eficiência do modelo proposto para o algoritmo NLMS utilizando plantas com respostas mais longas, a ordem do filtro adaptativo aqui é aumentada para $M = 64$. Nesse cenário, os coeficientes do processo AR(2) são $a_1 = -0,65$ e $a_2 = 0,8$, produzindo assim uma dispersão de autovalores na matriz de autocorrelação do sinal de entrada de $\chi(\mathbf{R}) = 155$. O passo de adaptação é mantido em $\mu = 0,1$, proporcionando uma condição de adaptação lenta. O sistema a ser identificado é obtido de maneira similar a do Exemplo 1 e é dado por (3.6).

A Figura 3.7 mostra o comportamento médio das partes real [Figura 3.7(a)] e imaginária [Figura 3.7(b)] de apenas quatro coeficientes [sublinhados em (3.6)] do filtro adaptativo (visando uma melhor visualização). Esses coeficientes são escolhidos considerando um espaçamento entre eles igual a dezesseis amostras. Novamente, verifica-se um casamento muito bom entre os resultados de simulação MC e os preditos pelo modelo proposto. Ainda referente ao mesmo cenário, a Figura 3.8 mostra a evolução do EQM do algoritmo NLMS. Constata-se também uma muito boa concordância entre os resultados de simulação MC e o do modelo proposto.

$$\begin{aligned}
 \mathbf{w}_o = & [0,1079 + j0,0095 \quad -0,0204 + j0,0524 \quad 0,0118 + j0,046 \\
 & 0,0667 + j0,1141 \quad 0,0723 + j0,1665 \quad 0,1350 - j0,0317 \\
 & -0,0834 + j0,0608 \quad 0,2083 - j0,0628 \quad -0,0255 - j0,0038 \\
 & -0,0111 - j0,0866 \quad -0,1267 + j0,0723 \quad 0,0199 + j0,070 \\
 & -0,02 + j0,0216 \quad -0,0230 + j0,0202 \quad 0,0675 + j0,1279 \\
 & 0,0143 - j0,0006 \quad \underline{0,0416 + j0,0246} \quad 0,0772 + j0,1576 \\
 & 0,086 - j0,0735 \quad 0,0666 + j0,1057 \quad -0,0831 + j0,1036 \\
 & -0,0409 + j0,0197 \quad -0,0272 + j0,0167 \quad -0,0472 - j0,0475 \\
 & 0,0906 + j0,165 \quad 0,094 - j0,0289 \quad -0,1552 + j0,0771 \\
 & -0,0041 - j0,0159 \quad 0,067 + j0,0303 \quad 0,1383 - j0,0572 \\
 & -0,0575 - j0,077 \quad 0,0127 + j0,1569 \quad \underline{0,1902 - j0,0751} \\
 & -0,0554 + j0,0985 \quad -0,1234 - j0,012 \quad 0,1131 + j0,0683 \\
 & 0,0795 + j0,069 \quad -0,1486 + j0,0744 \quad -0,0902 + j0,0847 \\
 & -0,0149 + j0,2029 \quad -0,0124 - j0,0458 \quad -0,0569 - j0,0036 \\
 & 0,0997 + j0,0563 \quad 0,1605 + j0,0038 \quad 0,0303 + j0,0196 \\
 & -0,0403 - j0,1039 \quad 0,0002 - j0,0326 \quad 0,006 + j0,0946 \\
 & \underline{-0,1338 + j0,1108} \quad 0,0119 - j0,0567 \quad -0,0617 - j0,2095 \\
 & -0,0674 - j0,0113 \quad -0,0925 - j0,0027 \quad 0,008 - j0,1948 \\
 & 0,1468 - j0,1283 \quad -0,0394 + j0,1185 \quad -0,0465 + j0,038 \\
 & -0,043 + j0,1461 \quad 0,0139 - j0,0415 \quad 0,0007 - j0,0485 \\
 & 0,0491 - j0,0616 \quad -0,144 + j0,0576 \quad 0,144 + j0,231 \\
 & -0,0097 + j0,0285]^T
 \end{aligned}
 \tag{3.6}$$

Diante dos exemplos apresentados, conclui-se que o modelo proposto exibe uma muito boa capacidade de prever o comportamento do algoritmo NLMS, tanto em regime transitório quanto em regime permanente, independentemente da ordem do filtro adaptativo e da natureza do sinal de entrada. Além disso, devido à escassez de modelos confiáveis para o algoritmo NLMS operando com sinais de entrada complexos, nenhuma outra comparação adicional com modelos da literatura pôde ser realizada. Cabe lembrar que a TI é a aproximação mais grosseira utilizada nesse desenvolvimento. Assim, possíveis imprecisões na predição podem ser reduzidas à medida que o passo de adaptação do algoritmo é reduzido.

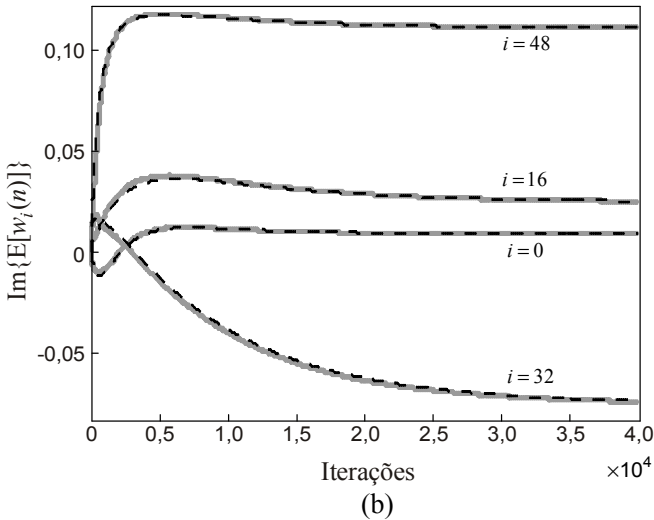
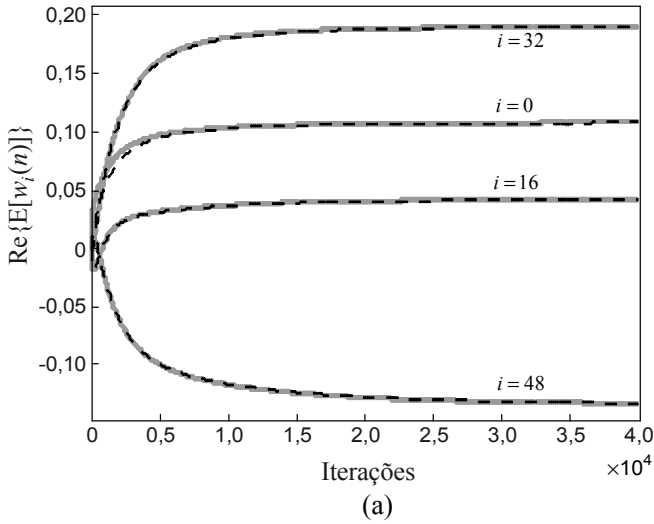


Figura 3.7: Exemplo 4. Comportamento médio de quatro coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras tracejadas) modelo proposto. (a) Parte real. (b) Parte imaginária.

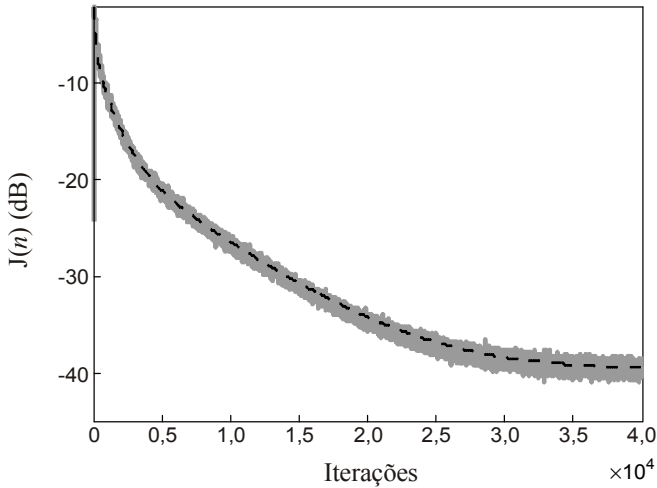


Figura 3.8: Exemplo 4. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura tracejada) modelo proposto.

3.4 Sinal de entrada gaussiano real

Nesta seção, considerando sinal de entrada gaussiano real, alguns exemplos são apresentados. Tais exemplos inicialmente tratam o caso de sinal de entrada não-correlacionado (branco) [veja (2.16), (2.39), (2.40) e (2.41)], sendo o caso de sinal de entrada correlacionado avaliado na sequência [veja (A.32), (B.24), (B.25) e (C.14)]. Quando cabível, visando avaliar a precisão do modelo proposto, o resultado aqui obtido é comparado com o de outros modelos propostos na literatura para o algoritmo NLMS (veja Seção 2.3).

Nos exemplos subsequentes, a planta utilizada é obtida a partir de uma função *sinc* (normalizada), a qual é definida como

$$\text{sinc}(m) = \begin{cases} 1, & m = 0 \\ \frac{\text{sen}(\pi m)}{\pi m}, & m \neq 0 \end{cases} \quad (3.7)$$

para

$$m = 0, \frac{1}{M}, \dots, \frac{M-1}{M}. \quad (3.8)$$

3.4.1 Exemplo 5

Neste exemplo, visando verificar a precisão do modelo proposto em relação aos modelos apresentados em [61] e [66], assume-se que a ordem do sistema desconhecido (planta) e do filtro adaptativo é $M = 6$, que o sinal de entrada é branco e que o passo de adaptação é $\mu = 0,05$. Especificamente em [61], são propostos dois modelos para o algoritmo NLMS, sendo apenas um deles considerado aqui, visto que, para a condição de adaptação lenta, ambos exibem praticamente o mesmo desempenho. Além disso, para o exemplo aqui apresentado, os modelos propostos em [14], [52] e [57] foram desconsiderados por apresentarem resultados similares ao modelo proposto, devido à forma como os momentos normalizados são calculados.

A Figura 3.9 mostra a evolução dos coeficientes do filtro adaptativo obtida via simulação MC juntamente com o comportamento predito pelos modelos considerados. Observa-se uma concordância muito boa entre os resultados de simulação MC e os obtidos pelo modelo proposto. Por outro lado, devido ao uso do PM no cálculo dos momentos normalizados e como nesse caso a ordem do filtro adaptativo é pequena ($M \ll 30$), os modelos de [61] e [66] exibem em regime transitório certo descasamento quando comparado com a simulação MC.

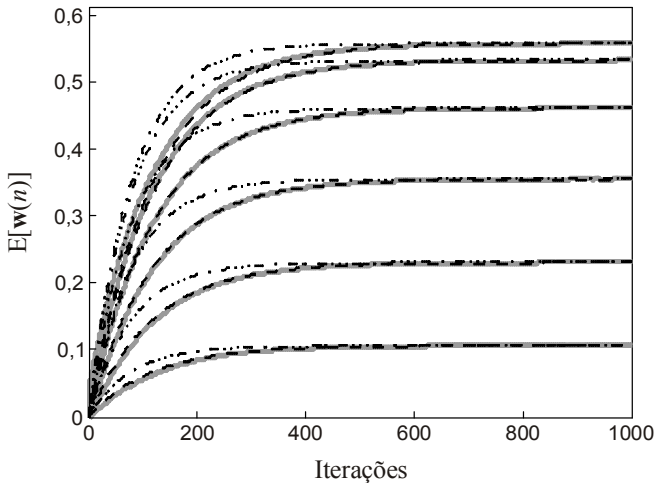


Figura 3.9: Exemplo 5. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [61]. (Linhas escuras traços-pontos) modelo de [66]. (Linhas escuras tracejadas) modelo proposto.

A Figura 3.10 ilustra a curva de aprendizagem do algoritmo NLMS para o cenário de operação em questão. Verifica-se um casamento muito bom entre a simulação MC e o modelo proposto. Por outro lado, observa-se um importante descasamento nos resultados obtidos com os modelos de [61] e [66]. Tal discrepância é atribuída à forma como os momentos normalizados [requeridos na derivação de $\mathbf{k}'(n)$] são calculados. Entretanto, para sinais de entrada não-correlacionados, constatou-se que à medida que a ordem do filtro adaptativo é aumentada, esse descasamento tende a diminuir, visto que, nessa condição, o critério de “variação lenta” do PM torna-se cada vez mais válido. Cabe aqui salientar que, no caso de sinais de entrada não-correlacionados, a abordagem utilizando a função densidade de probabilidade chi-quadrada para representar o termo $E\{\mathbf{x}^H(n)\mathbf{x}(n)\}^{-1}$ (no cálculo dos momentos normalizados) pode ser considerada conceitualmente correta.

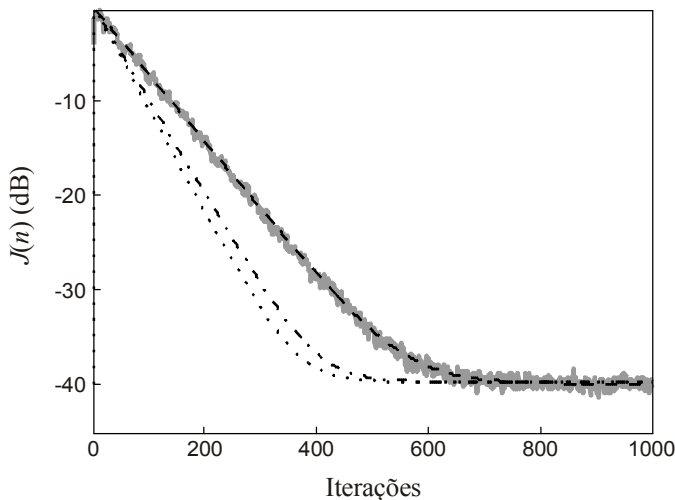


Figura 3.10: Exemplo 5. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [61]. (Linha escura traço-ponto) modelo de [66]. (Linha escura tracejada) modelo proposto.

Analisando detalhadamente o modelo proposto em [66], percebeu-se que tal modelo diverge para valores de passo de adaptação da ordem de $\mu = 0,5$, contrariando o comportamento efetivo do

algoritmo NLMS; visto que, valores de passo menores do que dois garantem a sua convergência. Além disso, constatou-se que os modelos descritos em [61] exibem certa dificuldade em prever a evolução do EQM tanto em regime transitório quanto em regime permanente, para valores de μ da ordem de 0,5. Por outro lado, tais características não puderam ser observadas no modelo estocástico proposto para o algoritmo NLMS, o qual tem mostrado (para o caso de sinais de entrada não-correlacionados) uma satisfatória capacidade de predição mesmo para valores de passo de adaptação próximos a $\mu = 0,5$.

3.4.2 Exemplo 6

Neste exemplo, busca-se comparar o desempenho dos algoritmos LMS e NLMS para sinais de entrada reais não-correlacionados. Aqui, os parâmetros do algoritmo são mantidos iguais aos do Exemplo 5, com exceção do passo de adaptação do algoritmo NLMS que é alterado para $\mu = 0,1$. Assim, visando à obtenção de uma mesma taxa de convergência, de (2.63) tem-se que o passo de adaptação do algoritmo LMS deve ser ajustado para $\mu_{\text{LMS}} = 0,0166$. Por outro lado, para obter um mesmo EQM em regime permanente, a partir de (2.66) obtém-se $\mu_{\text{LMS}} = 0,0238$.

A Figura 3.11 mostra a evolução do EQM em excesso dos algoritmos LMS e NLMS, ajustados para a mesma velocidade de convergência. Aqui, constata-se que o algoritmo NLMS apresenta um EQM em excesso em regime permanente (desajuste) maior do que o algoritmo LMS, sendo tal diferença mais evidente do que a observada para o caso de sinais de entrada complexos (veja Figura 3.3). Esta última afirmação pode ser verificada através de (2.56) ou (2.60). Por sua vez, a Figura 3.12 apresenta a evolução do EQM em excesso dos algoritmos LMS e NLMS considerando o ajuste do passo de adaptação para obter um mesmo EQM em excesso em regime permanente (desajuste). Novamente, verifica-se que o algoritmo NLMS exibe uma velocidade de convergência inferior quando comparada a do algoritmo LMS. Além disso, comparando a Figura 3.12 com a Figura 3.4, percebe-se claramente que a diferença na velocidade de convergência dos dois algoritmos (LMS e NLMS) é ainda mais evidente quando eles operam com sinais de entrada reais.

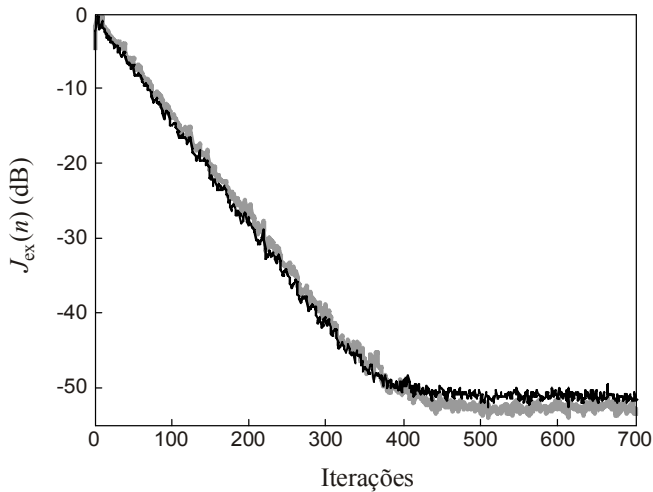


Figura 3.11: Exemplo 6. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando uma mesma velocidade de convergência.

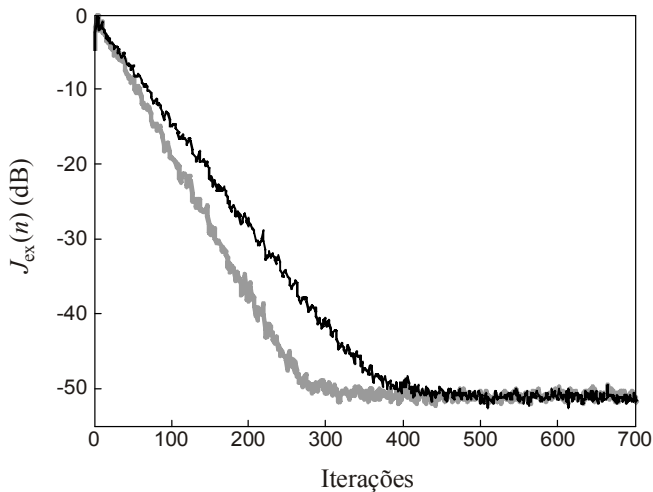


Figura 3.12: Exemplo 6. Evolução do EQM em excesso dos algoritmos LMS (linha irregular cinza) e NLMS (linha irregular escura) considerando um mesmo valor em regime permanente.

3.4.3 Exemplo 7

Neste exemplo, busca-se avaliar a influência do parâmetro de regularização no desempenho do algoritmo NLMS e também em seu modelo estocástico. Por consequência, a validade da Hipótese H1) (veja Seção 2.4.1) utilizada durante o processo de modelagem é verificada. Para tal, considere que o sinal de entrada é obtido através de um processo AR(2), cujos coeficientes são $a_1 = -0,55$ e $a_2 = 0,9$, resultando em $\chi(\mathbf{R}) = 115$; a planta utilizada tem ordem $M = 8$, o passo de adaptação é $\mu = 0,05$ e o parâmetro de regularização, $\varepsilon = 10^{-3}$. Nesse cenário, o modelo proposto é comparado com os de [57] e [60], pois somente esses modelos³ levam em consideração o parâmetro de regularização. Especificamente em [60], técnicas de integração numérica [63] são utilizadas para resolver as integrais requeridas no cálculo dos momentos normalizados.

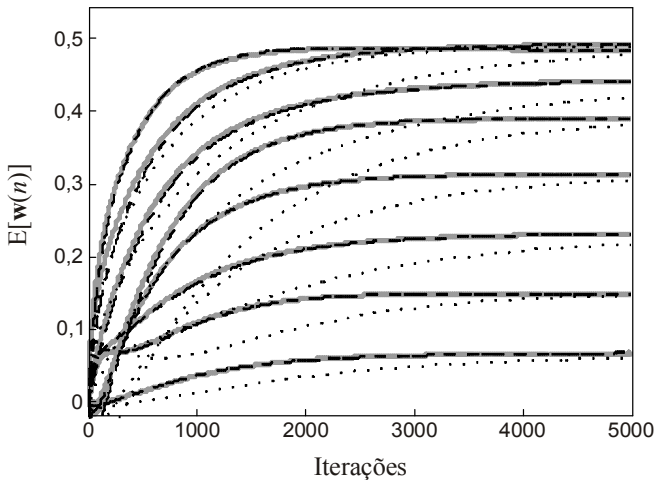


Figura 3.13: Exemplo 7. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [57]. (Linhas escuras traços-pontos) modelo de [60]. (Linhas escuras tracejadas) modelo proposto (praticamente sobreposto com [60]).

³ Cabe citar que o modelo apresentado em [69] também leva em consideração o parâmetro de regularização, porém ele é aqui desconsiderado uma vez que os resultados lá obtidos são idênticos aos de [60].

As Figuras 3.13 e 3.14 ilustram, respectivamente, o comportamento médio do vetor de coeficientes e a curva de aprendizagem do algoritmo NLMS. Constata-se que o parâmetro de regularização tem pouca influência no desempenho do algoritmo adaptativo e no seu modelo estocástico, visto que, tanto o modelo aqui proposto (com solução analítica) quanto o de [60] (o qual requer integração numérica) apresentam resultados muito semelhantes. Dessa forma, mostra-se que o parâmetro de regularização pode ser desconsiderado na derivação do modelo sem qualquer impacto significativo na sua precisão. Finalmente, é importante mencionar que tal afirmação está suportada no fato que ϵ é uma constante positiva de valor muito pequeno, como tipicamente utilizado em implementações práticas.

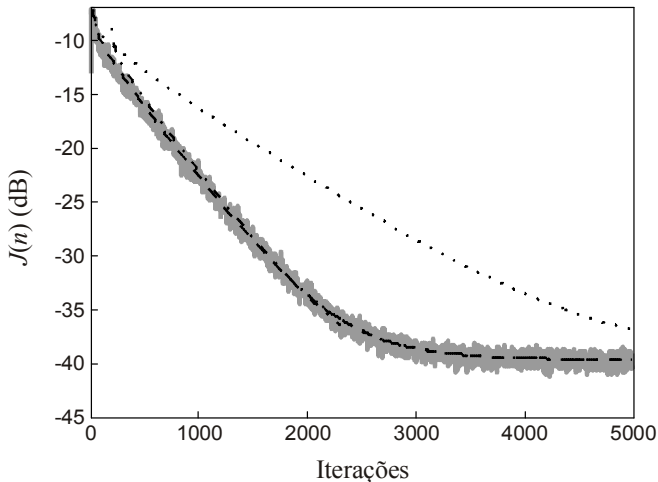


Figura 3.14: Exemplo 7. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [57]. (Linha escura traço-ponto) modelo de [60]. (Linha escura tracejada) modelo proposto (praticamente sobreposto com [60]).

Ainda referente às mesmas figuras, comparando a precisão da predição obtida pelo modelo proposto com a do modelo de [57], pode-se inferir que o erro decorrente do uso da Hipótese H1) tem impacto muito menos significativo do que o uso do PMM durante o processo de modelagem. Neste último modelo, pôde-se verificar que o erro de predição também é decorrente do fato de o sinal de entrada ser correlacionado. Ambas as parcelas (PMM e dados correlacionados)

contribuem para a baixa precisão do modelo dado em [57], fazendo com que tal modelo apresente uma menor velocidade de convergência quando comparada à simulação MC. Assim, por apresentar um desempenho mais restritivo do ponto de vista do projetista (em relação ao resultado da simulação MC), o modelo proposto em [57] é dito conservativo. Finalmente, no que tange a solução em regime permanente, é possível verificar que os modelos aqui considerados, tendem a exibir praticamente o mesmo comportamento.

3.4.4 Exemplo 8

Objetivando avaliar a eficácia do modelo proposto para filtros com maior comprimento, a ordem do filtro adaptativo é agora alterada para $M = 64$. Nesse cenário, os coeficientes do processo AR(2) são iguais a $a_1 = -0,2$ e $a_2 = 0,85$, implicando $\chi(\mathbf{R}) = 158$, o passo de adaptação utilizado no algoritmo NLMS é $\mu = 0,1$ e o parâmetro de regularização, $\varepsilon = 0$. Aqui, o modelo proposto é confrontado com os modelos de [14], [16], [57], [61], [62] e [66], sendo que apenas os resultados mais relevantes são apresentados.

Após exaustivas simulações, optou-se por omitir os modelos dados em [14], [16], [62] e [66] pelas seguintes razões:

- As abordagens utilizadas em [61] e [66] para o cálculo dos momentos normalizados são muito semelhantes, dessa forma, ambos os modelos exibem desempenho muito próximo. Assim, para uma melhor visualização, manteve-se apenas o modelo dado em [61], o qual apresentou melhor resultado com vistas ao de [66]. Cabe lembrar que, para sinais de entrada correlacionados, tal abordagem é conceitualmente incorreta.
- Comparando o modelo de [16] com o modelo proposto, pode-se observar que a diferença está no uso do PM associado à solução aproximada para as integrais hiperelípticas. Portanto, sendo o PM satisfeito, espera-se que ambos os modelos exibam um desempenho muito semelhante, não agregando valor a nossa análise.
- Os modelos dados em [14] e [62] são aqui desconsiderados por não apresentarem solução analítica para o caso de sinais de entrada correlacionados.

Na Figura 3.15, para uma melhor visualização, é ilustrado o comportamento médio de apenas quatro coeficientes do filtro

adaptativo. Esses coeficientes foram escolhidos considerando um espaçamento entre eles igual a dezesseis amostras. Exceto pelo transiente de $w_0(n)$, pode-se verificar um casamento muito bom entre os resultados de simulação MC e o do modelo proposto. Em contraste, principalmente no regime transitório, os modelos de [57] e [61] apresentam certo descasamento.

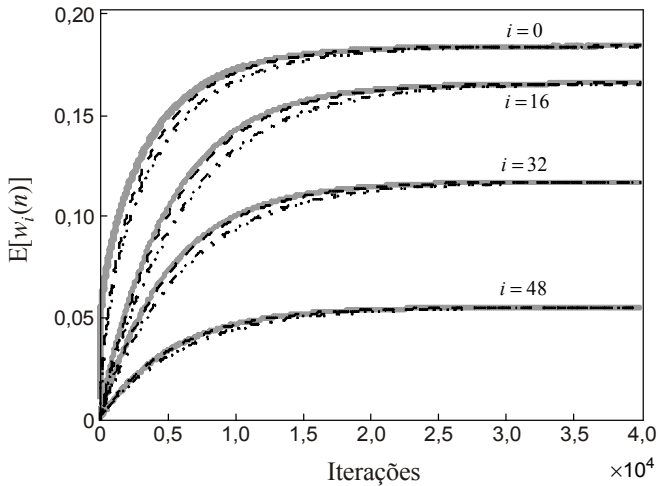


Figura 3.15: Exemplo 8. Comportamento médio dos coeficientes do filtro adaptativo. (Linhas cinzas sólidas) simulação MC. (Linhas escuras pontilhadas) modelo de [57]. (Linhas escuras traços-pontos) modelo de [61]. (Linhas escuras tracejadas) modelo proposto.

A Figura 3.16 ilustra a evolução do EQM do algoritmo NLMS obtida via simulação MC juntamente com os resultados de predição dos modelos sob análise. Tal figura revela que, dentre os modelos considerados, o modelo proposto é o que fornece maior precisão. Salienta-se que, apesar de exibir certo descasamento, o modelo de [57] merece destaque, pois ele é derivado sem o uso de hipóteses simplificativas conceitualmente incorretas.

Diante dos exemplos até então apresentados, independentemente do tipo do sinal de entrada (complexo, real, correlacionado ou não-correlacionado), pode-se afirmar que o modelo aqui proposto exibe um desempenho superior aos demais modelos encontrados na literatura que também possuem solução analítica. Tal resultado, a favor do modelo proposto, se deve ao reduzido número de aproximações utilizadas durante o processo de modelagem. Além disso, os exemplos

apresentados para o caso de sinais de entrada reais confirmam que a aproximação utilizada para obter a solução das integrais hiperelípticas proporciona resultados muito bons, principalmente, frente às abordagens comumente utilizadas na literatura.

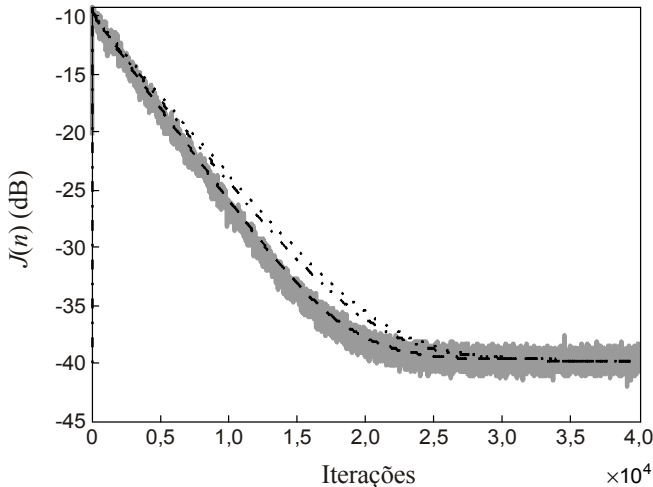


Figura 3.16: Exemplo 8. Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo de [57]. (Linha escura traço-ponto) modelo de [61]. (Linha escura tracejada) modelo proposto.

3.5 Desempenho do algoritmo em regime permanente

Nesta seção, busca-se verificar a precisão das expressões analíticas que descrevem o comportamento do algoritmo NLMS em regime permanente. Para tal, são apresentados dois exemplos contemplando as principais características do modelo proposto. O primeiro foca no caso de sinais de entrada reais não-correlacionados e o segundo, no caso de sinais de entrada complexos correlacionados. Aqui, os resultados teóricos são somente confrontados com os obtidos via simulação MC, devido à escassez de resultados conceitualmente consistentes (para os cenários adotados) [57], [61] e/ou confiáveis [70], [85] (veja Seção 2.3).

3.5.1 Exemplo 9

Considerando que o sinal de entrada seja real e não-correlacionado, baseado no Exemplo 5 (veja Seção 3.4.1) é possível avaliar a precisão das expressões obtidas para o EQM em excesso em

regime permanente e o desajuste, dadas por (2.56) e (2.60), respectivamente. Além disso, visando verificar a influência do passo de adaptação no comportamento do algoritmo, são utilizados aqui diferentes valores de passo de adaptação compreendendo a faixa de $\mu = 0,01$ a $0,5$.

Para o cenário em questão, a Figura 3.17 ilustra o EQM em excesso em regime permanente estimado via simulação MC e o predito pelo modelo proposto, para diferentes valores de passo de adaptação. Constata-se um casamento muito bom entre os resultados obtidos via simulação MC e a expressão de modelo (2.56). Da mesma forma, a Figura 3.18 mostra o desajuste obtido pelo algoritmo NLMS, verificando-se também um casamento muito bom. Nota-se ainda que, para o caso de sinais de entrada não-correlacionados, resultados satisfatórios são obtidos mesmo para valores elevados de μ .

3.5.2 Exemplo 10

Aqui, o Exemplo 3 (veja Seção 3.3.3) é reconsiderado visando avaliar a influência do parâmetro μ no comportamento do algoritmo NLMS, frente ao caso de sinais de entrada complexos e correlacionados. Assim, com o objetivo de verificar a precisão das expressões que descrevem o EQM em excesso em regime permanente e o desajuste, dados por (2.53) e (2.59), respectivamente, diferentes valores de passo de adaptação são utilizados. Vale lembrar que tais expressões foram derivadas considerando uma condição de adaptação lenta, o que permite a obtenção de uma solução aproximada para as expressões que descrevem o comportamento do algoritmo em regime permanente [veja (2.48)-(2.50)].

Para o cenário de operação considerado, a Figura 3.19 ilustra o comportamento do EQM em excesso em regime permanente como função do passo de adaptação do algoritmo NLMS. Para valores pequenos de μ , observa-se uma concordância muito boa entre os resultados obtidos via simulação MC e através do modelo proposto. Entretanto, em função das hipóteses e aproximações utilizadas na modelagem, observa-se uma ligeira degradação na precisão da expressão de modelo (2.53) à medida que o parâmetro μ é aumentado.

A Figura 3.20 ilustra o desajuste em função do passo de adaptação utilizado no algoritmo NLMS. Novamente, uma precisão muito boa é obtida principalmente para valores pequenos de passo de

adaptação. No entanto, com o aumento de μ , percebe-se certo descasamento entre os valores obtidos via simulação MC e o modelo teórico, devido principalmente às hipóteses simplificativas consideradas.

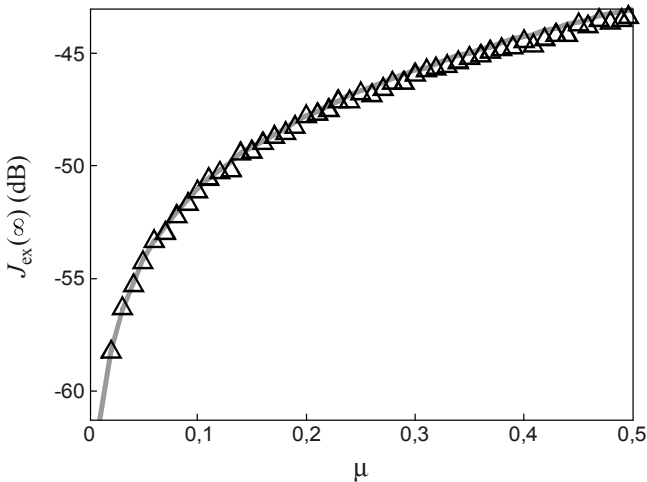


Figura 3.17: Exemplo 9. EQM em excesso em regime permanente considerando diferentes valores de μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.

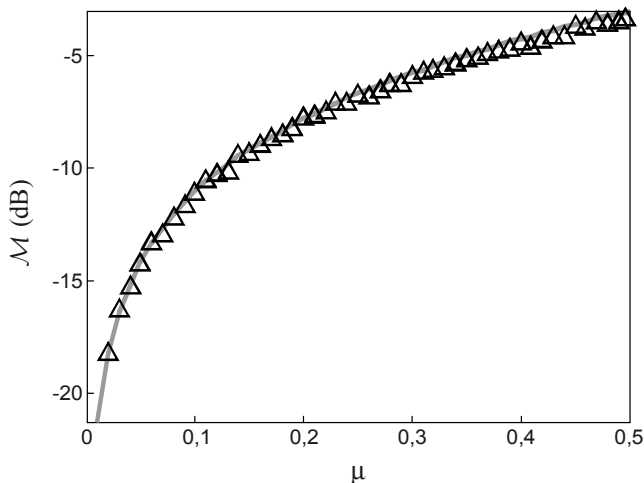


Figura 3.18: Exemplo 9. Desajuste em função do parâmetro μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.

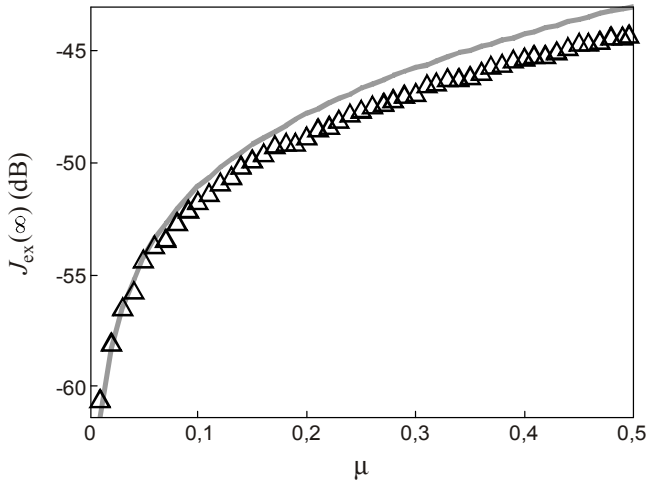


Figura 3.19: Exemplo 10. EQM em excesso em regime permanente considerando diferentes valores de μ . (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.

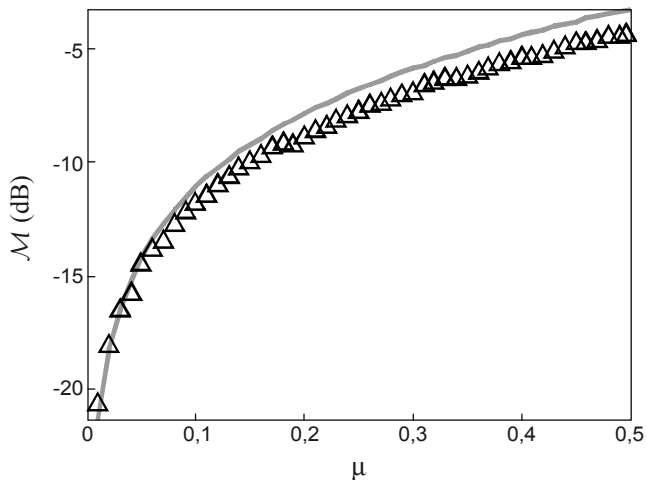


Figura 3.20: Exemplo 10. Desajuste em função do passo de adaptação. (Triângulo escuro) simulação MC. (Linha cinza sólida) modelo proposto.

Como conclusão, com base nos exemplos apresentados, verifica-se uma precisão muito boa das expressões analíticas (desenvolvidas neste trabalho) que descrevem o comportamento do

algoritmo NLMS em regime permanente, tanto no caso de sinais de entrada complexos quanto reais como também correlacionados e não-correlacionados.

3.6 Conclusões

Neste capítulo, com o intuito de avaliar a precisão do modelo estocástico proposto neste trabalho para o algoritmo NLMS, foram apresentados alguns exemplos considerando um problema de identificação de sistema com planta estacionária. A análise apresentada teve o objetivo de comparar, quando cabível, os resultados obtidos via simulação MC com o comportamento predito pelos modelos encontrados na literatura e também com o modelo proposto no presente trabalho. Como constatado, independentemente das características dos sinais de entrada e da ordem do filtro adaptativo, o modelo proposto conseguiu prever muito bem o comportamento do algoritmo NLMS tanto em regime transitório quanto em regime permanente. Tal precisão é consequência do reduzido número de hipóteses simplificativas utilizadas no desenvolvimento do modelo e da forma como o cálculo dos momentos normalizados é efetuado. Além disso, com respeito às expressões que descrevem o comportamento do algoritmo em regime permanente, os resultados obtidos ratificam a precisão das expressões analíticas aqui propostas.

CONSIDERAÇÕES FINAIS

4.1 Introdução

Neste capítulo, as considerações finais do presente trabalho de pesquisa são apresentadas. Inicialmente, é realizada uma discussão dos principais e mais relevantes resultados deste trabalho, em seguida algumas sugestões para futuros trabalhos na área de modelagem estocástica de algoritmos adaptativos são apresentadas.

4.2 Conclusões e comentários finais

Neste trabalho, foi derivado um modelo estocástico com formulação fechada para o algoritmo NLMS operando em um ambiente estacionário, considerando diferentes tipos de sinais de entrada gaussianos (complexos, reais, correlacionados e não-correlacionados). Para o desenvolvimento do modelo, apenas três hipóteses simplificativas foram consideradas, dentre as quais se destaca, como a mais restritiva, a hipótese de independência entre o vetor de coeficientes do filtro adaptativo e o vetor das amostras mais recentes do sinal de entrada (Teoria da Independência). Devido ao número reduzido de hipóteses simplificativas utilizadas, o modelo obtido é mais preciso do que os até então apresentados na literatura, sobretudo para sinais de entrada correlacionados.

A precisão do modelo proposto se deve principalmente à forma como o cálculo das matrizes de autocorrelação normalizadas é efetuado. Nesse quesito, o presente trabalho contribuiu com o desenvolvimento de soluções analíticas para o cálculo dos momentos normalizados, considerando diferentes tipos de sinais de entrada. Como consequência da forma como tais momentos são calculados, o modelo proposto é mais complexo computacionalmente do que os demais encontrados na literatura que também possuem uma formulação fechada. Entretanto, a abordagem utilizada no cálculo preserva no modelo o efeito da normalização do sinal de entrada do algoritmo adaptativo, a qual se reflete (no modelo) através da redução na dispersão dos autovalores das matrizes de autocorrelação normalizadas com vistas à matriz de

autocorrelação do sinal de entrada (veja Seção 2.5.4). Esse efeito não pode ser reproduzido no modelo quando utilizado o princípio da média (PM), visto que tal abordagem apenas escala os autovalores em função da ordem do filtro adaptativo e da potência do sinal de entrada. A redução na dispersão dos autovalores das matrizes envolvidas representa uma das principais vantagens do algoritmo NLMS sobre o algoritmo LMS para sinais de entrada correlacionados.

A partir das expressões que descrevem o comportamento médio do algoritmo NLMS e das soluções para o cálculo dos momentos normalizados, foram derivadas expressões que descrevem o comportamento do algoritmo NLMS em regime permanente. Particularmente, soluções analíticas para duas figuras de mérito foram determinadas, a saber, o erro quadrático médio (EQM) em excesso em regime permanente e o desajuste (veja Seção 2.5). Tais expressões permitem ao projetista inferir sobre o funcionamento do sistema adaptativo, possibilitando determinar relações de causa e efeito entre os parâmetros de controle do algoritmo. Além disso, considerando sinais de entrada não-correlacionados, foram desenvolvidas expressões que relacionam os algoritmos LMS e NLMS através do passo de adaptação, permitindo ajustá-los para se obter uma mesma velocidade de convergência ou um mesmo EQM em regime permanente (desajuste).

Os resultados de simulação apresentados corroboram com o fato de que o modelo proposto é bastante preciso, independentemente da ordem do filtro adaptativo ou do tipo do sinal de entrada. Observou-se, nos exemplos considerados, um casamento muito bom entre o comportamento descrito pelo modelo proposto e o obtido via simulação Monte Carlo (MC), tanto em regime transitório quanto em regime permanente (veja Capítulo 3). Quando cabível, o modelo proposto foi confrontado com os demais modelos disponíveis na literatura obtidos para o algoritmo NLMS, constatando-se em todas as situações avaliadas uma muito boa precisão, principalmente, frente aos modelos desenvolvidos considerando o PM ou o princípio da média modificado (PMM) (veja Seção 2.3).

Finalmente, destaca-se que o presente trabalho fornece novas ferramentas para a análise de outros algoritmos adaptativos que também consideram sinais de entrada normalizados e possibilita, ainda, melhorar a precisão de modelos existentes obtidos através de outras estratégias, tais como o princípio de conservação de energia e a equação de diferenças estocásticas, as quais também requerem o cálculo de momentos normalizados com formas similares às apresentadas para o algoritmo NLMS. Em resumo, diante do até então apresentado,

acredita-se que o presente trabalho cumpriu com os objetivos estabelecidos (veja Seção 1.3), fomentando ainda novas perspectivas de pesquisa na área.

4.3 Sugestões para trabalhos futuros

Um grande número de trabalhos de pesquisa podem ainda ser desenvolvidos sobre a modelagem de algoritmos adaptativos. A seguir, algumas sugestões para trabalhos futuros são apresentadas.

- Estender o modelo proposto para o caso de planta não-estacionária, derivando assim novas expressões para o EQM em excesso e para o desajuste. A partir de tais expressões, determinar o passo de adaptação para mínimo EQM em excesso e a relação entre o grau de não-estacionariedade e o desajuste.
 - Aprimorar o modelo proposto visando contemplar o caso de ruído de medição variante no tempo (não-estacionário).
 - Generalizar a expressão que relaciona a dispersão de autovalores da matriz de autocorrelação do sinal de entrada com sua versão normalizada, aprofundando assim o estudo desse efeito nas demais matrizes de autocorrelação normalizadas.
 - Utilizar as ferramentas aqui desenvolvidas para a modelagem de outros algoritmos adaptativos que consideram sinal de entrada normalizado, tais como o algoritmo TD-LMS, os algoritmos adaptativos em sub-banda, os algoritmos da classe proporcional (PNLMS) e de passo variável (VSS-LMS).
-

CÁLCULO DE H

Neste apêndice, considerando que o sinal de entrada tem função densidade de probabilidade gaussiana, uma metodologia para o cálculo da matriz de autocorrelação normalizada [definida em (2.1)] é apresentada. Tal metodologia utiliza como base a técnica proposta em [14], a qual foi posteriormente estendida para o caso de sinais de entrada complexos em [19], [20], [23]. Especificamente para o caso de sinais de entrada reais, visando à obtenção de uma solução analítica, a aproximação proposta em [15]-[18], [21], [22] é aqui utilizada.

Aplicando a definição de valor esperado¹ em (2.1), obtém-se

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] = \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \quad (\text{A.1})$$

onde $f_{\mathbf{x}}(\mathbf{x})$ representa a função densidade de probabilidade do vetor $\mathbf{x}(n)$ (o qual contém as amostras do sinal de entrada) e M , a ordem do filtro adaptativo. Para determinar (A.1), de forma similar a [14], [23], [62], uma função auxiliar² é definida como

$$\mathbf{F}_1(\omega) = \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{\mathbf{xx}^H}{\mathbf{x}^H\mathbf{x}} e^{-\omega(\mathbf{x}^H\mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \quad (\text{A.2})$$

¹ O valor esperado de uma função $\varphi(z)$ (denotado $E[\varphi(z)]$) é definido como $E[\varphi(z)] = \int_{-\infty}^{\infty} \varphi(z) f_z(z) dz$, sendo $f_z(z)$ a função densidade de probabilidade de z . Tal operação, quando aplicada a um vetor ou matriz, deve ser interpretada como o valor esperado de cada elemento, resultando em um vetor ou matriz de mesma dimensão [25].

² A função auxiliar é construída de forma que, quando derivada, o termo presente no denominador dessa função (o fator de normalização) seja eliminado. Cabe ressaltar que a função auxiliar pode apresentar formas diferentes das exibidas aqui (veja [82], [89]).

sendo o índice de tempo n aqui desconsiderado por simplicidade de notação. Note que, fazendo $\omega=0$ em (A.2), obtém-se (A.1), isto é,

$$E \left[\frac{\mathbf{x}(n)\mathbf{x}^H(n)}{\mathbf{x}^H(n)\mathbf{x}(n)} \right] = \mathbf{F}_1(\omega) \Big|_{\omega=0} = \mathbf{F}_1(0). \quad (\text{A.3})$$

Logo, diferenciando (A.2) em relação à ω , o termo presente no denominador é eliminado, resultando em

$$\frac{\partial \mathbf{F}_1(\omega)}{\partial \omega} = - \underbrace{\int \dots \int}_{M \text{ integrais}} \mathbf{xx}^H e^{-\omega(\mathbf{x}^H \mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x}. \quad (\text{A.4})$$

A partir deste ponto, a derivação requer o conhecimento de $f_{\mathbf{x}}(\mathbf{x})$. Dessa forma, visto que as funções densidades de probabilidades gaussianas complexa e real apresentam formas distintas [veja (A.5) e (A.19)] [25], soluções particulares para cada caso devem ser obtidas. Assim, visando tornar evidente tal diferença, os símbolos \mathbb{C} e \mathbb{R} serão utilizados nas soluções relativas ao caso de sinais de entrada complexos e reais, respectivamente.

A.1 Sinais de entrada gaussianos complexos

Nesta seção, considera-se que o sinal de entrada seja complexo e tenha distribuição de probabilidade gaussiana. Tal consideração implica o uso da função densidade de probabilidade gaussiana complexa, a qual é dada por

$$f_{\mathbf{x}}^{\mathbb{C}}(\mathbf{x}) = \frac{1}{\pi^M \det(\mathbf{C})} e^{-[(\mathbf{x}-\mathbf{m})^H \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})]} \quad (\text{A.5})$$

onde \mathbf{C} denota a matriz de autocovariância de \mathbf{x} , $\det(\cdot)$, o operador determinante e \mathbf{m} , um vetor contendo o valor esperado (média) de \mathbf{x} [25].

Agora, substituindo (A.5) em (A.4) e considerando que \mathbf{x} tem média zero (o que permite a substituição da matriz de autocovariância \mathbf{C} pela matriz de autocorrelação \mathbf{R}), obtém-se

$$\frac{\partial \mathbf{F}_1^{\mathbb{C}}(\omega)}{\partial \omega} = - \frac{1}{\pi^M \det(\mathbf{R})} \underbrace{\int \dots \int}_{M \text{ integrais}} \mathbf{xx}^H e^{-[\mathbf{x}^H(\omega \mathbf{I} + \mathbf{R}^{-1})\mathbf{x}]} d\mathbf{x} \quad (\text{A.6})$$

com \mathbf{I} representando a matriz identidade. Manipulando algebricamente os termos de (A.6), pode-se reescrever (A.6) como segue:

$$\frac{\partial \mathbf{F}_1^{\mathbb{C}}(\omega)}{\partial \omega} = -\frac{\det[\mathbf{L}_{\mathbb{C}}(\omega)]}{\det(\mathbf{R})} \mathbf{G}_1^{\mathbb{C}}(\omega) \quad (\text{A.7})$$

onde

$$\mathbf{L}_{\mathbb{C}}^{-1}(\omega) = (\mathbf{I} + \omega \mathbf{R}) \mathbf{R}^{-1} \quad (\text{A.8})$$

e

$$\mathbf{G}_1^{\mathbb{C}}(\omega) = \frac{1}{\pi^M \det[\mathbf{L}_{\mathbb{C}}(\omega)]} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{x} \mathbf{x}^H e^{-[\mathbf{x}^H \mathbf{L}_{\mathbb{C}}^{-1}(\omega) \mathbf{x}]} d\mathbf{x}. \quad (\text{A.9})$$

Observe que, por definição, (A.9) é a correlação cruzada entre x_i e x_j^* , quando essas variáveis aleatórias são conjuntamente gaussianas com matriz de autocorrelação $\mathbf{L}_{\mathbb{C}}(\omega)$ [2], [25]. Assim, fazendo $\mathbf{G}_1^{\mathbb{C}}(\omega) = \mathbf{L}_{\mathbb{C}}(\omega)$ em (A.7), utilizando as propriedades do determinante³, (2.14) e integrando ambos os lados da expressão resultante com respeito à ω , obtém-se

$$\mathbf{F}_1^{\mathbb{C}}(\omega) = \mathbf{Q} \mathbf{H} \mathbf{Q}^H \quad (\text{A.10})$$

com \mathbf{H} denotando uma matriz diagonal cujos elementos são dados por

$$h_{i,i}^{\mathbb{C}} = -\int \frac{\lambda_i}{(1 + \omega \lambda_i) \prod_{k=1}^M (1 + \omega \lambda_k)} d\omega + C_1^{\mathbb{C}} \quad (\text{A.11})$$

e $C_1^{\mathbb{C}}$ uma constante obtida através da seguinte condição:

$$C_1^{\mathbb{C}} = \lim_{\omega \rightarrow \infty} \mathbf{F}_1^{\mathbb{C}}(\omega) = 0.$$

Conforme discutido na literatura, considerando o caso mais geral, a integral presente em (A.11) não possui solução analítica [84]. Isso se deve, principalmente, à variedade de formas que o polinômio do denominador pode tomar. Assim, para prosseguir com o desenvolvimento aqui apresentado, restrições devem ser aplicadas as raízes de (A.11) de maneira similar a [14], [22], [23], [53], [60], [62], [65].

³ As propriedades utilizadas são: $\det(\mathbf{A}\mathbf{B}) = \det(\mathbf{A})\det(\mathbf{B})$; $\det(\mathbf{A}^{-1}) = [\det(\mathbf{A})]^{-1}$ [2], [15], [30], [53].

A.1.1 Sinais de entrada correlacionados

Assumindo que a matriz de autocorrelação do sinal de entrada possua somente autovalores distintos (situação comumente encontrada na prática [18], [20]-[23]), a integral de (A.11) pode ser calculada utilizando a técnica de expansão em frações parciais. Para tal, pode-se reescrever (A.11) como

$$h_{i,i}^{\text{C}} = -\frac{1}{G^{\text{C}}} \int \frac{1}{(\omega - \omega_i^{\text{C}}) \prod_{k=1, k \neq i}^M (\omega - \omega_k^{\text{C}})} d\omega \quad (\text{A.12})$$

onde

$$\omega_k^{\text{C}} = -\frac{1}{\lambda_k}, \quad k = 1, 2, \dots, M \quad (\text{A.13})$$

e

$$G^{\text{C}} = \prod_{k=1}^M \lambda_k. \quad (\text{A.14})$$

Vale salientar que a constante de ganho G^{C} surge devido ao fato de o polinômio do denominador de (A.11) não ser mônico [86].

Logo, considerando apenas raízes distintas em (A.12), utilizando a técnica de expansão em frações parciais, integrando termo a termo a expressão resultante [84] e fazendo $\omega = 0$, obtém-se

$$h_{i,i}^{\text{C}} = -\frac{1}{G^{\text{C}}} \left[\frac{A_{1,i}^{\text{C}}}{\omega_i^{\text{C}}} - \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^{\text{C}} \log(-\omega_l^{\text{C}}) + \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^{\text{C}} \log(-\omega_i^{\text{C}}) \right] \quad (\text{A.15})$$

com

$$A_{1,i}^{\text{C}} = \frac{1}{\prod_{\substack{k=1 \\ k \neq i}}^M (\omega_i^{\text{C}} - \omega_k^{\text{C}})} \quad (\text{A.16})$$

e

$$B_{l,i}^{\text{C}} = \frac{1}{(\omega_l^{\text{C}} - \omega_i^{\text{C}}) \prod_{\substack{k=1 \\ k \neq l}}^M (\omega_l^{\text{C}} - \omega_k^{\text{C}})}. \quad (\text{A.17})$$

De (A.15), a partir de uma das propriedades de logaritmo⁴, segue (2.15).

⁴ $\log_a(AB^{-1}) = \log_a(A) - \log_a(B)$

A.1.2 Sinais de entrada não-correlacionados

Na análise estatística de algoritmos adaptativos, conhecer o comportamento do algoritmo para o caso de sinais de entrada não-correlacionados tem grande relevância, mesmo sabendo que tal condição não seja realista na maioria dos casos práticos. Tais resultados, muitas vezes, possibilitam estabelecer relações de causa e efeito entre os parâmetros de controle do algoritmo e as métricas de desempenho usualmente utilizadas. Nesse tipo de cenário, a matriz de autocorrelação do sinal de entrada pode ser escrita como

$$\mathbf{R} = \sigma_x^2 \mathbf{I} \quad (\text{A.18})$$

implicando que todos os seus autovalores são iguais a σ_x^2 . Assim, fazendo $\lambda_k = \lambda_i = \sigma_x^2$ em (A.11), e resolvendo a expressão resultante para $\omega = 0$ [84], obtém-se (2.16).

A.2 Sinais de entrada gaussianos reais

Considerando que o sinal de entrada do filtro adaptativo seja agora gaussiano real, com função densidade de probabilidade definida como [25]

$$f_{\mathbf{x}}^{\mathbb{R}}(\mathbf{x}) = \frac{1}{(2\pi)^{M/2} \sqrt{\det(\mathbf{C})}} e^{-\frac{1}{2}[(\mathbf{x}-\mathbf{m})^T \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})]} \quad (\text{A.19})$$

é possível prosseguir com a derivação da solução da matriz de autocorrelação normalizada, seguindo um procedimento semelhante ao proposto em [14], [18], [22], [62]. Assim, substituindo (A.19) em (A.4), a expressão resultante pode ser reescrita como

$$\frac{\partial \mathbf{F}_1^{\mathbb{R}}(\omega)}{\partial \omega} = -\sqrt{\frac{\det[\mathbf{L}_{\mathbb{R}}(\omega)]}{\det(\mathbf{R})}} \mathbf{G}_1^{\mathbb{R}}(\omega) \quad (\text{A.20})$$

onde

$$\mathbf{L}_{\mathbb{R}}^{-1}(\omega) = (\mathbf{I} + 2\omega\mathbf{R})\mathbf{R}^{-1} \quad (\text{A.21})$$

e

$$\mathbf{G}_1^{\mathbb{R}}(\omega) = \frac{1}{(2\pi)^{M/2} \sqrt{\det[\mathbf{L}_{\mathbb{R}}(\omega)]}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{x}\mathbf{x}^T e^{-\frac{1}{2}[\mathbf{x}^T \mathbf{L}_{\mathbb{R}}^{-1}(\omega)\mathbf{x}]} d\mathbf{x}. \quad (\text{A.22})$$

A expressão (A.22) pode ser vista como a correlação cruzada entre x_i e x_j , sendo essas variáveis aleatórias conjuntamente

gaussianas com matriz de autocorrelação $\mathbf{L}_{\mathbb{R}}(\omega)$ [2], [25]. Dessa forma, considerando $\mathbf{G}_1^{\mathbb{R}}(\omega) = \mathbf{L}_{\mathbb{R}}(\omega)$ em (A.20), utilizando as propriedades do determinante, a autodecomposição de \mathbf{R} e integrando ambos os lados da expressão resultante em relação a ω , obtém-se

$$\mathbf{F}_1^{\mathbb{R}}(\omega) = \mathbf{Q}\mathbf{H}\mathbf{Q}^T \quad (\text{A.23})$$

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

$$h_{i,i}^{\mathbb{R}} = -\int \frac{\lambda_i}{(1 + 2\omega\lambda_i) \sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega + C_1^{\mathbb{R}} \quad (\text{A.24})$$

$$\text{e } C_1^{\mathbb{R}} = \lim_{\omega \rightarrow \infty} \mathbf{F}_1^{\mathbb{R}}(\omega) = 0.$$

Na literatura especializada, quando a ordem do polinômio do denominador é maior do que quatro, a integral definida em (A.24) é denominada hiperelíptica ou abeliana [87], [88]. Tal integral, exceto em casos muito particulares [84], não apresenta solução analítica, devido principalmente à presença do operador raiz quadrada [15], [16], [22], [24], [62]. Assim, de acordo com [14], [62], [69], a solução de (A.24) envolve geralmente o uso de técnicas de integração numérica [63]. Contudo, aplicando restrições às características do sinal de entrada (de forma similar a [14], [62], [65]), tem-se abordagens que podem ser utilizadas para resolver a integral em questão [15], [16], [22]. A seguir, duas situações de grande interesse são tratadas, a saber: os casos de sinais de entrada correlacionados e não-correlacionados.

A.2.1 Sinais de entrada correlacionados

Mesmo assumindo que os autovalores de \mathbf{R} sejam distintos, a técnica de expansão em frações parciais não pode ser aplicada diretamente em (A.24) (de maneira análoga ao que foi utilizado para o caso de sinais de entrada gaussianos complexos), uma vez que tal técnica requer que a função a ser expandida seja racional. Além disso, outro fator complicador é que, em aplicações práticas, a ordem do filtro adaptativo geralmente é muito superior a quatro, inviabilizando a solução analítica da integral de (A.24) e tornando ainda a solução numérica computacionalmente custosa. Nesse contexto, visando à obtenção de uma formulação fechada para (A.24), o procedimento aproximado proposto em [16] e aprimorado em [15] e [22], é aqui considerado. Tal procedimento consiste em substituir pares de raízes

adjacentes de $h_{i,i}^{\mathbb{R}}$ pela sua média geométrica (agora com multiplicidade dois). Para ilustrar o procedimento, denota-se o polinômio do denominador de (A.24) por $P^{\mathbb{R}}(\omega)$, isto é,

$$P^{\mathbb{R}}(\omega) = (1 + 2\omega\lambda_i) \sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)} \quad (\text{A.25})$$

cujas raízes são

$$\omega_k^{\mathbb{R}} = -\frac{1}{2\lambda_k}, \quad k = 1, 2, \dots, M. \quad (\text{A.26})$$

Como \mathbf{R} é quase sempre uma matriz definida positiva [11], seus autovalores são maiores do que zero, implicando que (A.26) assume apenas valores negativos. Dessa forma, sem perda de generalidade, (A.25) pode ser reescrita como

$$P^{\mathbb{R}}(\omega) = 2\lambda_i(\omega - \omega_i^{\mathbb{R}}) \sqrt{G^{\mathbb{R}} \prod_{k=1}^M (\omega - \omega_k^{\mathbb{R}})} \quad (\text{A.27})$$

sendo

$$G^{\mathbb{R}} = 2^M \prod_{k=1}^M \lambda_k \quad (\text{A.28})$$

decorrente do fato de que $P^{\mathbb{R}}(\omega)$ não é um polinômio mônico [86].

Para prosseguir com o desenvolvimento, a raiz quadrada presente em (A.27) deve ser eliminada. Para tal, utilizando a relação [15]-[18], [21], [89]

$$\tilde{\omega}_q^{\mathbb{R}} = -\sqrt{\omega_{(2q-1)}^{\mathbb{R}} \omega_{2q}^{\mathbb{R}}}, \quad q = 1, 2, \dots, M/2 \text{ com } M \text{ par}, \quad (\text{A.29})$$

pares de raízes adjacentes são substituídos pela sua média geométrica, resultando em

$$P^{\mathbb{R}}(\omega) \cong 2\lambda_i(\omega - \omega_i^{\mathbb{R}}) \sqrt{G^{\mathbb{R}}} \sqrt{\prod_{q=1}^{M/2} (\omega - \tilde{\omega}_q^{\mathbb{R}})^2}. \quad (\text{A.30})$$

Agora, cancelando a raiz quadrada com o expoente quadrático das raízes em (A.30), um polinômio aproximado é obtido como

$$P^{\mathbb{R}}(\omega) \cong 2\lambda_i(\omega - \omega_i^{\mathbb{R}}) \sqrt{G^{\mathbb{R}}} \left| \prod_{q=1}^{M/2} (\omega - \tilde{\omega}_q^{\mathbb{R}}) \right|. \quad (\text{A.31})$$

Por fim, substituindo (A.31) em (A.24), desconsiderando o operador valor absoluto [15] (visto que tem-se interesse somente na solução para $\omega=0$), realizando a expansão em frações parciais do argumento da expressão resultante e resolvendo para $\omega=0$ [84], tem-se

$$h_{i,i}^{\mathbb{R}} \cong -\frac{1}{2\sqrt{G^{\mathbb{R}}}} \left[A_{1,i}^{\mathbb{R}} \log(-\omega_i^{\mathbb{R}}) + \sum_{l=1}^{M/2} B_{l,i}^{\mathbb{R}} \log(-\tilde{\omega}_l^{\mathbb{R}}) \right] \quad (\text{A.32})$$

onde

$$A_{1,i}^{\mathbb{R}} = \frac{1}{\prod_{q=1}^{M/2} (\omega_i^{\mathbb{R}} - \tilde{\omega}_q^{\mathbb{R}})} \quad (\text{A.33})$$

e

$$B_{l,i}^{\mathbb{R}} = \frac{1}{(\tilde{\omega}_l^{\mathbb{R}} - \omega_i^{\mathbb{R}}) \prod_{\substack{q=1 \\ q \neq l}}^{M/2} (\tilde{\omega}_l^{\mathbb{R}} - \tilde{\omega}_q^{\mathbb{R}})}. \quad (\text{A.34})$$

A partir de (A.32), observa-se que os elementos de \mathbf{H} podem ser calculados de forma analítica, ou seja, sem a necessidade do emprego de técnicas de integração numérica [63]. Salienta-se ainda que a solução proposta, mesmo não sendo exata, apresenta resultados muito superiores aos obtidos considerando as aproximações comumente utilizadas na derivação de modelos para o algoritmo NLMS (sobretudo, quando a ordem do filtro adaptativo é alta [15]).

A.2.2 Sinais de entrada não-correlacionados

Assumindo que o sinal de entrada seja branco, os autovalores da matriz de autocorrelação são todos iguais a σ_x^2 . Assim, fazendo $\lambda_i = \lambda_k = \sigma_x^2$ em (A.24), assumindo M par e resolvendo a integral da expressão resultante para $\omega=0$ [84], obtém-se (2.16).

CÁLCULO DE T E P

Neste apêndice, o cálculo do momento de quarta ordem normalizado [definido em (2.31)] é tratado considerando que o sinal de entrada tenha distribuição de probabilidade gaussiana. O procedimento utilizado aqui se baseia na técnica proposta em [14], a qual foi estendida para o caso de sinais complexos em [19], [20], [23]. No caso de sinais de entrada complexos, tal procedimento resulta em uma solução exata para o valor esperado em questão (sendo essa uma das contribuições do presente trabalho). No que tange à solução para sinais de entrada reais, visando à obtenção de uma formulação fechada, a abordagem aproximada desenvolvida em [15]-[18], [21], [22] é aqui utilizada.

Para determinar o valor esperado dado pelo termo E de (2.31), desconsiderando o índice de tempo n por simplicidade de notação, define-se inicialmente a seguinte função auxiliar [14], [23], [62]:

$$\mathbf{F}_2(\omega) = \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{\mathbf{x}\mathbf{x}^H \mathbf{v}\mathbf{v}^H \mathbf{x}\mathbf{x}^H}{(\mathbf{x}^H \mathbf{x})^2} e^{-\omega(\mathbf{x}^H \mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x}. \quad (\text{B.1})$$

tal que, para $\omega = 0$, (B.1) retorne ao valor esperado requerido, isto é,

$$\mathbb{E} \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n) \mathbf{v}(n)\mathbf{v}^H(n) \mathbf{x}(n)\mathbf{x}^H(n)}{[\mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\} = \mathbf{F}_2(\omega) \Big|_{\omega=0} = \mathbf{F}_2(0). \quad (\text{B.2})$$

Agora, diferenciando (B.1) duas vezes com respeito à ω , obtém-se

$$\frac{\partial^2 \mathbf{F}_2(\omega)}{\partial \omega^2} = \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{x}\mathbf{x}^H \mathbf{v}\mathbf{v}^H \mathbf{x}\mathbf{x}^H e^{-\omega(\mathbf{x}^H \mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x}. \quad (\text{B.3})$$

Novamente, para prosseguir com o desenvolvimento, o conhecimento de $f_{\mathbf{x}}(\mathbf{x})$ é requerido. Dessa forma, a solução passa a ser particular dependendo do tipo do sinal de entrada considerado.

B.1 Sinais de entrada gaussianos complexos

Neste contexto, substituindo a função densidade de probabilidade gaussiana complexa, dada por (A.5), em (B.3) e considerando que o sinal de entrada tem média zero, a expressão resultante pode ser reescrita como

$$\frac{\partial^2 \mathbf{F}_2^{\mathbb{C}}(\omega)}{\partial \omega^2} = \frac{\det[\mathbf{L}_{\mathbb{C}}(\omega)]}{\det(\mathbf{R})} \mathbf{G}_2^{\mathbb{C}}(\omega) \quad (\text{B.4})$$

com $\mathbf{L}_{\mathbb{C}}^{-1}(\omega)$ dada por (A.8) e

$$\mathbf{G}_2^{\mathbb{C}}(\omega) = \frac{1}{\pi^M \det[\mathbf{L}_{\mathbb{C}}(\omega)]} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{x} \mathbf{x}^H \mathbf{v} \mathbf{v}^H \mathbf{x} \mathbf{x}^H e^{-[\mathbf{x}^H \mathbf{L}_{\mathbb{C}}^{-1}(\omega) \mathbf{x}]} d\mathbf{x}. \quad (\text{B.5})$$

Considerando que (B.5) é, por definição, um momento de quarta ordem de uma variável gaussiana ponderado por $\mathbf{K}(n)$ [2], [10] e utilizando a decomposição (2.14) bem como as propriedades do traço e do determinante, tem-se

$$\begin{aligned} \frac{\partial^2 \mathbf{F}_2^{\mathbb{C}}(\omega)}{\partial \omega^2} = \mathbf{Q} \left[\prod_{k=1}^M (1 + \omega \lambda_k) \right]^{-1} & \left([\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}] \mathbf{K}'(n) [\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}] \right. \\ & \left. + [\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}] \text{tr}\{\mathbf{K}'(n) [\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}]\} \right) \mathbf{Q}^H \end{aligned} \quad (\text{B.6})$$

sendo $\mathbf{K}'(n)$ definida em (2.25) e o fator ζ , igual a um. Logo, integrando ambos os lados de (B.6) duas vezes em relação à ω , obtém-se

$$\begin{aligned} \mathbf{F}_2^{\mathbb{C}}(\omega) = \mathbf{Q} \left(\iint \frac{[\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}] \mathbf{K}'(n) [\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}]}{\prod_{k=1}^M (1 + \omega \lambda_k)} d\omega \right. \\ \left. + \iint \frac{[\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}] \text{tr}\{\mathbf{K}'(n) [\mathbf{\Lambda}(\mathbf{I} + \omega \mathbf{\Lambda})^{-1}]\}}{\prod_{k=1}^M (1 + \omega \lambda_k)} d\omega \right) \mathbf{Q}^H. \end{aligned} \quad (\text{B.7})$$

Visto que somente os elementos da diagonal de $\mathbf{K}'(n)$ são considerados em (2.32), e lembrando que o resultado aqui obtido é pré- e pós-multiplicado por \mathbf{Q}^H e \mathbf{Q} , respectivamente, visando prosseguir com o desenvolvimento, (B.7) é reescrita como segue:

$$\mathbf{f}_2^{\mathbb{C}}(\omega) = (\mathbf{T} + \mathbf{P}) \mathbf{k}'(n) \quad (\text{B.8})$$

com $\mathbf{f}_2^{\text{C}}(\omega)$ representando os elementos da diagonal de $\mathbf{Q}^{\text{H}}\mathbf{F}_2^{\text{C}}(\omega)\mathbf{Q}$ e $\mathbf{k}'(n)$, os elementos da diagonal de $\mathbf{K}'(n)$. A matriz diagonal \mathbf{T} tem seus elementos dados por

$$t_{i,i}^{\text{C}} = \iint \frac{\lambda_i^2}{(1 + \omega\lambda_i)^2 \prod_{k=1}^M (1 + \omega\lambda_k)} d\omega + C_2^{\text{C}} \quad (\text{B.9})$$

e a matriz simétrica \mathbf{P} por

$$p_{i,j}^{\text{C}} = \iint \frac{\lambda_i \lambda_j}{(1 + \omega\lambda_i)(1 + \omega\lambda_j) \prod_{k=1}^M (1 + \omega\lambda_k)} d\omega + C_2^{\text{C}} \quad (\text{B.10})$$

sendo $C_2^{\text{C}} = \lim_{\omega \rightarrow \infty} \mathbf{F}_2^{\text{C}}(\omega) = 0$. Portanto, o momento de quarta ordem definido em (B.2) é completamente caracterizado se a solução de (B.9) e (B.10) for conhecida. No entanto, não foi encontrada na literatura uma expressão fechada para o cálculo da integral em questão [84]. Assim, para seguir com a derivação, restrições precisam ser consideradas sobre as raízes de (B.9) e (B.10).

B.1.1 Sinais de entrada correlacionados

Considerando que (B.9) e (B.10) possuem somente raízes reais e distintas (caso muito comum para sinais correlacionados [18], [20]-[23]), uma solução fechada pode ser obtida através da técnica de expansão em frações parciais. Assim, com o objetivo de evidenciar as raízes de $t_{i,i}^{\text{C}}$ e $p_{i,j}^{\text{C}}$, (B.9) e (B.10) são, respectivamente, reescritas como

$$t_{i,i}^{\text{C}} = \frac{1}{G^{\text{C}}} \iint \frac{1}{(\omega - \omega_i^{\text{C}})^2 \prod_{k=1}^M (\omega - \omega_k^{\text{C}})} d\omega \quad (\text{B.11})$$

e

$$p_{i,j}^{\text{C}} = \frac{1}{G^{\text{C}}} \iint \frac{1}{(\omega - \omega_i^{\text{C}})(\omega - \omega_j^{\text{C}}) \prod_{k=1}^M (\omega - \omega_k^{\text{C}})} d\omega \quad (\text{B.12})$$

com G^{C} dado por (A.14) e ω_k^{C} por (A.13). Então, assumindo raízes distintas, realizando a expansão em frações parciais de (B.11) e (B.12), e resolvendo a expressão resultante para $\omega = 0$ [84], de (B.11), obtém-se

$$t_{i,i}^{\text{C}} = \frac{1}{G^{\text{C}}} \left(\frac{-A_{i,i}^{\text{C}}}{2\omega_i^{\text{C}}} + \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^{\text{C}} \log(-\omega_i^{\text{C}}) \right. \\ \left. + \sum_{\substack{l=1 \\ l \neq i}}^M \frac{B_{l,i}^{\text{C}} \{ \omega_l^{\text{C}} [1 - \log(-\omega_l^{\text{C}})] - \omega_i^{\text{C}} [1 - \log(-\omega_i^{\text{C}})] \}}{(\omega_l^{\text{C}} - \omega_i^{\text{C}})} \right) \quad (\text{B.13})$$

e, a partir de (B.12), assumindo $i \neq j$, tem-se

$$p_{i,j}^{\text{C}} = \frac{1}{G^{\text{C}}} \left\{ -\frac{A_{i,i}^{\text{C}} \log(-\omega_i^{\text{C}})}{(\omega_i^{\text{C}} - \omega_j^{\text{C}})} - B_{j,i}^{\text{C}} \log(-\omega_j^{\text{C}}) + \frac{2B_{j,i}^{\text{C}} \omega_i^{\text{C}} [1 - \log(-\omega_i^{\text{C}})]}{(\omega_j^{\text{C}} - \omega_i^{\text{C}})} \right. \\ + \sum_{\substack{l=1 \\ l \neq i,j}}^M \frac{\omega_l^{\text{C}} [1 - \log(-\omega_l^{\text{C}})]}{(\omega_l^{\text{C}} - \omega_i^{\text{C}})(\omega_l^{\text{C}} - \omega_j^{\text{C}}) \prod_{\substack{k=1 \\ k \neq l}}^M (\omega_l^{\text{C}} - \omega_k^{\text{C}})} - \sum_{\substack{l=1 \\ l \neq i,j}}^M \frac{B_{l,i}^{\text{C}} \omega_i^{\text{C}} [1 - \log(-\omega_i^{\text{C}})]}{(\omega_l^{\text{C}} - \omega_j^{\text{C}})} \\ + \sum_{\substack{l=1 \\ l \neq i,j}}^M \frac{\omega_i^{\text{C}} [1 - \log(-\omega_i^{\text{C}})]}{(\omega_l^{\text{C}} - \omega_j^{\text{C}})(\omega_i^{\text{C}} - \omega_j^{\text{C}})^2 \prod_{\substack{k=1 \\ k \neq i,l}}^M (\omega_l^{\text{C}} - \omega_k^{\text{C}})} + \frac{2A_{i,j}^{\text{C}} \omega_j^{\text{C}} [1 - \log(-\omega_j^{\text{C}})]}{(\omega_j^{\text{C}} - \omega_i^{\text{C}})^2} \\ \left. + \sum_{\substack{l=1 \\ l \neq i,j}}^M \frac{\omega_j^{\text{C}} [1 - \log(-\omega_j^{\text{C}})]}{(\omega_j^{\text{C}} - \omega_i^{\text{C}})^2 (\omega_l^{\text{C}} - \omega_i^{\text{C}}) \prod_{\substack{k=1 \\ k \neq j,l}}^M (\omega_l^{\text{C}} - \omega_k^{\text{C}})} - \sum_{\substack{l=1 \\ l \neq i,j}}^M \frac{B_{l,i}^{\text{C}} \omega_j^{\text{C}} [1 - \log(-\omega_j^{\text{C}})]}{(\omega_l^{\text{C}} - \omega_j^{\text{C}})} \right\} \quad (\text{B.14})$$

onde $A_{i,i}^{\text{C}}$ e $B_{l,i}^{\text{C}}$ são, respectivamente, dados por (A.16) e (A.17). Finalmente, agrupando os termos comuns, considerando (A.16), (A.17) e as propriedades básicas de logaritmo, de (B.13) e (B.14) obtém-se (2.36) e (2.37), respectivamente.

B.1.2 Sinais de entrada não-correlacionados

Assumindo que o sinal de entrada seja branco, de (B.9) e (B.10) tem-se

$$t_{i,i}^{\text{C}} = p_{i,j}^{\text{C}} = \frac{1}{M(M+1)}. \quad (\text{B.15})$$

B.2 Sinais de entrada gaussianos reais

Substituindo a função densidade de probabilidade gaussiana real, dada por (A.19), em (B.3) e levando em consideração que o sinal de entrada tem média nula, pode-se reescrever a expressão resultante como

$$\frac{\partial^2 \mathbf{F}_2^{\mathbb{R}}(\omega)}{\partial \omega^2} = \sqrt{\frac{\det[\mathbf{L}_{\mathbb{R}}(\omega)]}{\det(\mathbf{R})}} \mathbf{G}_2^{\mathbb{R}}(\omega) \quad (\text{B.16})$$

onde $\mathbf{L}_{\mathbb{R}}^{-1}(\omega)$ é dada por (A.21) e

$$\mathbf{G}_2^{\mathbb{R}}(\omega) = \frac{1}{(2\pi)^{M/2} \sqrt{\det[\mathbf{L}_{\mathbb{R}}(\omega)]}} \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{xx}^T \mathbf{vv}^T \mathbf{xx}^T e^{-\frac{1}{2}[\mathbf{x}^T \mathbf{L}_{\mathbb{R}}^{-1}(\omega) \mathbf{x}]} d\mathbf{x}. \quad (\text{B.17})$$

A expressão (B.17) pode ser vista como um momento de quarta ordem de variáveis aleatórias conjuntamente gaussianas, com $\mathbf{K}(n)$ [veja (2.24)] representando a matriz de ponderação e $\mathbf{L}_{\mathbb{R}}(\omega)$, a matriz de autocorrelação [2], [10]. Assim, fazendo $\mathbf{G}_2^{\mathbb{R}}(\omega) = \mathbf{L}_{\mathbb{R}}(\omega)$ em (B.17), considerando as propriedades do determinante, a autodecomposição de \mathbf{R} e integrando duas vezes com respeito a ω ambos os lados da expressão resultante, obtém-se

$$\mathbf{F}_2^{\mathbb{R}}(\omega) = \mathbf{Q} \left\{ 2 \iint \frac{[\Lambda(\mathbf{I} + 2\omega\Lambda)^{-1}] \mathbf{K}'(n) [\Lambda(\mathbf{I} + 2\omega\Lambda)^{-1}]}{\sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega + \iint \frac{[\Lambda(\mathbf{I} + 2\omega\Lambda)^{-1}] \text{tr}\{\mathbf{K}'(n) [\Lambda(\mathbf{I} + 2\omega\Lambda)^{-1}]\}}{\sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega \right\} \mathbf{Q}^T \quad (\text{B.18})$$

com $\mathbf{K}'(n)$ dado em (2.25) [por inspeção em (B.18), verifica-se agora que ζ é igual a 2].

Como apenas os elementos da diagonal de $\mathbf{K}'(n)$ afetam (2.32) e lembrando que o resultado aqui obtido é pré- e pós-multiplicado por \mathbf{Q}^H e \mathbf{Q} , respectivamente, (B.18) é reescrita como

$$\mathbf{f}_2^{\mathbb{R}}(\omega) = (2\mathbf{T} + \mathbf{P})\mathbf{k}'(n) \quad (\text{B.19})$$

onde $\mathbf{f}_2^{\mathbb{R}}(\omega)$ corresponde aos elementos da diagonal de $\mathbf{Q}^T \mathbf{F}_2^{\mathbb{R}}(\omega) \mathbf{Q}$ e $\mathbf{k}'(n)$, aos elementos da diagonal de $\mathbf{K}'(n)$. Dessa forma, a matriz diagonal \mathbf{T} tem seus elementos dados por

$$t_{i,i}^{\mathbb{R}} = \iint \frac{\lambda_i^2}{(1 + 2\omega\lambda_i)^2 \sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega + C_2^{\mathbb{R}} \quad (\text{B.20})$$

e a matriz simétrica \mathbf{P} por

$$p_{i,j}^{\mathbb{R}} = \iint \frac{\lambda_i \lambda_j}{(1 + 2\omega\lambda_i)(1 + 2\omega\lambda_j) \sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega + C_2^{\mathbb{R}} \quad (\text{B.21})$$

para $C_2^{\mathbb{R}} = \lim_{\omega \rightarrow \infty} \mathbf{F}_2^{\mathbb{R}}(\omega) = 0$. Portanto, a solução do momento de quarta ordem definido em (B.2) envolve o cálculo das integrais duplas presentes em (B.9) e (B.10).

B.2.1 Sinais de entrada correlacionados

A solução apresentada na sequência tem grande interesse prático, pois é comumente encontrada quando o sinal de entrada é correlacionado. Nessa situação, assume-se que todos os autovalores de \mathbf{R} sejam distintos [15]-[18], [21], [22]. Assim, similarmente ao desenvolvimento de (A.25)-(A.31), a partir de (B.9) e (B.10) obtém-se

$$t_{i,i}^{\mathbb{R}} \cong \frac{1}{4\sqrt{G^{\mathbb{R}}}} \iint \frac{1}{(\omega - \omega_i^{\mathbb{R}})^2 \prod_{q=1}^{M/2} (\omega - \tilde{\omega}_q^{\mathbb{R}})} d\omega \quad (\text{B.22})$$

e

$$p_{i,j}^{\mathbb{R}} \cong \frac{1}{4\sqrt{G^{\mathbb{R}}}} \iint \frac{1}{(\omega - \omega_i^{\mathbb{R}})(\omega - \omega_j^{\mathbb{R}}) \prod_{q=1}^{M/2} (\omega - \tilde{\omega}_q^{\mathbb{R}})} d\omega \quad (\text{B.23})$$

onde $\omega_k^{\mathbb{R}}$, $\tilde{\omega}_q^{\mathbb{R}}$, e $G^{\mathbb{R}}$ são, respectivamente, calculados através de (A.26), (A.29), e (A.28).

Logo, considerando apenas raízes distintas, realizando a expansão em frações parciais em (B.22) e (B.23), integrando as expressões resultantes e fazendo $\omega = 0$ [84], tem-se

$$t_{i,i}^{\mathbb{R}} \equiv \frac{1}{4\sqrt{G^{\mathbb{R}}}} \left(-A_{i,i}^{\mathbb{R}} \log(-\omega_i^{\mathbb{R}}) + \sum_{q=1}^{M/2} \frac{B_{i,i}^{\mathbb{R}} \{\tilde{\omega}_i^{\mathbb{R}} [1 - \log(\tilde{\omega}_i^{\mathbb{R}})] - \omega_i^{\mathbb{R}} [1 - \log(\omega_i^{\mathbb{R}})]\}}{(\tilde{\omega}_i^{\mathbb{R}} - \omega_i^{\mathbb{R}})} \right) \quad (\text{B.24})$$

e, para $i \neq j$,

$$p_{i,j}^{\mathbb{R}} \equiv \frac{1}{4\sqrt{G^{\mathbb{R}}}} \left\{ \frac{A_{i,i}^{\mathbb{R}} \omega_i^{\mathbb{R}} [1 - \log(-\omega_i^{\mathbb{R}})]}{(\omega_i^{\mathbb{R}} - \omega_j^{\mathbb{R}})} + \frac{A_{i,j}^{\mathbb{R}} \omega_j^{\mathbb{R}} [1 - \log(-\omega_j^{\mathbb{R}})]}{(\omega_j^{\mathbb{R}} - \omega_i^{\mathbb{R}})} + \sum_{l=1}^{M/2} \frac{B_{i,i}^{\mathbb{R}} \tilde{\omega}_l^{\mathbb{R}} [1 - \log(-\tilde{\omega}_l^{\mathbb{R}})]}{(\tilde{\omega}_l^{\mathbb{R}} - \omega_j^{\mathbb{R}})} \right\} \quad (\text{B.25})$$

com $A_{i,i}^{\mathbb{R}}$ e $B_{i,i}^{\mathbb{R}}$ dados, respectivamente, por (A.33) e (A.34). Dessa forma, a partir de (B.24) e (B.25), os elementos de **T** e **P** podem ser obtidos de forma analítica.

B.2.2 Sinais de entrada não-correlacionados

Assumindo que o sinal de entrada seja não-correlacionado, levando em conta (A.18), a partir de (B.20) e (B.21) obtém-se

$$t_{i,i}^{\mathbb{R}} = p_{i,j}^{\mathbb{R}} = \iint \frac{\sigma_x^4}{(1 + 2\omega\sigma_x^2)^2 \sqrt{(1 + 2\omega\sigma_x^2)^M}} d\omega. \quad (\text{B.26})$$

Agora, considerando M par e resolvendo a expressão resultante para $\omega = 0$ [84], tem-se

$$t_{i,i}^{\mathbb{R}} = p_{i,j}^{\mathbb{R}} = \frac{1}{M(M+2)}. \quad (\text{B.27})$$

CÁLCULO DE S

De forma similar ao Apêndice A, este apêndice apresenta um procedimento para o cálculo do momento normalizado dado pelo termo D de (2.31). Na literatura, esse tipo de valor esperado é somente calculado para o caso de sinais de entrada reais [17], [18], [21], [24]. Assim, a solução aqui proposta para sinais de entrada complexos pode ser vista como uma das contribuições deste trabalho. Para determinar o valor esperado em questão, uma função auxiliar $F_3(\omega)$ é definida como [14], [23], [63]

$$F_3(\omega) = \underbrace{\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \frac{\mathbf{x}\mathbf{x}^H}{(\mathbf{x}^H\mathbf{x})^2} e^{-\omega(\mathbf{x}^H\mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \quad (C.1)$$

tal que

$$E \left\{ \frac{\mathbf{x}(n)\mathbf{x}^H(n)}{[\mathbf{x}^H(n)\mathbf{x}(n)]^2} \right\} = F_3(\omega)|_{\omega=0} = F_3(0). \quad (C.2)$$

Assim, diferenciando duas vezes (C.1) em relação à ω , tem-se

$$\frac{\partial^2 F_3(\omega)}{\partial \omega^2} = \underbrace{\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty}}_{M \text{ integrais}} \mathbf{x}\mathbf{x}^H e^{-\omega(\mathbf{x}^H\mathbf{x})} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x}. \quad (C.3)$$

Portanto, conhecida $f_{\mathbf{x}}(\mathbf{x})$, soluções para o momento normalizado podem ser determinadas.

C.1 Sinais de entrada gaussianos complexos

Agora, considerando que o sinal de entrada tenha distribuição gaussiana, média nula e seja complexo, substituindo (A.5) em (C.3), obtém-se

$$\frac{\partial^2 F_3^C(\omega)}{\partial \omega^2} = \frac{\det[\mathbf{L}_C(\omega)]}{\det(\mathbf{R})} \mathbf{G}_1^C(\omega) \quad (C.4)$$

com $\mathbf{L}_C^{-1}(\omega)$ e $\mathbf{G}_1^C(\omega)$ dadas por (A.8) e (A.9), respectivamente. Então, fazendo $\mathbf{G}_1^C(\omega) = \mathbf{L}_C(\omega)$ em (C.4) (veja Apêndice A), utilizando as propriedades do determinante, a decomposição (2.14) e integrando duas vezes ambos os lados da expressão resultante em relação a ω , tem-se

$$\mathbf{F}_3^C(\omega) = \mathbf{Q}\mathbf{S}\mathbf{Q}^H \quad (\text{C.5})$$

onde a matriz diagonal \mathbf{S} tem seus elementos dados por

$$s_{i,i}^C = \iint \frac{\lambda_i}{(1 + \omega\lambda_i) \prod_{k=1}^M (1 + \omega\lambda_k)} d\omega + C_3^C \quad (\text{C.6})$$

com $C_3^C = \lim_{\omega \rightarrow \infty} \mathbf{F}_3^C(\omega) = 0$.

C.1.1 Sinais de entrada correlacionados

Admitindo que a matriz de autocorrelação do sinal de entrada não possua autovalores idênticos, a técnica de expansão em frações parciais nos permite obter uma solução fechada para (C.6). Assim, reescrevendo (C.6) como

$$s_{i,i}^C = \frac{1}{G^C} \iint \frac{1}{(\omega - \omega_i^C) \prod_{k=1}^M (\omega - \omega_k^C)} d\omega \quad (\text{C.7})$$

obtém-se

$$s_{i,i}^C = \frac{1}{G^C} \left\{ -A_{1,i}^C \log(-\omega_i^C) - \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^C \omega_l^C [1 - \log(-\omega_l^C)] + \sum_{\substack{l=1 \\ l \neq i}}^M B_{l,i}^C \omega_l^C [1 - \log(-\omega_l^C)] \right\} \quad (\text{C.8})$$

onde ω_k^C , G^C , $A_{1,i}^C$ e $B_{l,i}^C$ são, respectivamente, dados por (A.13), (A.14), (A.16) e (A.17). Finalmente, agrupando os termos comuns em (C.8), segue (2.38).

C.1.2 Sinais de entrada não-correlacionados

Considerando que o sinal de entrada seja não-correlacionado, a partir de (A.18) e (C.6) tem-se

$$s_{i,i}^C = \iint \frac{\sigma_x^2}{(1 + \omega\sigma_x^2)^{M+1}} d\omega \quad (\text{C.9})$$

Logo, resolvendo a integral em (C.9) para $\omega = 0$ [84], obtém-se (2.41).

C.2 Sinais de entrada gaussianos reais

Assumindo agora que o sinal de entrada seja real, tenha média zero e distribuição de probabilidade gaussiana, (C.3) pode ser reescrita como

$$\frac{\partial^2 \mathbf{F}_3^{\mathbb{R}}(\omega)}{\partial \omega^2} = \sqrt{\frac{\det[\mathbf{L}_{\mathbb{R}}(\omega)]}{\det(\mathbf{R})}} \mathbf{G}_1^{\mathbb{R}}(\omega) \quad (\text{C.10})$$

com $\mathbf{L}_{\mathbb{R}}^{-1}(\omega)$ dada por (A.21) e $\mathbf{G}_1^{\mathbb{R}}(\omega)$ por (A.22). Então, fazendo $\mathbf{G}_1^{\mathbb{R}}(\omega) = \mathbf{L}_{\mathbb{R}}(\omega)$, utilizando as propriedades do determinante, a decomposição (2.14) e integrando ambos os lados da expressão resultante duas vezes em relação à ω , obtém-se

$$\mathbf{F}_3^{\mathbb{R}}(\omega) = \mathbf{Q}\mathbf{S}\mathbf{Q}^T \quad (\text{C.11})$$

onde \mathbf{S} representa uma matriz diagonal, cujos elementos são dados por

$$s_{i,i}^{\mathbb{R}} = \iint \frac{\lambda_i}{(1 + 2\omega\lambda_i) \sqrt{\prod_{k=1}^M (1 + 2\omega\lambda_k)}} d\omega + C_3^{\mathbb{R}} \quad (\text{C.12})$$

com $C_3^{\mathbb{R}} = \lim_{\omega \rightarrow \infty} \mathbf{F}_3^{\mathbb{R}}(\omega) = 0$.

C.2.1 Sinais de entrada correlacionados

De maneira similar ao realizado em (A.25)-(A.31), a partir de (C.12) obtém-se

$$s_{i,i}^{\mathbb{R}} \cong \frac{1}{2\sqrt{G^{\mathbb{R}}}} \iint \frac{1}{(\omega - \omega_i^{\mathbb{R}}) \prod_{q=1}^{M/2} (\omega - \tilde{\omega}_q^{\mathbb{R}})} d\omega. \quad (\text{C.13})$$

Então, realizando expansão em frações parciais e resolvendo para $\omega = 0$, de (C.13) tem-se

$$s_{i,i}^{\mathbb{R}} \cong \frac{1}{2\sqrt{G^{\mathbb{R}}}} \left\{ A_{1,i}^{\mathbb{R}} \omega_i^{\mathbb{R}} [1 - \log(-\omega_i^{\mathbb{R}})] + \sum_{l=1}^{M/2} B_{l,i}^{\mathbb{R}} \tilde{\omega}_l^{\mathbb{R}} [1 - \log(-\tilde{\omega}_l^{\mathbb{R}})] \right\} \quad (\text{C.14})$$

sendo os parâmetros $\omega_k^{\mathbb{R}}$, $G^{\mathbb{R}}$, $\tilde{\omega}_q^{\mathbb{R}}$, $A_{1,i}^{\mathbb{R}}$ e $B_{l,i}^{\mathbb{R}}$ dados na Seção A.2.1.

C.2.2 Sinais de entrada não-correlacionados

Considerando que o sinal de entrada do sistema seja não-correlacionado e assumindo ordem do filtro adaptativo par, a partir de (C.12) obtém-se (2.41).

ANÁLISE DE [71]

Neste apêndice, uma análise da solução proposta em [71] para o cálculo da matriz de autocorrelação normalizada [dada por (2.1)] é apresentada. Tal análise tem como objetivo mostrar que a solução apresentada por Al-Naffouri *et al.* é instável para filtros com ordem maior do que dois (condição comumente encontrada em aplicações práticas). Cabe salientar que, apesar de a abordagem proposta em [71] utilizar a função distribuição de probabilidade cumulativa (CDF) no cálculo de \mathbf{R}^N ao invés da função densidade de probabilidade (PDF) como em [19], [20], [23], sabe-se que ambas as soluções devem obter resultados equivalentes. Particularmente em [71], visando diagonalizar \mathbf{R} , é utilizado um procedimento de branqueamento do sinal de entrada através da transformada Karhunen-Loève (*Karhunen-Loève Transform – KLT*) [25]. Tal processo equivale a decompor a matriz de autocorrelação normalizada como

$$\mathbf{R}^N = \mathbf{Q}\tilde{\mathbf{H}}\mathbf{Q}^H \quad (\text{D.1})$$

sendo $\tilde{\mathbf{H}}$ uma matriz diagonal cujos elementos são dados por

$$\tilde{h}_k = 1 - \sum_{m=1}^M \frac{\lambda_m^{M-1}}{\prod_{\substack{j=1 \\ j \neq m}}^M (\lambda_m - \lambda_j)} \left[1 - \frac{\sigma_{m,k}}{\lambda_m} + \sum_{\substack{j=1 \\ j \neq m}}^M c_j \log \left(\frac{1 + \bar{\sigma}_{j,m}}{\sigma_{m,k} \lambda_m^{-1} + \bar{\sigma}_{j,m}} \right) \right] \quad (\text{D.2})$$

com

$$\bar{\sigma}_{j,m} = \frac{\sigma_{j,k} - \sigma_{m,k}}{\lambda_m - \lambda_j}, \quad (\text{D.3})$$

$$\bar{\sigma}_{l,m} = \frac{\sigma_{l,k} - \sigma_{m,k}}{\lambda_m - \lambda_l}, \quad (\text{D.4})$$

$$\sigma_{m,k} = \begin{cases} \lambda_k, & m = k \\ 0, & m \neq k, \end{cases} \quad (\text{D.5})$$

e

$$c_j = \frac{(-\bar{\sigma}_{j,m} - \sigma_{m,k} \lambda_m^{-1})^{M-1}}{\prod_{\substack{l=1 \\ l \neq j,m}}^M (\bar{\sigma}_{l,m} - \bar{\sigma}_{j,m})}. \quad (D.6)$$

Para prosseguir com a análise, substituindo (D.3) e (D.4) em (D.6), c_j pode ser reescrita como

$$c_j = \frac{1}{(\lambda_m - \lambda_j)} \frac{\left(\frac{\lambda_j}{\lambda_m} \sigma_{m,k} - \sigma_{j,k} \right)^{M-1}}{\prod_{\substack{l=1 \\ l \neq j,m}}^M \left[\frac{\sigma_{l,k} (\lambda_m - \lambda_j) + \sigma_{m,k} (\lambda_m - \lambda_j) + \sigma_{j,k} (\lambda_l - \lambda_m)}{(\lambda_m - \lambda_l)} \right]}. \quad (D.7)$$

Agora, visando mostrar a instabilidade da solução, considere a condição

$$l \neq j \neq m \neq k \quad (D.8)$$

a qual ocorre naturalmente quando $M > 2$. Assim, a partir de (D.8) e (D.5), é possível mostrar que (D.7) resulta em uma condição de indeterminação. Além disso, utilizando a mesma abordagem no logaritmo de (D.2), levando-se em conta (D.3) e (D.4), tem-se

$$\log \left(\frac{1 + \bar{\sigma}_{j,m}}{\sigma_{m,k} \lambda_m^{-1} + \bar{\sigma}_{j,m}} \right) = \log \left[\frac{\lambda_m (\lambda_m - \lambda_j + \sigma_{j,k} - \sigma_{m,k})}{\lambda_m \sigma_{j,k} - \lambda_j \sigma_{m,k}} \right]. \quad (D.9)$$

Novamente, a partir de (D.8), constata-se que (D.9) tende para um valor não limitado. Portanto, para $l \neq j \neq m \neq k$, verifica-se que (D.7) e (D.9) contribuem para a instabilidade da solução apresentada em [71].

No intuito de identificar a origem do problema de instabilidade mencionado, substitui-se (D.7) e (D.9) em (D.2), decompondo o somatório em duas parcelas, a primeira válida para $m = k$ e segunda para $m \neq k$, o que resulta em

$$\tilde{h}_k = 1 - \underbrace{\frac{\lambda_k^{M-1}}{\prod_{\substack{j=1 \\ j \neq k}}^M (\lambda_k - \lambda_j)}}_{\text{para } m=k} \underbrace{(1 - \sigma_{k,k} \lambda_k^{-1} + \tilde{H}_1)}_{=0} - \sum_{\substack{m=1 \\ m \neq k}}^M \underbrace{\frac{\lambda_m^{M-1}}{\prod_{\substack{j=1 \\ j \neq m}}^M (\lambda_m - \lambda_j)}}_{=1} \underbrace{(1 - \sigma_{m,k} \lambda_m^{-1} + \tilde{H}_2)}_{\text{para } m \neq k} \quad (D.10)$$

com

$$\tilde{H}_1 = \sum_{\substack{j=1 \\ j \neq k}}^M \frac{1}{(\lambda_k - \lambda_j)} \frac{(\lambda_j - \sigma_{j,k})^{M-1} \log \left[\frac{\lambda_k(\lambda_k - \lambda_j + \sigma_{j,k} - \lambda_k)}{\lambda_k \sigma_{j,k} - \lambda_j \lambda_k} \right]}{\prod_{\substack{l=1 \\ l \neq j,k}}^M \left[\frac{\sigma_{l,k}(\lambda_k - \lambda_j) + \lambda_k(\lambda_j - \lambda_l) + \sigma_{j,k}(\lambda_l - \lambda_k)}{(\lambda_k - \lambda_l)} \right]} \quad (D.11)$$

e

$$\tilde{H}_2 = \sum_{\substack{j=1 \\ j \neq m}}^M \frac{1}{(\lambda_m - \lambda_j)} \frac{\left(\frac{\lambda_j}{\lambda_m} \sigma_{m,k} - \sigma_{j,k} \right)^{M-1} \log \left[\frac{\lambda_m(\lambda_m - \lambda_j + \sigma_{j,k} - \sigma_{m,k})}{\lambda_m \sigma_{j,k} - \lambda_j \sigma_{m,k}} \right]}{\prod_{\substack{l=1 \\ l \neq j,m}}^M \left[\frac{\sigma_{l,k}(\lambda_m - \lambda_j) + \sigma_{m,k}(\lambda_j - \lambda_l) + \sigma_{j,k}(\lambda_l - \lambda_m)}{(\lambda_m - \lambda_l)} \right]} \quad (D.12)$$

Então, a partir de (D.5) e (D.11) é possível mostrar que a primeira parcela do lado direito de (D.10) é nula, podendo ser desconsiderada. Nesse contexto, o problema de instabilidade da solução apresentada em [71] incide sobre o termo \tilde{H}_2 . Dessa forma, com base em (D.5) e (D.12), notando que $m \neq k$ implica $j \neq k$, obtém-se

$$\tilde{H}_2 = \sum_{\substack{j=1 \\ j \neq m}}^M \frac{1}{(\lambda_m - \lambda_j)} \frac{\overbrace{(-\sigma_{j,k})^{M-1}}^{=0}}{\prod_{\substack{l=1 \\ l \neq j,m}}^M \underbrace{\left[\frac{\sigma_{l,k}(\lambda_m - \lambda_j) + \sigma_{j,k}(\lambda_l - \lambda_m)}{(\lambda_m - \lambda_l)} \right]}_{=0}} \times \log \left[\frac{\lambda_m(\lambda_m - \lambda_j + \sigma_{j,k})}{\underbrace{\lambda_m \sigma_{j,k}}_{=0}} \right] \quad (D.13)$$

evidenciando que o mal-condicionamento de (D.2) se deve ao termo \tilde{H}_2 .

Para tornar (D.2) estável, após analisar (D.13), verificou-se que o somatório presente em (D.12) produz um resultado finito unicamente quando $j = k$. Dessa forma, (D.12) é aqui modificada para

$$\tilde{H}_2 = \frac{-\lambda_k}{(\lambda_m - \lambda_k)} \log\left(\frac{\lambda_m}{\lambda_k}\right). \quad (\text{D.14})$$

Finalmente, substituindo (D.14) em (D.10) e lembrando que $\tilde{H}_1 = 0$, obtém-se

$$\tilde{h}_k = 1 - \sum_{\substack{m=1 \\ m \neq k}}^M \frac{\lambda_m^{M-1}}{\prod_{\substack{j=1 \\ j \neq m}}^M (\lambda_m - \lambda_j)} \left[1 - \frac{\lambda_k}{(\lambda_m - \lambda_k)} \log\left(\frac{\lambda_m}{\lambda_k}\right) \right] \quad (\text{D.15})$$

cuja forma é muito similar à apresentada em [19], [20], [23]. Para ilustrar, considere a relação

$$1 - \sum_{\substack{m=1 \\ m \neq k}}^M \frac{\lambda_m^{M-1}}{\prod_{\substack{j=1 \\ j \neq m}}^M (\lambda_m - \lambda_j)} = \frac{\lambda_k^{M-1}}{\prod_{\substack{j=1 \\ j \neq k}}^M (\lambda_k - \lambda_j)} \quad (\text{D.16})$$

a qual possibilita reescrever (D.15) como em [23]. Portanto, a solução apresentada por Al-Naffouri *et al.* em 2011 para o cálculo de \mathbf{R}^N , quando modificada visando a sua estabilidade, leva à solução proposta por Kolodziej *et al.* em 2009 [23]; o que nos permite questionar os resultados de [71], já que não podem ser obtidos através da expressão fornecida por seus autores.

SOLUÇÃO APROXIMADA PARA (2.32)

Neste apêndice, uma solução aproximada para a recursão dos elementos da diagonal da matriz de autocovariância do vetor de erro nos coeficientes é desenvolvida. Tal recursão representa uma solução alternativa à (2.32), podendo ser usada na expressão que descreve a evolução do EQM do algoritmo NLMS [veja (2.27)-(2.29)]. Aqui, utiliza-se a aproximação apresentada em [90], [91], a qual vem sendo considerada na modelagem de diversos algoritmos adaptativos da literatura [19], [20], [23], [79]. Tal aproximação estabelece que, durante os estágios iniciais da aprendizagem, a norma do comportamento médio do vetor de erro nos coeficientes domina as flutuações em $\mathbf{K}(n)$, permitindo considerar que

$$\begin{aligned}\mathbf{K}(n) &= E[\mathbf{v}(n)\mathbf{v}^H(n)] \\ &\cong E[\mathbf{v}(n)]E[\mathbf{v}^H(n)].\end{aligned}\tag{E.1}$$

Para avaliar o efeito desta aproximação no modelo do algoritmo NLMS, uma solução alternativa para a recursão de $\mathbf{k}'(n)$ é determinada a partir de (2.74) e (E.1), isto é,

$$\begin{aligned}E[\mathbf{v}(n+1)]E[\mathbf{v}^H(n+1)] &= E[\mathbf{v}(n)]E[\mathbf{v}^H(n)] - \mu E[\mathbf{v}(n)]E[\mathbf{v}^H(n)]\mathbf{R}^N \\ &\quad - \mu\mathbf{R}^N E[\mathbf{v}(n)]E[\mathbf{v}^H(n)] \\ &\quad + \mu^2\mathbf{R}^N E[\mathbf{v}(n)]E[\mathbf{v}^H(n)]\mathbf{R}^N.\end{aligned}\tag{E.2}$$

Agora, utilizando os resultados obtidos no Apêndice A em (E.2), pré- e pós-multiplicando a expressão resultante por \mathbf{Q}^H e \mathbf{Q} , respectivamente, e tomando os elementos da diagonal da expressão resultante, sem perda de generalidade, a recursão pode ser reescrita como

$$\tilde{\mathbf{k}}(n) = \tilde{\mathbf{B}}^n \tilde{\mathbf{k}}(0)\tag{E.3}$$

em que $\tilde{\mathbf{k}}(n) = \text{diag}\{\mathbf{Q}^H E[\mathbf{v}(n)]E[\mathbf{v}^H(n)]\mathbf{Q}\}$ e

$$\tilde{\mathbf{B}} = \mathbf{I} - 2\mu\mathbf{H} + \mu^2\mathbf{H}^2.\tag{E.4}$$

Logo, fazendo $n \rightarrow \infty$ em (E.3) e assumindo convergência, percebe-se que os elementos de $\tilde{\mathbf{k}}(n)$ tendem para zero, contrariando o resultado

obtido através de (2.45). Adicionalmente, observa-se que (E.3) consegue capturar apenas de forma aproximada o comportamento do algoritmo na região transitória, mas não o erro em regime permanente. Especificamente, em regime transitório, analisando (2.33) e (E.4), constata-se que, para $\mu \rightarrow 0$, tem-se $\tilde{\mathbf{B}} \rightarrow \mathbf{B}$.

Para ilustrar o efeito de (E.3) na curva de aprendizagem do algoritmo NLMS, o Exemplo 7 (veja Seção 3.4.3) é aqui reconsiderado. Assumindo que o parâmetro de regularização é igual a zero, o passo de adaptação, $\mu = 0,1$, e a variância do ruído de medição, $\sigma_z^2 = 10^{-2}$, obtém-se o resultado ilustrado na Figura E.1. Verifica-se que tanto o modelo proposto quanto sua versão simplificada exibem resultados semelhantes no estágio inicial de adaptação, conforme indicado em [90] e [91], se distanciando à medida que o algoritmo se aproxima da solução em regime permanente. Além disso, constata-se que a convergência do modelo simplificado é mais rápida do que a prevista pelo modelo proposto, apresentando também um erro em regime permanente menor, o que contradiz o resultado obtido na simulação MC. Portanto, conclui-se que, salvo condições muito particulares, a abordagem simplificada deve ser evitada, uma vez que ela produz uma solução pouco precisa e que, em geral, não condiz com o comportamento efetivo do algoritmo (especialmente, em regime permanente).

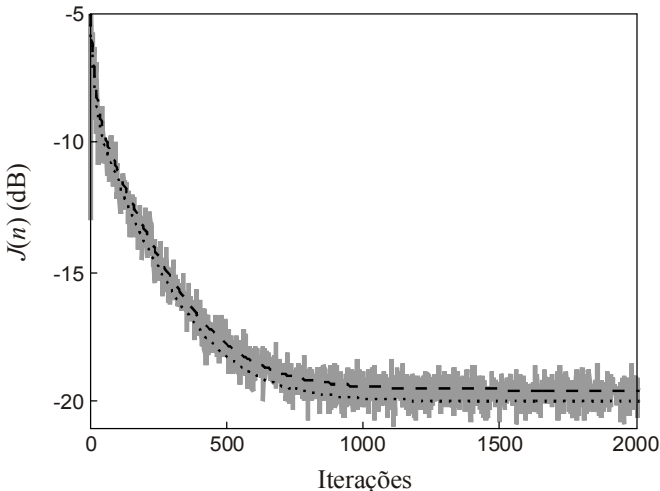


Figura E.1: Curva de aprendizagem do algoritmo NLMS. (Linha irregular cinza) simulação MC. (Linha escura pontilhada) modelo simplificado. (Linha escura tracejada) modelo proposto.

REFERÊNCIAS

- [1] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications*, 1st ed. Baffins Lane Chichester, U.K.: John Wiley & Sons, 1999.
 - [2] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing: Spectral Estimation, Signal Modeling, Adaptive Filtering and Array Processing*, 1st ed. Norwood, MA: McGraw-Hill, 2000.
 - [3] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, 3rd ed. Boston, MA: Springer, 2008.
 - [4] S. L. Gay and J. Benesty, *Acoustic Signal Processing for Telecommunication*, 1st ed. Norwell, MA: Kluwer Academic Publishers, 2000.
 - [5] M. M. Sondhi, “The history of echo cancellation,” *IEEE Signal Process. Mag.*, vol. 23, no. 5, pp. 95–102, Sept. 2006.
 - [6] M. M. Sondhi and D. A. Berkley, “Silencing echoes on the telephone network,” *Proc. IEEE*, vol. 68, no. 8, pp. 948–963, Aug. 1980.
 - [7] D. Messerschmitt, “Echo cancellation in speech and data transmission,” *IEEE J. Sel. Areas Commun.*, vol. 2, no. 2, pp. 283–297, Mar. 1984.
 - [8] M. L. Honig and D. G. Messerschmitt, *Adaptive Filters: Structure, Algorithms, and Applications*, 1st ed. Norwell, MA: Kluwer Academic Publishers, 1984.
 - [9] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, 1st ed. Englewood Cliffs, NJ: Prentice-Hall, 1985.
 - [10] A. H. Sayed, *Adaptive Filters*, 1st ed. Hoboken, NJ: John Wiley & Sons, 2008.
 - [11] S. Haykin, *Adaptive Filter Theory*, 4th ed. Upper Saddle River, NJ: Prentice-Hall, 2002.
-

-
- [12] S. M. Kuo and D. R. Morgan, “Active noise control: a tutorial review,” *Proc. IEEE*, vol. 87, no. 6, pp. 943–975, Jun. 1999.
- [13] Y. Huang and J. Benesty, *Audio Signal Processing: For Next-Generation Multimedia Communication Systems*, 1st ed. Norwell, MA: Kluwer Academic Publishers, 2004.
- [14] N. J. Bershad, “Analysis of the normalized LMS algorithm with Gaussian inputs,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 4, pp. 793–806, Aug. 1986.
- [15] J. E. Kolodziej, O. J. Tobias, and R. Seara, “Cálculo aproximado de integrais hiperelípticas aplicado à modelagem estatística do algoritmo NLMS,” in *Encuentro de Educación Matemática en Carreras de Ingeniería* (EMCI), Oberá, Misiones, Argentina, Oct. 2006, pp. 1-11.
- [16] E. M. Lobato, O. J. Tobias, and R. Seara, “Stochastic model for the NLMS algorithm with correlated Gaussian data,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* (ICASSP), Toulouse, France, May 2006, vol. 3, pp. 760–763.
- [17] J. E. Kolodziej, O. J. Tobias, and R. Seara, “Modelagem de algoritmos adaptativos normalizados para sinais gaussianos e não-gaussianos,” in *Encuentro del Grupo Latinoamericano de Emisión Acústica* (E-GLEA5), Oberá, Misiones, Argentina, Aug. 2007, pp. 1–6.
- [18] J. E. Kolodziej, O. J. Tobias, and R. Seara, “An improved stochastic model of the NLMS algorithm for correlated input data,” in *Proc. Eur. Signal Process. Conf.* (EUSIPCO), Poznan, Poland, Sept. 2007, pp. 360–364.
- [19] J. E. Kolodziej, O. J. Tobias, and R. Seara, “Modelagem estatística do algoritmo CSG aplicada para um arranjo adaptativo de antenas,” in *Simpósio Brasileiro de Telecomunicações* (SBrT), Recife, PE, Brazil, Sept. 2007, pp. 1–6.
- [20] J. E. Kolodziej, O. J. Tobias, R. Seara, and D. R. Morgan, “An analytical model for the constrained stochastic gradient algorithm,” in *Proc. Eur. Signal Process. Conf.* (EUSIPCO), Lausanne, Switzerland, Aug. 2008, pp. 1–5.
- [21] J. E. Kolodziej, O. J. Tobias, and R. Seara, “Um modelo preciso para o algoritmo LMS normalizado,” in *Congresso Brasileiro de Automática* (CBA), Juiz de Fora, MG, Brazil, Sept. 2008, pp. 1-6.
-

-
- [22] 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.
- [23] J. E. Kolodziej, O. J. Tobias, R. Seara, and D. R. Morgan, “On the constrained stochastic gradient algorithm: Model, performance, and improved version,” *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1304–1315, Apr. 2009.
- [24] J. E. Kolodziej, “Algoritmos adaptativos com sinal de entrada normalizado: Modelagem estatística e aprimoramentos,” Tese, Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brazil, 2010.
- [25] C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*, 1st ed. Englewood Cliffs, NJ: Prentice-Hall, 1992.
- [26] S. M. Kay, *Fundamentals of statistical signal processing: Estimation theory*, 1st ed. Upper Saddle River, NJ: Prentice-Hall, 1993.
- [27] A. N. Kolmogorov, “Sur l’interpolation et extrapolation des suites stationnaires,” *C. R. Acad. Sci. Paris*, vol. 208, pp. 2043–2045, 1939.
- [28] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series* (Originally issued as a classified National Defense Research Report in February 1942). Cambridge, MA: M.I.T Press, 1949.
- [29] H. O. A. Wold, *A study in the analysis of stationary time series*, Uppsala, Sweden: Almqvist & Wiksell, 1938.
- [30] A. D. Poularikas and Z. M. Ramadan, *Adaptive Filtering Primer with MATLAB*, 1st ed. Boca Raton, FL: Taylor & Francis Group, 2006.
- [31] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. London, U.K.: The Johns Hopkins University Press, 1996.
- [32] A. J. Laub, *Matrix Analysis for Scientists and Engineers*, 1st ed. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 2005.
- [33] D. S. Bernstein, *Matrix Mathematics: Theory, Facts, and Formulas*, 2nd ed. Woodstock, OX: Princeton University Press, 2009.
-

-
- [34] 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.
- [35] E. L. O. Batista, “Filtros Volterra adaptativos: Estruturas interpoladas e modelos estocásticos,” Tese, Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brazil, 2009.
- [36] H. E. Kallmann, “Transversal filters,” *Proc. IRE*, vol. 28, no. 7, pp. 302–310, Jul. 1940.
- [37] B. Widrow and M. E. Hoff Jr., “Adaptive switching circuits,” *IRE WESCON Conv. Rec.*, vol. 4, pp. 96–104, Aug. 1960.
- [38] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson Jr., “Stationary and nonstationary learning characteristics of the LMS adaptive filter,” *Proc. IEEE*, vol. 64, no. 8, pp. 1151–1162, Aug. 1976.
- [39] S. Kaczmarz, “Angenäherte Auflösung von Systemen linearer Gleichungen,” *Bull. Int. Acad. Polon. Sci. Lettres A*, vol. 35, no. III, pp. 355–357, 1937.
- [40] 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.
- [41] A. E. Albert and L. A. Gardner Jr., *Stochastic Approximation and Nonlinear Regression*, 1st ed. Cambridge, MA: MIT Press, 1967.
- [42] T. Aboulnasr and K. Mayyas, “A robust variable step-size LMS-type algorithm: analysis and simulations,” *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 631–639, Mar. 1997.
- [43] R. H. Kwong and E. W. Johnston, “A variable step size LMS algorithm,” *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1633–1642, Jul. 1992.
- [44] B. Farhang-Boroujeny, “Variable-step-size LMS algorithm: new developments and experiments,” *Proc. IEE Vis. Image Signal Process.*, vol. 141, no. 5, p. 311–317, Oct. 1994.
- [45] R. W. Harris, D. M. Chabries, and F. A. Bishop, “A variable step (VS) adaptive filter algorithm,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 2, pp. 309–316, Apr. 1986.
- [46] D. L. Duttweiler, “Proportionate normalized least-mean-squares adaptation in echo cancelers,” *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508–518, Sept. 2000.
-

-
- [47] H. Deng and M. Doroslovacki, "Proportionate adaptive algorithms for network echo cancellation," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1794–1803, May 2006.
- [48] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* (ICASSP), Orlando, FL, May 2002, vol. 2, no. II, pp. 1881–1884.
- [49] P. A. Naylor, J. Cui, and M. Brookes, "Adaptive algorithms for sparse echo cancellation," *Signal Process.*, vol. 86, no. 6, pp. 1182–1192, Jun. 2006.
- [50] F. C. de Souza, O. J. Tobias, R. Seara, and D. R. Morgan, "A PNLMS algorithm with individual activation factors," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2036–2047, Apr. 2010.
- [51] B. Widrow, J. McCool, and M. Ball, "The complex LMS algorithm," *Proc. IEEE*, vol. 63, no. 4, pp. 719–720, Apr. 1975.
- [52] D. T. M. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms," *IEEE Trans. Signal Process.*, vol. 41, no. 9, pp. 2811–2825, Sept. 1993.
- [53] P. E. An, M. Brown, and C. J. Harris, "On the convergence rate performance of the normalized least-mean-square adaptation," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 1211–1214, Sept. 1997.
- [54] D. R. Morgan and S. G. Kratzer, "On a class of computationally efficient, rapidly converging, generalized NLMS algorithms," *IEEE Signal Process. Lett.*, vol. 3, no. 8, pp. 245–247, Aug. 1996.
- [55] R. R. Bitmead and B. D. O. Anderson, "Performance of adaptive estimation algorithms in dependent random environments," *IEEE Trans. Autom. Contr.*, vol. 25, no. 4, pp. 788–794, Aug. 1980.
- [56] 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.
- [57] J. C. M. Bermudez and M. H. Costa, "A statistical analysis of the ϵ -NLMS and NLMS algorithms for correlated Gaussian signals," *J. Commun. Inform. Syst.*, vol. 20, no. 2, pp. 7–13, Aug. 2005.
- [58] S. C. Douglas and T. H.-Y. Meng, "Normalized data nonlinearities for LMS adaptation," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1352–1365, Jun. 1994.
-

-
- [59] J. Benesty, C. Paleologu, and S. Ciochina, “On regularization in adaptive filtering,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 6, pp. 1734–1742, Aug. 2011.
- [60] N. J. Bershad, “Behavior of the ε -normalized LMS algorithm with Gaussian inputs,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 5, pp. 636–644, May 1987.
- [61] G. Barrault, M. H. Costa, J. C. M. Bermudez, and A. Lenzi, “A new analytical model for the NLMS algorithm,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Philadelphia, PA, Mar. 2005, vol. 4, no. IV, pp. 41–44.
- [62] 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.
- [63] J. Kiusalaas, *Numerical Methods in Engineering with MATLAB®*, 2nd ed. New York, NY: Cambridge University Press, 2010.
- [64] D. R. Morgan, “Comments on ‘Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data’,” *IEEE Trans. Inf. Theory*, vol. 35, no. 6, pp. 1299, Nov. 1989.
- [65] M. Tarrab and A. Feuer, “Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data,” *IEEE Trans. Inf. Theory*, vol. 34, no. 4, pp. 680–691, Jul. 1988.
- [66] M. H. Costa and J. C. M. Bermudez, “An improved model for the normalized LMS algorithm with Gaussian inputs and large number of coefficients,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Orlando, FL, May 2002, vol. 2, no. II, pp. 1385–1388.
- [67] C. G. Samson and V. 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.
- [68] A. Benveniste and G. Ruget, “A measure of the tracking capability of recursive stochastic algorithms with constant gains,” *IEEE Trans. Autom. Control*, vol. 27, no. 3, pp. 639–649, Jun. 1982.
-

-
- [69] S. C. Chan and Y. Zhou, “Convergence behavior of NLMS algorithm for Gaussian inputs: Solutions using generalized Abelian integral functions and step size selection,” *J. Signal Process. Syst.*, vol. 59, no. 3, pp. 255–265, Jul. 2009.
- [70] T. Y. Al-Naffouri and M. Moinuddin, “Exact performance analysis of the ε -NLMS algorithm for colored circular Gaussian inputs,” *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5080–5090, Oct. 2010.
- [71] T. Y. Al-Naffouri, M. Moinuddin, and M. S. Sohail, “Mean weight behavior of the NLMS algorithm for correlated Gaussian inputs,” *IEEE Signal Process. Lett.*, vol. 18, no. 1, pp. 7–10, Jan. 2011.
- [72] S. C. Douglas and W. Pan, “Exact expectation analysis of the LMS adaptive filter,” *IEEE Trans. Signal Process.*, vol. 43, no. 12, pp. 2863–2871, Dec. 1995.
- [73] S. C. Douglas and T. H. Y. Meng, “Exact expectation analysis of the LMS adaptive filter without the independence assumption,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, San Francisco, CA, Mar. 1992, vol. 4, no. IV, pp. 61–64.
- [74] S. C. Douglas, “Exact expectation analysis of the LMS adaptive filter for correlated Gaussian input data,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Minneapolis, MN, Apr. 1993, vol. 3, no. III, pp. 519–522.
- [75] B. Fisher and N. J. Bershad, “The complex LMS adaptive algorithm: Transient weight mean and covariance with applications to the ALE,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 31, no. 1, pp. 34–44, Feb. 1983.
- [76] L. L. Horowitz and K. D. Senne, “Performance advantage of complex LMS for controlling narrow-band adaptive arrays,” *IEEE Trans. Circuits Syst.*, vol. 28, no. 6, pp. 562–576, Jun. 1981.
- [77] L. L. Horowitz and K. D. Senne, “Performance advantage of complex LMS for controlling narrow-band adaptive arrays,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 3, pp. 722–736, Jun. 1981.
-

- [78] T. A. C. M. Claasen and W. F. G. Mecklenbrauker, "Comparison of the convergence of two algorithms for adaptive FIR digital filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 3, pp. 670–678, Jun. 1981.
- [79] J. R. V. López, O. J. Tobias, and R. Seara, "Algoritmo LMS com entrada filtrada/erro filtrado modificado: Algoritmo e modelo estocástico," in *Simpósio Brasileiro de Telecomunicações (SBrT)*, Blumenau, SC, Brazil, Sept. 2009, pp. 1–6.
- [80] J. R. V. López, O. J. Tobias, and R. Seara, "Modified filtered-reference/filtered-error LMS algorithm: Algorithm and stochastic model," in *Proc. European Signal Process. Conf. (EUSIPCO)*, Glasgow, Scotland, Aug. 2009, pp. 1740–1744.
- [81] A. Feuer and E. Weinstein, "Convergence analysis of LMS filters with uncorrelated Gaussian data," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 1, pp. 222–230, Feb. 1985.
- [82] J. E. Kolodziej, O. J. Tobias, and R. Seara, "Stochastic analysis of the transform domain LMS algorithm for a non-stationary environment," in *Proc. European Signal Process. Conf. (EUSIPCO)*, Glasgow, Scotland, Aug. 2009, pp. 1730–1734.
- [83] D. R. Morgan, "Adaptive signal processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 4, pp. 1017–1018, Aug. 1986.
- [84] I. S. Gradshteyn and I. M. Ryzhik (Editors A. Jeffrey and D. Zwillinger), *Table of Integrals, Series and Products*, 7th ed. San Diego, CA: Academic Press, 2007.
- [85] M. Moinuddin, T. Y. Al-Naffouri, and M. S. Sohail, "Exact tracking analysis of the ϵ -NLMS algorithm for circular complex correlated Gaussian input," in *Proc. IEEE Int. Symp. Signal Process. Inform. Technol. (ISSPIT)*, Luxor, Egypt, Dec. 2010, pp. 225–230.
- [86] C. T. Chen, *Linear System Theory and Design*, 3th ed. New York, NY: Oxford University Press, 2009.
- [87] C. L. Siegel, *Topics in Complex Function Theory: Automorphic Functions and Abelian Integrals*, 1st ed. New York, NY: John Wiley & Sons, 1988.
-

-
- [88] I. N. Bronshtein and K. A. Semendyayev, *Handbook of Mathematics* (English translation edited by K.A. Hirsch). 3rd ed. Berlin, Germany: Springer-Verlag, 1997.
- [89] F. C. de Souza, O. J. Tobias, R. Seara, and D. R. Morgan, “Stochastic model for the mean weight evolution of the IAF-PNLMS algorithm,” *IEEE Trans. Signal Process.*, vol. 58, no. 11, pp. 5895–5901, Nov. 2010.
- [90] N. J. Bershad, P. Celka, and J.-M. Vesin, “Stochastic analysis of gradient adaptive identification of nonlinear systems with memory,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* (ICASSP), Seattle, WA, May 1998, vol. 3, pp. 1421-1424.
- [91] 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.
-