



COPPE/UFRJ

ANÁLISE DO ALGORITMO SET-MEMBERSHIP AFFINE PROJECTION

Markus Vinícius Santos Lima

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Paulo Sergio Ramirez Diniz

Rio de Janeiro
Setembro de 2009

ANÁLISE DO ALGORITMO SET-MEMBERSHIP AFFINE PROJECTION

Markus Vinícius Santos Lima

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

Prof. Paulo Sergio Ramirez Diniz, Ph.D.

Prof. Marcello Luiz Rodrigues de Campos, Ph.D.

Prof. José Antonio Apolinário Junior, D.Sc.

Prof. Vitor Heloiz Nascimento, Ph.D.

RIO DE JANEIRO, RJ – BRASIL

SETEMBRO DE 2009

Lima, Markus Vinícius Santos

Análise do Algoritmo Set-Membership Affine Projection/Markus Vinícius Santos Lima. – Rio de Janeiro: UFRJ/COPPE, 2009.

XIX, 80 p.: il.; 29, 7cm.

Orientador: Paulo Sergio Ramirez Diniz

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2009.

Referências Bibliográficas: p. 73 – 78.

1. Filtragem Adaptativa. 2. Análise de Algoritmo.
3. Reuso de Dados. 4. Seleção de Dados. I. Diniz, Paulo Sergio Ramirez. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

*À minha família, à Bruna e aos
meus grandes amigos.*

Agradecimentos

Agradeço aos meus pais, Luiz Álvaro e Aracy, pelo amor, carinho, compreensão e apoio que me foram dados ao longo de todos esses anos. Além deles, agradeço ao meu irmão, Álvaro, pelos bons momentos que passamos juntos.

Agradeço à minha namorada, Bruna Cesario, por ser uma grande companheira e por me fazer rir, mesmo nos momentos mais difíceis, e por estar sempre disposta a ajudar.

Agradeço ao Paulo Diniz, meu orientador, professor e amigo, pelos seus conselhos, ensinamentos e também pelos momentos de descontração. Diniz, você é a pessoa em quem eu tento me espelhar.

Agradeço aos membros da banca, pois seus comentários foram de grande valor e muito me ajudaram a melhorar esta dissertação e/ou a expandir meus conhecimentos.

Agradeço aos professores do Programa de Engenharia Elétrica, em especial aos professores Amit Bhaya e Luiz Wagner, pelos excelentes cursos ministrados.

Agradeço aos companheiros de laboratório que fazem do LPS, uma grande família. Inúmeras vezes as pessoas param suas atividades para ajudar no problema de outras. Acredito que a cooperação é o fator chave para o sucesso.

Agradeço ao Projeto COPPETEX que, para facilitar a vida dos alunos, disponibiliza um modelo de documento que já respeita as especificações exigidas pela COPPE, fazendo com que o aluno não perca tempo com detalhes de formatação e possa se concentrar apenas na escrita do documento.

Agradeço também à CAPES pelos dois anos de bolsa de mestrado concedida a mim.

Por último, mas não menos importante, agradeço a Deus.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

ANÁLISE DO ALGORITMO SET-MEMBERSHIP AFFINE PROJECTION

Markus Vinícius Santos Lima

Setembro/2009

Orientador: Paulo Sergio Ramirez Diniz

Programa: Engenharia Elétrica

Filtros adaptativos estão presentes em nosso cotidiano em uma infinidade de aplicações, tais como equalização do sinal recebido em sistemas de comunicações, cancelamento de eco e estimação de parâmetros. Na prática, os algoritmos do tipo gradiente estocástico são os mais utilizados, embora já existam algoritmos mais eficientes, como por exemplo os da família *set-membership*, que conseguem obter um resultado superior em termos de erro quadrático médio (MSE), além de requerer um número reduzido de operações aritméticas. Como os algoritmos do tipo *set-membership* são relativamente recentes (surgimento a partir de 1998) e um pouco mais complexos, características importantes tais como desempenho de MSE em regime permanente, velocidade de convergência e estabilidade ainda não foram plenamente estudadas. O objetivo desta dissertação é estudar o comportamento de MSE em regime permanente do algoritmo *set-membership affine projection* (SM-AP). O SM-AP foi escolhido por conciliar duas características importantes, conhecidas como reuso e seleção de dados, além de generalizar vários outros algoritmos.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

ANALYSIS OF THE SET-MEMBERSHIP AFFINE PROJECTION
ALGORITHM

Markus Vinícius Santos Lima

September/2009

Advisor: Paulo Sergio Ramirez Diniz

Department: Electrical Engineering

Adaptive filters are present in our daily life in various applications, such as equalization of the received signal in communication systems, echo cancellation, and parameter estimation. In practice, stochastic gradient algorithms are the most widely used, even though there are more efficient algorithms, for example those belonging to the set-membership family, which obtain better result in terms of mean-square error (MSE), and also require a reduced number of computations. Since the set-membership algorithms are relatively recent (emerged in 1998) and somewhat more complex, important characteristics such as MSE performance in steady-state, convergence speed, and stability have not yet been studied. This dissertation aims to study the MSE behavior of the set-membership affine projection (SM-AP) algorithm during steady-state. The SM-AP was addressed since it combines two important features, viz., data reuse and data selection, in addition to generalizing many other algorithms.

Sumário

Lista de Figuras	xi
Lista de Tabelas	xiv
Lista de Símbolos	xv
Lista de Abreviaturas	xviii
1 Introdução	1
1.1 Objetivo	3
1.2 Motivação	4
1.3 Organização da Dissertação	4
1.4 Notação	5
2 Filtragem Adaptativa	7
2.1 Conceitos Básicos	8
2.2 Gradiente Estocástico	9
2.2.1 LMS	11
2.2.2 NLMS	12
2.2.3 AP: Reuso de Dados	14
2.2.4 Considerações Finais	16
2.3 <i>Set-Membership Filtering</i> : Seleção de Dados	17
2.3.1 SM-NLMS	19
2.3.2 SM-AP	20
2.3.3 Considerações Finais	22

3	Principais Técnicas de Análise de EMSE	24
3.1	Equação a Diferenças	24
3.2	Equação Diferencial Ordinária	27
3.3	Conservação de Energia	28
3.4	Escolha do Método	30
4	Análise de EMSE do SM-AP	32
4.1	Resumo das Principais Variáveis	32
4.2	Preliminares	34
4.2.1	O Modelo de Análise	36
4.3	Expressões de Conservação de Energia	37
4.4	Vetor de Restrição com Módulo Constante Baseado no Erro	41
4.4.1	Aproximação para $E[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k)]$	42
4.4.2	Aproximação para $E[\mathbf{n}(k)\mathbf{n}^T(k)]$	45
4.4.3	Aproximação para $E[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k)]$	45
4.4.4	Aproximação para $E[\boldsymbol{\gamma}(k)\mathbf{n}^T(k)]$	45
4.4.5	Aproximação para $E[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k)]$	46
4.4.6	Aproximação para $E[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k)]$	46
4.4.7	EMSE para SM-AP	46
4.4.8	Eurísticas sobre $\rho_0(k)$ e P_{up}	48
4.5	Vetor Restrição Trivial	51
5	Resultados	53
5.1	Cenário 1	53
5.1.1	SM-NLMS ($L = 0$)	54
5.1.2	SM-BNLMS ($L = 1$)	59
5.1.3	$L = 2$	63
5.1.4	$L = 4$	67
5.2	Cenário 2	68
5.3	Resumo dos resultados	69
6	Conclusões	70
7	Trabalhos Futuros	72

Referências Bibliográficas	73
A Prova do Teorema 1	79

Lista de Figuras

2.1	Modelo básico de operação do filtro adaptativo.	9
2.2	Interpretação geométrica da atualização do NLMS no espaço dos coeficientes.	13
2.3	Interpretação geométrica da atualização do AP no espaço dos coeficientes, considerando $N = 2$ e $L = 1$	15
2.4	Interpretação geométrica do SMF no espaço dos coeficientes, considerando $k = 1$	18
2.5	Interpretação geométrica da atualização do SM-NLMS no espaço dos coeficientes.	19
2.6	Interpretação geométrica da atualização do SM-AP no espaço dos coeficientes.	23
5.1	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada branca ($L = 0$). . .	54
5.2	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada branca ($L = 0$).	55
5.3	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada branca ($L = 0$).	55
5.4	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada com razão de autovalores 20 ($L = 0$).	55
5.5	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada com razão de autovalores 20 ($L = 0$). . .	56
5.6	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ - entrada com razão de autovalores 20 ($L = 0$).	56

5.7	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 0$).	56
5.8	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 0$).	57
5.9	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 0$).	57
5.10	Desajuste do SM-NLMS em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – (a) entrada branca (b) entrada com razão de autovalores 20 (c) entrada com razão de autovalores 80	58
5.11	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).	60
5.12	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).	60
5.13	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).	60
5.14	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).	61
5.15	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).	61
5.16	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).	61
5.17	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).	62
5.18	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).	62
5.19	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).	62
5.20	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).	64

5.21	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).	64
5.22	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).	64
5.23	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).	65
5.24	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).	65
5.25	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).	65
5.26	Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 2$).	66
5.27	MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 2$).	66
5.28	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 2$).	67
5.29	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 4$).	68
5.30	Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores igual a 80 ($L = 1$) e $\sigma_n^2 = 10^{-2}$	68

Lista de Tabelas

4.1	Escolha de α em função da L e da razão de autovalores.	50
4.2	Escolha de P_{\min} em função de L e da razão de autovalores.	51
5.1	Valores de τ_o considerando $L = 0$	59
5.2	Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 0$).	59
5.3	Valores de τ_o considerando $L = 1$	63
5.4	Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 1$).	63
5.5	Valores de τ_o considerando $L = 2$	67
5.6	Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 2$).	67

Lista de Símbolos

$\tilde{e}(k)$	erro-sem-ruído <i>a priori</i> , p. 29
$(\cdot)^T$	operação de transposição aplicada a (\cdot) , p. 5
L	fator de reuso de dados, p. 15
N	ordem do filtro, p. 8
$\Delta\mathbf{w}(k)$	vetor desvio (ou vetor de erro dos coeficientes), p. 25
\mathbf{I}	matrix identidade, p. 6
$\mathbf{\Lambda}$	matriz diagonal formada pelos autovalores de \mathbf{R} , p. 26
\mathbf{Q}	matriz ortogonal cujas colunas são os autovetores de \mathbf{R} , p. 26
\mathbf{R}	matriz de auto-correlação do vetor de entrada, p. 10
$\text{tr}\{\cdot\}$	operador traço, p. 5
Θ	conjunto viável, p. 18
$\mathbf{X}(k)$	matriz de entrada, p. 15
$\mathbf{a} \perp \mathbf{b}$	\mathbf{a} é ortogonal a \mathbf{b} , p. 13
\mathcal{S}	conjunto formado por todos os possíveis pares de entrada (\mathbf{x}, d) , p. 18
$\mathbf{d}(k)$	vetor desejado, p. 15
δ	fator de regularização, p. 14
$\mathbf{e}(k)$	vetor erro <i>a priori</i> (ou simplesmente, vetor erro), p. 15

$\bar{\gamma}$	limite superior do erro, p. 18
$\gamma(k)$	vetor de restrição, p. 21
$\mathbf{g}(k)$	gradiente (ou vetor gradiente), p. 10
$\hat{\mathbf{g}}(k)$	aproximação instantânea do gradiente, p. 10
\mathbb{R}	conjunto dos números reais, p. 5
\mathbb{R}_+	conjunto dos números reais não-negativos, p. 5
$\mathbf{0}$	vetor ou matriz com todas as entradas zero, p. 6
μ	fator de convergência (ou tamanho do passo), p. 10
\mathbf{p}	vetor de correlação-cruzada entre o sinal desejado e o vetor de entrada, p. 10
$\psi(k)$	<i>exact membership set</i> , p. 18
σ_d^2	variância do sinal desejado, p. 10
σ_n^2	variância do ruído, p. 19
EMSE	excesso de MSE, p. 35
M	desajuste (em inglês, <i>misadjustment</i>), p. 35
$\text{diag}\{\mathbf{A}\}$	vetor formado pelos elementos da diagonal principal da matriz \mathbf{A} , p. 6
$\tilde{\mathbf{e}}(k)$	vetor erro-sem-ruído <i>a priori</i> , p. 38
$\tilde{\mathbf{e}}(k)$	vetor erro-sem-ruído <i>a posteriori</i> , p. 38
$\tilde{e}(k)$	erro-sem-ruído <i>a posteriori</i> , p. 29
$\tilde{e}_0(k)$	primeiro elemento do vetor erro-sem-ruído <i>a priori</i> , p. 35
$\boldsymbol{\varepsilon}(k)$	vetor erro <i>a posteriori</i> , p. 33
$\mathbf{w}(k)$	vetor de coeficientes, p. 8

\mathbf{w}_o	vetor de coeficientes dado pela solução de Wiener, p. 14
$\mathbf{x}(k)$	vetor de entrada (ou regressor), p. 8
$\mathbf{y}(k)$	vetor saída, p. 15
$d(k)$	sinal desejado, p. 8
$e(k)$	sinal erro, p. 9
$e_0(k)$	primeiro elemento do vetor $\mathbf{e}(k)$, p. 33
$e_o(k)$	erro existente quando os coeficientes do filtro adaptativo são dados pela solução de Wiener, p. 25
$n(k)$	ruído, p. 8
$y(k)$	saída do filtro, p. 8
$P[\cdot]$	operador de probabilidade (medida de probabilidade), p. 5
$E[\cdot]$	operador valor esperado, p. 5
$\mathcal{H}(k)$	conjunto restrito considerando-se o par $(\mathbf{x}(k), d(k))$, p. 18
$\text{cov}[\Delta\mathbf{w}(k)]$	matriz de covariância de $\Delta\mathbf{w}(k)$, p. 26

Lista de Abreviaturas

ADD	assunção de diagonal dominante, p. 45
AP	<i>affine projection</i> , p. 2
BNLMS	<i>binormalized least-mean-square</i> , p. 2
DS-CDMA	<i>direct-sequence code-division multiple access</i> , p. 2
EDO	equação diferencial ordinária, p. 5
ED	equação a diferenças, p. 5
EMSE	excesso de MSE, p. 3
FAP	<i>fast AP</i> , p. 16
LMF	<i>least-mean fourth</i> , p. 11
LMS	<i>least-mean-square</i> , p. 4
MMSE	mínimo MSE, p. 7
MSE	<i>mean-square error</i> (erro quadrático médio), p. 3
NLMS	<i>normalized least-mean-square</i> , p. 2
OBE	<i>optimal bounding ellipsoids</i> , p. 17
RLS	<i>recursive least-squares</i> , p. 7
SM-AP	<i>set-membership affine projection</i> , p. 2
SM-BNLMS	<i>set-membership binormalized least-mean-square</i> , p. 2
SM-NLMS	<i>set-membership normalized least-mean-square</i> , p. 2

SMF	<i>set-membership filtering</i> , p. 17
SM	<i>set-membership</i> , p. 4
WSS	<i>wide sense stationary</i> (estacionário no sentido amplo), p. 10
i.i.d.	independente e identicamente distribuído, p. 34

Capítulo 1

Introdução

Um problema típico de processamento digital de sinais é projetar um filtro digital que possua um conjunto de características desejáveis, isto é, que atenda determinadas especificações. Esse problema aparece em várias situações tais como [1, 2, 3]: em processamento de imagens utilizam-se filtros digitais para realçar uma imagem, em sistemas de comunicações tem-se os filtros formatadores de pulso responsáveis por modelar o sinal transmitido de forma que ele não interfira em outros sistemas de comunicações que estejam utilizando bandas adjacentes, em bancos de filtros para comprimir sinais, etc.

No entanto, existem muitas aplicações em que não dispomos das especificações do filtro ou elas são variantes no tempo (há inclusive casos em que essas duas situações ocorrem ao mesmo tempo) [4]. Nesses casos, torna-se necessário utilizar um filtro especial, capaz de aprender/capturar as suas características desejáveis de forma iterativa¹. A esse filtro especial dá-se o nome de filtro adaptativo.

Filtragem adaptativa faz-se presente em nosso cotidiano em diversas situações. Em sistemas de comunicações, por exemplo, o sinal transmitido é distorcido pelo canal e chega corrompido ao receptor. No receptor, pode-se utilizar um equalizador adaptativo com a finalidade de minimizar os efeitos do canal sobre o sinal transmitido [4].

Outro exemplo clássico onde filtros adaptativos são utilizados é no cancelamento

¹Optou-se por utilizar o termo “de forma iterativa” ao invés de “ao longo do tempo” pois, de forma geral, filtros adaptativos podem ser utilizados em aplicações nas quais o aprendizado ocorre ao longo de outra grandeza que não o tempo. Em processamento de imagem, por exemplo, o aprendizado é feito ao longo do espaço, conforme a imagem vai sendo varrida.

de eco, seja ele elétrico ou acústico [5]. A telefonia fixa é um exemplo de sistema em que há eco elétrico, gerado por um equipamento denominado híbrida. Atualmente, esse tipo de problema é praticamente imperceptível graças aos filtros adaptativos. Já o eco acústico surge devido à existência de um caminho de retorno entre o sinal que sai de uma caixa de som e um microfone, fazendo com que o usuário receba uma versão de seu próprio sinal transmitido de volta. Esse problema ocorre em sistemas de teleconferência e sua solução é bem mais difícil, pois o ambiente de teleconferência (uma sala) proporciona múltiplos percursos para que o sinal que sai da caixa retorne pelo microfone. Em outras palavras, a resposta ao impulso das salas costuma ser bem longa, da ordem de milhares de coeficientes, além de que o sinal realimentado é correlacionado com o sinal de voz do outro usuário.

Enfim, outras aplicações de filtros adaptativos, além das citadas acima, podem ser encontradas em [4, 6, 7].

Portanto, dada a gama de situações nas quais filtragem adaptativa se aplica, cabe ao projetista conhecer os diferentes algoritmos adaptativos de forma que ele possa escolher aquele que melhor atenda às suas necessidades. Dentre esses algoritmos, o *set-membership affine projection* (SM-AP) destaca-se por possuir graus de liberdade que definem sua velocidade de convergência, complexidade computacional e desajuste. Além disso, o SM-AP inclui diversos outros algoritmos como casos particulares, tais como o *set-membership normalized least-mean-square* (SM-NLMS), o *set-membership binormalized least-mean-square* (SM-BNLMS), o *affine projection* (AP) sem coeficiente de relaxamento, e outros algoritmos que o AP abrange, como por exemplo, o *normalized least-mean-square* (NLMS), o *binormalized least-mean-square* (BNLMS), dentre outros.

Diversas publicações têm confirmado o poder do algoritmo SM-AP em diferentes aplicações. Em [5, 8] o algoritmo SM-NLMS mostrou-se superior aos algoritmos tradicionalmente utilizados em cancelamento de eco. Em [9], o SM-AP (e os algoritmos que ele abrange) proporcionou melhores resultados que os algoritmos comumente utilizados para supressão de interferência em um cenário de comunicação baseado em DS-CDMA (*direct-sequence code-division multiple access*). Em [10, 11] o algoritmo SM-AP é utilizado em um cenário de equalização semi-supervisionada e mostra um desempenho superior ao do clássico *constant modulus* no que diz respeito a taxa de

erro de *bits*; além disso, o número de atualizações realizadas pelo SM-AP é bastante inferior. Uma variante do SM-AP é utilizada em [12] para controle ativo de ruído.

Contudo, mesmo com todo o sucesso que o SM-AP vem tendo, ainda não existe nenhuma análise para esse algoritmo; possivelmente porque ele é um algoritmo relativamente novo (surgiu em 2001 [13]) e bastante complexo de se analisar. O mais próximo que se encontra na literatura são dois trabalhos relacionados com algoritmos que são casos particulares do SM-AP [14, 15]. Em [14] é apresentada uma análise para o algoritmo SM-BNLMS, porém ela se baseia em modelos simplificados para o sinal de entrada do filtro adaptativo. Em [15], a principal contribuição é uma equação cuja solução fornece o erro quadrático médio (MSE, do inglês *mean-square error*) após convergência relativo ao algoritmo SM-NLMS. O problema dessa abordagem é que a equação derivada é bastante complexa de forma que pouco pode ser extraído a respeito da influência das variáveis (mencionadas anteriormente como os graus de liberdade do SM-AP). Além disso, o MSE teórico derivado em [15] apresenta inconsistência com a prática².

Recentemente, resultados referentes ao MSE em ambiente não-estacionário para uma versão simplificada do SM-AP, conhecida como *simple-choice* [4] ou também SM-AP simplificado³, foram exibidos em [16]. Porém, as derivações foram omitidas.

Este trabalho visa cobrir essa lacuna existente na literatura.

1.1 Objetivo

O objetivo desta dissertação é analisar o algoritmo *set-membership affine projection* (SM-AP) quanto ao seu desempenho de erro quadrático médio (MSE, do inglês *mean-square error*) em estado estacionário, isto é, após a convergência do algoritmo. Mais especificamente, pretende-se derivar expressões para o excesso de MSE (EMSE) e para o desajuste que revelem como os graus de liberdade presentes no SM-AP se relacionam, e dessa forma, ajudar ao projetista.

²Esse problema será explicitado no Capítulo 5 quando o leitor já tiver sido apresentado ao SM-AP e estivermos discutindo os resultados de simulações.

³O SM-AP simplificado é um caso particular do SM-AP no qual o algoritmo atualiza os coeficientes do filtro adaptativo seguindo unicamente a direção do vetor de entrada mais atual. Para maiores detalhes, vide Capítulo 6 de [4].

1.2 Motivação

A idéia de analisar o algoritmo SM-AP surgiu devido a muitos fatores:

1. SM-AP é um algoritmo bastante rico e complexo, que oferece ao projetista mais graus de liberdade do que os algoritmos tradicionais. Entender como essas variáveis se relacionam é uma tarefa interessante e desafiadora;
2. O SM-AP combina duas características importantes: a seleção de dados que faz com que os recursos computacionais sejam poupados quando os dados de entrada não trazem inovação, e o reuso de dados que representa um compromisso entre a velocidade de convergência e a complexidade computacional por atualização do algoritmo;
3. Esse algoritmo generaliza diversos outros, tais como o SM-NLMS, SM-BNLMS, AP, BNLMS e NLMS, dentre outros;
4. Publicações vem demonstrando o potencial do SM-AP em uma grande variedade de aplicações.

1.3 Organização da Dissertação

A presente dissertação está estruturada da seguinte forma.

No Capítulo 2 relembramos alguns aspectos de filtragem adaptativa bem como alguns dos principais algoritmos adaptativos. O objetivo desse capítulo é apresentar o algoritmo SM-AP utilizando uma abordagem simples. Acreditamos que uma breve exposição a alguns algoritmos menos complexos pode facilitar a compreensão e apreciação das principais características do SM-AP. Esses algoritmos menos complexos são:

- LMS (do inglês, *least-mean-square*);
- NLMS (do inglês, *normalized least-mean-square*);
- AP (do inglês, *affine projection*) e
- SM-NLMS (do inglês, *set-membership normalized least-mean-square*).

O Capítulo 3 apresenta as principais técnicas utilizadas para analisar algoritmos adaptativos quanto ao seu desempenho de MSE. São elas: equação a diferenças (ED), equação diferencial ordinária (EDO), e conservação de energia. Atenção especial é dada ao método de conservação de energia, pois ele facilita a análise do SM-AP (o motivo ficará claro posteriormente).

No Capítulo 4 exibimos de forma detalhada a análise do algoritmo SM-AP. Inicialmente, introduzimos (e justificamos) um modelo para o algoritmo SM-AP sem a presença da não-linearidade do erro⁴. Em seguida, utilizamos o método de conservação de energia para analisar o algoritmo. Ao longo de nossa análise, algumas suposições serão feitas. Algumas serão perfeitamente válidas, enquanto outras, embora apenas aproximações, ainda assim serão feitas para que consigamos chegar em fórmulas fechadas para o excesso de erro quadrático médio e desajuste desse algoritmo. Este capítulo constitui a principal contribuição desta dissertação.

O Capítulo 5 apresenta os resultados de simulações comparando os desempenhos de MSE, prático e teórico, referentes ao SM-AP, considerando diferentes tipos de entrada do filtro adaptativo. Neste capítulo, ficará claro uma das grandes vantagens de nossa análise: não assumir um modelo para o sinal de entrada do filtro adaptativo.

Finalmente, os Capítulos 6 e 7 encerram esta dissertação apresentando as conclusões e os trabalhos futuros, respectivamente.

1.4 Notação

Nesta dissertação utilizamos letras minúsculas (por exemplo, x) para denotar variáveis escalares, vetores são denotados por letras minúsculas em negrito (por exemplo, \mathbf{x}), e matrizes são denotadas por letras maiúsculas em negrito (por exemplo, \mathbf{X}). Todos os vetores são vetores coluna. Denotamos o corpo dos números reais por \mathbb{R} , e utilizamos \mathbb{R}_+ para representar o conjunto dos números reais não-negativos. Dessa forma, o produto interno entre dois vetores $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$, denotado por $\langle \mathbf{x}, \mathbf{y} \rangle$, pode ser escrito como $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$, onde $(\cdot)^T$ é a operação de transposição aplicada a (\cdot) . A norma euclidiana de um vetor é dada por $\|\cdot\|$. Também definimos $E[\cdot]$, $\text{tr}\{\cdot\}$, e $P[\cdot]$ como os operadores valor esperado, traço, e probabilidade, respectiva-

⁴Essa não-linearidade é inerente aos algoritmos do tipo SM.

mente. O operador $\text{diag}\{\mathbf{A}\}$ aplicado à matriz \mathbf{A} gera um vetor cujos elementos são a diagonal principal de \mathbf{A} . Por fim, \mathbf{I} é a matrix identidade, e $\mathbf{0}$ representa um vetor (ou uma matriz) cujos elementos são todos iguais a zero.

Capítulo 2

Filtragem Adaptativa

Filtros adaptativos, como o próprio nome sugere, são filtros digitais capazes de auto-ajustar seus coeficientes, isto é, sua função de transferência, ao longo das iterações. A forma como os coeficientes são atualizados é determinada pelo algoritmo adaptativo [7, 17] e depende do sinal de entrada. Logo, filtros adaptativos são sistemas não-lineares e variantes no tempo. Contudo, para a grande maioria dos filtros, o mapeamento entrada-saída em uma determinada iteração é linear, ou seja, o princípio da superposição [1] vale quando analisamos uma dada iteração. Filtros com essa característica são conhecidos, devido a um abuso de linguagem, como filtros adaptativos lineares [7].

Filtragem adaptativa linear tem suas raízes em estimação linear. Os primeiros estudos envolvendo estimação linear foram realizados por astrônomos, com destaque para Galileo, por volta de 1632, e visavam compreender o movimento dos astros celestes. No entanto, o surgimento da teoria de estimação linear é creditada a Gauss, inventor do método de mínimos quadrados,¹ em 1795. O método de mínimos quadrados constitui, em filtragem adaptativa, uma importante vertente que deu origem a uma famosa família de algoritmos conhecida como RLS (*recursive least-squares*) [7].

Outra importante vertente é o método de estimação por mínimo erro quadrático médio (MMSE, mínimo MSE) em processos estocásticos, cujos primeiros estudos

¹A origem desse método foi, inicialmente, contestada por alguns, pois embora Gauss o tenha descoberto primeiro, ele não o publicou. Foi Legendre, em 1805, quem derivou esse mesmo método, de forma independente, e o publicou.

foram realizados por matemáticos renomados, dentre eles: Kolmogorov (1939) e Wiener (1949) [7]. O MMSE deu origem a uma família de algoritmos adaptativos conhecida como gradiente estocástico. Dentre os algoritmos pertencentes a essa família está o famoso LMS.

Iniciamos este capítulo apresentando, na Seção 2.1, conceitos básicos de filtragem adaptativa. O objetivo dessa seção é estabelecer a notação e relações entre as principais variáveis que aparecem no contexto de filtragem adaptativa, e que serão extensivamente utilizadas ao longo da dissertação. Na Seção 2.2 apresentamos alguns algoritmos do tipo gradiente estocástico e, na Seção 2.3 apresentamos os algoritmos da família SM. A motivação para essa abordagem é que os algoritmos da família SM que serão apresentados são baseados em algoritmos da família gradiente estocástico.

2.1 Conceitos Básicos

Antes de apresentar os algoritmos, é importante definir a notação que será utilizada ao longo da dissertação. Cabe salientar que a estrutura do filtro adaptativo considerada é sempre a transversal, também conhecida como *tapped-delay line* [4, 7].

A Figura 2.1 ilustra as principais variáveis envolvidas no processo de filtragem adaptativa, em uma dada iteração k . Definimos o sinal

$$\mathbf{x}(k) = [x(k) \ x(k-1) \ \dots \ x(k-N)]^T \in \mathbb{R}^{N+1} \quad (2.1)$$

como o vetor de entrada do filtro adaptativo, também conhecido como regressor [6], onde N é a ordem do filtro. Os coeficientes do filtro adaptativo estão representados por

$$\mathbf{w}(k) = [w_0(k) \ w_1(k) \ \dots \ w_N(k)]^T \in \mathbb{R}^{N+1} \quad (2.2)$$

que, para sermos sucintos, denominaremos de vetor de coeficientes. A saída do filtro adaptativo é

$$y(k) = \mathbf{w}^T(k)\mathbf{x}(k) \in \mathbb{R} \quad (2.3)$$

e o sinal de referência $d(k) \in \mathbb{R}$, o qual chamaremos de sinal desejado, costuma se apresentar, em situações práticas, corrompido por algum tipo de ruído $n(k) \in \mathbb{R}$.

Por fim, comparando-se o sinal desejado com a saída do filtro temos o sinal erro

$$e(k) = d(k) - y(k) \in \mathbb{R} \quad (2.4)$$

que é utilizado para atualizar o vetor de coeficientes conforme uma regra determinada pelo algoritmo adaptativo.

2.2 Gradiente Estocástico

Os algoritmos adaptativos do tipo gradiente estocástico são baseados em métodos de otimização conhecidos como métodos de gradiente. Esses métodos recebem esse nome porque a minimização ocorre seguindo a direção contrária à do gradiente, ou de uma função do gradiente. Por exemplo, o método de máxima descida (*steepest descent*) segue a direção contrária à do gradiente, enquanto o método de Newton segue a direção contrária à do vetor definido como o produto da matriz Hessiana vezes o gradiente [18]. Métodos de gradiente são bastante simples, porém só convergem para o mínimo global quando a função objetivo é convexa [18, 19, 20].

Considerando as observações acima, o problema de otimização mais comum em filtragem adaptativa é o de encontrar um $\mathbf{w}(k)$ que minimize a função objetivo $E[e^2(k)]$, isto é, o MSE. De fato, utilizando as relações apresentadas na Seção 2.1 temos

$$\begin{aligned} E[e^2(k)] &= E[(d(k) - y(k))^2] \\ &= E[(d(k) - \mathbf{w}^T(k)\mathbf{x}(k))^2] \\ &= E[d^2(k) - 2d(k)\mathbf{w}^T(k)\mathbf{x}(k) + \mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)] \end{aligned} \quad (2.5)$$

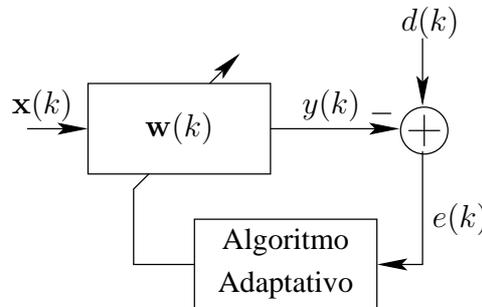


Figura 2.1: Modelo básico de operação do filtro adaptativo.

Assumindo que $\mathbf{x}(k)$ é estacionário no sentido amplo [21] (WSS, do inglês *wide sense stationary*) e $d(k)$ é também WSS com média 0 e variância σ_d^2 , podemos re-escrever a equação acima como

$$\mathbb{E} [e^2(k)] = \sigma_d^2 - 2\mathbf{w}^T(k)\mathbf{p} + \mathbf{w}^T(k)\mathbf{R}\mathbf{w}(k) \quad (2.6)$$

onde \mathbf{p} é o vetor de correlação-cruzada entre o sinal desejado e o vetor de entrada, e \mathbf{R} é a matriz de auto-correlação do vetor de entrada. Suas definições são dadas por

$$\mathbf{p} = \mathbb{E} [d(k)\mathbf{x}(k)] \quad (2.7)$$

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

Pode-se observar que o MSE é uma função quadrática em $\mathbf{w}(k)$, vide equação (2.6), e a sua matriz Hessiana é dada por \mathbf{R} , definida na equação (2.8), que é uma matriz simétrica e semi-definida positiva [22]. Essas duas características estabelecem que a superfície de MSE é um parabolóide convexo [18] e, portanto, podemos garantir que os métodos de gradiente convergem para o mínimo global, sendo então desnecessário utilizar técnicas de otimização mais elaboradas e computacionalmente mais custosas [18].

Uma vez definido o MSE como função objetivo, o gradiente com respeito a $\mathbf{w}(k)$ é

$$\mathbf{g}(k) = \left[\frac{\partial \mathbb{E} [e^2(k)]}{\partial \mathbf{w}(k)} \right]^T = -2\mathbf{p} + 2\mathbf{R}\mathbf{w}(k) \in \mathbb{R}^{N+1} \quad (2.9)$$

e a solução iterativa seguindo o método de máxima descida é

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu\mathbf{g}(k) \\ &= \mathbf{w}(k) - \mu(-2\mathbf{p} + 2\mathbf{R}\mathbf{w}(k)) \end{aligned} \quad (2.10)$$

onde μ é conhecido como fator de convergência ou tamanho do passo.

Contudo, como podemos notar pela equação (2.10), a minimização do MSE utilizando o método de máxima descida requer algum conhecimento estatístico (no caso, conhecimento de \mathbf{R} e \mathbf{p}) do problema, que, em geral, não temos. Dessa forma, o que faremos nas subseções seguintes é apresentar um conjunto de algoritmos adaptativos que seguem a direção de um vetor $\hat{\mathbf{g}}(k)$ que é uma aproximação instantânea (no sentido de só utilizar uma realização do processo estocástico) do verdadeiro gradiente $\mathbf{g}(k)$ [4].

Como observação final, gostaríamos de salientar que existem algoritmos adaptativos baseados em outros métodos de gradiente (vide [4, 6, 7]). Porém, para a nossa finalidade aqui, só estamos interessados nos algoritmos baseados no de máxima descida. Analogamente, existem algoritmos do tipo gradiente estocástico que minimizam outras funções objetivo, diferentes do MSE, como por exemplo o LMF (*least-mean fourth*) cuja função objetivo relacionada é $E[e^4(k)]$ [23].

2.2.1 LMS

O algoritmo LMS (*least-mean-square*) surgiu em 1960 com Bernard Widrow e seu aluno de doutorado Marcian E. Hoff [24], e marca o surgimento do campo atualmente conhecido como processamento adaptativo de sinais, ou simplesmente, filtragem adaptativa. Na época, o algoritmo foi apresentado com o nome Adaline (*adaptive linear*). Algumas das clássicas referências sobre o LMS são [25, 26], porém existem diversas fontes onde o LMS é coberto com profundidade, por exemplo [4, 6, 7].

O LMS realiza a seguinte aproximação instantânea para o gradiente [4]

$$\hat{\mathbf{g}}(k) = -2\hat{\mathbf{p}}(k) + 2\hat{\mathbf{R}}(k)\mathbf{w}(k)$$

onde

$$\hat{\mathbf{p}}(k) = d(k)\mathbf{x}(k) \quad (2.11)$$

$$\hat{\mathbf{R}}(k) = \mathbf{x}(k)\mathbf{x}^T(k) \quad (2.12)$$

Logo, podemos re-escrever o gradiente aproximado como

$$\begin{aligned} \hat{\mathbf{g}}(k) &= -2(d(k)\mathbf{x}(k)) + 2(\mathbf{x}(k)\mathbf{x}^T(k))\mathbf{w}(k) \\ &= -2(d(k) - \mathbf{x}^T(k)\mathbf{w}(k))\mathbf{x}(k) \\ &= -2e(k)\mathbf{x}(k) \end{aligned} \quad (2.13)$$

Note que $\hat{\mathbf{g}}(k)$ é uma variável aleatória, pois depende das variáveis aleatórias $e(k)$ e $\mathbf{x}(k)$, daí o nome gradiente estocástico. Definida a aproximação para o gradiente, a equação de atualização do LMS é obtida

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu\hat{\mathbf{g}}(k) \\ &= \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k) \end{aligned} \quad (2.14)$$

Vale ressaltar que o problema de otimização que é efetivamente resolvido pelo LMS, isto é, a função cujo gradiente resulta em $\hat{\mathbf{g}}(k)$, é [4]

$$\min_{\mathbf{w}(k)} e^2(k) \quad (2.15)$$

isto é, minimizar o erro quadrático (instantâneo) em função de $\mathbf{w}(k)$.

A seguir, descrevemos o algoritmo LMS [4]. O algoritmo LMS, juntamente com sua versão normalizada (NLMS, que será apresentado na próxima subseção), são os algoritmos mais utilizados na prática, pois eles são eficientes e simples.

Algoritmo *Least-Mean-Square*

Etapa 1. Inicialização:

$$\mathbf{x}(0) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$$

escolha μ satisfazendo $0 < \mu < \frac{1}{\text{tr}\{\mathbf{R}\}}$ ²

Etapa 2. Recursão:

Para $k \geq 0$ faça

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k)$$

2.2.2 NLMS

O algoritmo NLMS (*normalized least-mean-square*) foi proposto, de forma independente, por [27] e [28], ambos em 1967. A princípio esse algoritmo recebeu outros nomes, o mais conhecido sendo: *learning identification*. Aparentemente, o nome NLMS foi adotado por volta de 1980 graças aos trabalhos [29, 30].

A motivação do NLMS foi superar um problema presente no LMS conhecido como amplificação de ruído do gradiente (*gradient noise amplification* [7]). Esse problema ocorre porque a aproximação do gradiente utilizada no LMS é proporcional ao vetor de entrada $\mathbf{x}(k)$ e, conseqüentemente, quando a norma desse vetor é grande, a atualização também o será, porém em uma direção que não é a “correta”, isto é, diferente da direção de $-\mathbf{g}(k)$.

Portanto, a proposta do NLMS é modificar a atualização do algoritmo LMS de forma a torná-lo menos sensível ao tamanho do vetor de entrada. Com esse intuito,

²Essas restrições para escolha de μ decorrem da análise de convergência do LMS (vide subseções 3.3.2 e 3.3.3 do livro [4]). Na prática, costuma-se escolher um μ pequeno, pois não conhecemos \mathbf{R} .

a aproximação para o gradiente adotada pelo NLMS é

$$\hat{\mathbf{g}}(k) = -\frac{e(k)}{\|\mathbf{x}(k)\|^2}\mathbf{x}(k) \quad (2.16)$$

e a equação de atualização é

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu\hat{\mathbf{g}}(k) \\ &= \mathbf{w}(k) + \frac{\mu}{\|\mathbf{x}(k)\|^2}e(k)\mathbf{x}(k) \end{aligned} \quad (2.17)$$

O problema de otimização relacionado com o algoritmo NLMS com $\mu = 1$, é

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2}\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & d(k) - \mathbf{w}^T(k+1)\mathbf{x}(k) = 0 \end{aligned} \quad (2.18)$$

O problema de otimização descrito acima informa que desejamos encontrar um $\mathbf{w}(k+1)$ mais próximo possível a $\mathbf{w}(k)$ (conhecido na literatura como critério de mínima perturbação), de forma que o erro *a posteriori*³ seja 0. Em outras palavras, definindo o hiper-plano onde o erro *a posteriori* é nulo

$$\Pi_k = \{\mathbf{w} \in \mathbb{R}^{N+1} : d(k) - \mathbf{w}^T\mathbf{x}(k) = 0\} \quad (2.19)$$

a atualização $\mathbf{w}(k+1)$ será uma projeção de $\mathbf{w}(k)$ em Π_k , pois é a atualização de menor norma (mínima perturbação), considerando $\mu = 1$. Isso também nos diz que $\mathbf{x}(k) \perp \Pi_k$, pois a direção de atualização é dada por $\mathbf{x}(k)$, vide equação (2.17). A Figura 2.2 resume a interpretação geométrica da atualização do NLMS no \mathbb{R}^3 . Note que, geometricamente, a influência de μ no processo de atualização é:

³Erro calculado a partir dos dados atuais, porém com o vetor de coeficientes já atualizados.

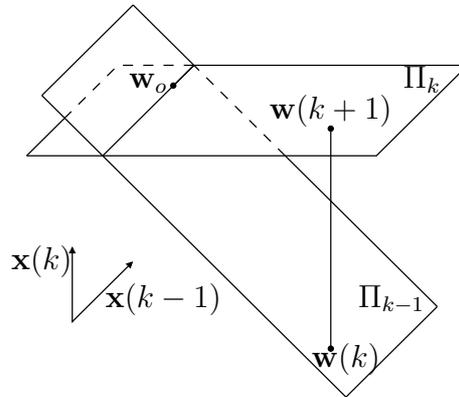


Figura 2.2: Interpretação geométrica da atualização do NLMS no espaço dos coeficientes.

- $\mu = 0 \rightarrow \mathbf{w}(k+1) = \mathbf{w}(k)$;
- $0 < \mu < 1 \rightarrow \mathbf{w}(k+1) \notin \Pi_k$, nem chega até o hiper-plano Π_k ;
- $\mu = 1 \rightarrow \mathbf{w}(k+1) \in \Pi_k$;
- $1 < \mu < 2 \rightarrow \mathbf{w}(k+1) \notin \Pi_k$, passa do hiper-plano Π_k ;
- $\mu \geq 2 \rightarrow \|\mathbf{w}_o - \mathbf{w}(k+1)\|^2 \geq \|\mathbf{w}_o - \mathbf{w}(k)\|^2$, isto é, tende a se distanciar de \mathbf{w}_o (algoritmo não converge);

onde \mathbf{w}_o são os coeficientes ótimos do filtro adaptativo dados pela solução de Wiener.

Portanto, para que o NLMS convirja, devemos escolher $\mu \in (0, 2)$. Na prática, escolhe-se $\mu \in (0, 1]$, pois os resultados obtidos para $\mu \in (1, 2)$ podem ser obtidos no intervalo $\mu \in (0, 1)$, vide Figura 2.2, e além disso, escolher $\mu \in (1, 2)$ gera uma amplificação desnecessária do ruído.

A seguir apresentamos o algoritmo NLMS com fator de regularização δ [4]. Esse fator de regularização costuma ser escolhido como uma constante muito pequena cuja finalidade é evitar problemas numéricos que possam ocorrer quando $\|\mathbf{x}(k)\|^2 \rightarrow 0$.

Algoritmo *Normalized Least-Mean-Square*

Etapa 1. Inicialização:

$$\mathbf{x}(0) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$$

escolha μ satisfazendo $0 < \mu < 2$

Etapa 2. Recursão:

Para $k \geq 0$ faça

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu}{\delta + \|\mathbf{x}(k)\|^2} e(k)\mathbf{x}(k)$$

2.2.3 AP: Reuso de Dados

O algoritmo AP (*affine projection*) surgiu em 1984, com o trabalho [31], e visava aumentar a velocidade de convergência do NLMS, especialmente em situações nas quais o sinal de entrada é colorido.

Analisando a Figura 2.2, Ozeki e Umeda notaram que a velocidade de convergência do algoritmo poderia aumentar se a atualização fosse uma projeção em

$\Pi_{k-1} \cap \Pi_k$. Note que assim como $\mathbf{x}(k) \perp \Pi_k$, temos $\mathbf{x}(k-1) \perp \Pi_{k-1}$ e, portanto, a nova direção de atualização proposta em [31] é uma combinação linear de $\mathbf{x}(k)$ e $\mathbf{x}(k-1)$, vide Figura 2.3.

Na Figura 2.3, temos um exemplo de atualização no espaço dos coeficientes $\mathbf{w}(k) \in \mathbb{R}^3$ (ou seja, a ordem do filtro é $N = 2$) realizada pelo algoritmo AP com reuso de dados $L = 1$ e com $\mu = 1$.

Generalizando essa idéia, podemos adotar

$$\hat{\mathbf{g}}(k) = -\mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k)]^{-1} \mathbf{e}(k) \quad (2.20)$$

onde a matriz $\mathbf{X}(k) \in \mathbb{R}^{N+1 \times L+1}$ é a matriz de entrada que contém o vetor atual de entrada $\mathbf{x}(k)$ e L vetores de entrada passados

$$\mathbf{X}(k) = [\mathbf{x}(k) \quad \mathbf{x}(k-1) \quad \dots \quad \mathbf{x}(k-L)] \quad (2.21)$$

O vetor erro $\mathbf{e}(k) \in \mathbb{R}^{L+1}$ é definido como

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{y}(k) \quad (2.22)$$

onde o vetor desejado e o vetor saída são definidos, respectivamente, como

$$\mathbf{d}(k) = [d(k) \quad d(k-1) \quad \dots \quad d(k-L)]^T \quad (2.23)$$

$$\mathbf{y}(k) = \mathbf{X}^T(k)\mathbf{w}(k) \quad (2.24)$$

Dadas as definições acima, a equação de atualização do AP é

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) - \mu \hat{\mathbf{g}}(k) \\ &= \mathbf{w}(k) + \mu \mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k)]^{-1} \mathbf{e}(k) \end{aligned} \quad (2.25)$$

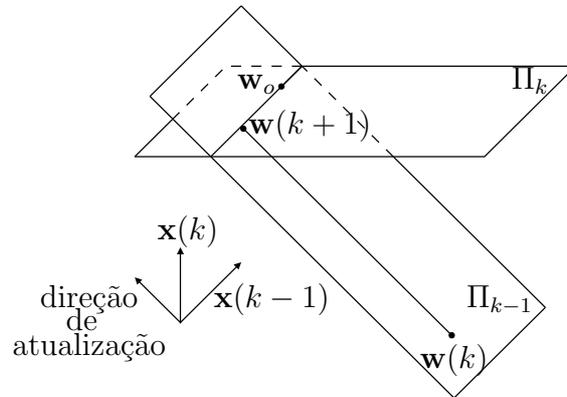


Figura 2.3: Interpretação geométrica da atualização do AP no espaço dos coeficientes, considerando $N = 2$ e $L = 1$.

O problema de otimização relacionado com o algoritmo AP com $\mu = 1$, é

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k+1) = 0 \end{aligned} \quad (2.26)$$

Analogamente ao algoritmo NLMS, para garantirmos a convergência do AP devemos escolher o fator de convergência respeitando $0 < \mu < 2$, e na prática utiliza-se $\mu \in (0, 1]$.

A seguir, apresentamos o algoritmo AP com fator de regularização δ [4]. Esse fator de regularização costuma ser escolhido como uma constante muito pequena cuja finalidade é evitar problemas numéricos que possam ocorrer devido a inversão da matriz $\mathbf{X}^T(k)\mathbf{X}(k)$.

Algoritmo *Affine Projection*

Etapa 1. Inicialização:

$$\mathbf{x}(0) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$$

escolha μ satisfazendo $0 < \mu < 2$

Etapa 2. Recursão:

Para $k \geq 0$ faça

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1} \mathbf{e}(k)$$

2.2.4 Considerações Finais

O algoritmo NLMS pode ser obtido a partir do AP fazendo-se $L = 0$. Analogamente, o AP com $L = 1$ recai num caso particular chamado BNLMS. Sabe-se que a velocidade de convergência desse algoritmo aumenta com L , porém o seu desajuste também aumenta [4, 32]. Além disso, o número de multiplicações realizadas por atualização do vetor de coeficientes é avaliado em [33]. Obviamente, para o AP, devido à necessidade de inverter uma matriz, esse número de multiplicações cresce com L de forma cúbica (L^3). Também em [33], é apresentada uma implementação rápida para o AP, denominada de FAP (*fast affine projection*). O FAP é baseado numa técnica conhecida como janela deslizante (*sliding window*) utilizada em [34] para o RLS.

Existem trabalhos, por exemplo [35] e referências nesse trabalho, que propõem μ variável para o AP de forma que o algoritmo apresente uma alta velocidade de convergência ($\mu \rightarrow 1$) no início da operação, porém um desajuste pequeno após a convergência ($\mu \rightarrow 0$). Analogamente, existem trabalhos que buscam encontrar um δ que aumente a velocidade de convergência do AP (δ pequeno), porém sem piorar o seu desajuste após convergência (δ maior), por exemplo [36] e referências nele.

2.3 *Set-Membership Filtering*: Seleção de Dados

A família SM (*set-membership*) de algoritmos adaptativos segue a filosofia *set-membership filtering* (SMF) que surgiu com o trabalho de Huang e seus alunos [37], em 1998. Ao contrário da abordagem conhecida como *set-membership identification* [38, 39], que assume a existência de um limite superior para o ruído, a abordagem SMF assume um limite superior na variável erro e , conseqüentemente, pode ser aplicada em uma variedade de aplicações, inclusive em identificação de sistema [37].

A filosofia SMF pode ser implementada de diferentes formas. Dentre elas, estão as implementações baseadas nos métodos de otimização conhecidos como *optimal bounding ellipsoids* (OBE) [40]. Esses métodos de OBE são interessantes pois [37]:

- Realizam a estimação de uma região de possíveis soluções, ao invés de estimar um único ponto;
- Exibem boas propriedades de convergência e rastreamento;
- Existe prova de convergência para a maioria dos OBE [38, 41];
- Realizam a seleção de dados, isto é, só atualizam os coeficientes se os dados de entrada trouxerem alguma inovação.

Os algoritmos da família SM possuem dois estágios:

1. Avaliação da informação (*information evaluation* ou também *innovation check*): checa-se o quanto de inovação existe no sinal erro;
2. Atualização: caso a quantidade de inovação presente no sinal erro seja superior a um determinado valor, efetua-se a atualização dos coeficientes do filtro.

Essa característica confere baixo desajuste e número de atualizações reduzido (economiza recursos computacionais) aos algoritmos do tipo SM [4, 14, 37, 42].

O objetivo da abordagem SMF é encontrar o conjunto viável Θ , definido como

$$\Theta = \bigcap_{(\mathbf{x}, d) \in \mathcal{S}} \{ \mathbf{w} \in \mathbb{R}^{N+1} : |d - \mathbf{w}^T \mathbf{x}| \leq \bar{\gamma} \} \quad (2.27)$$

onde $\bar{\gamma} \in \mathbb{R}_+$ é o limite superior aplicado no módulo do sinal erro, \mathcal{S} é o conjunto formado por todos os possíveis pares de entrada (\mathbf{x}, d) . Na prática, isso é feito iterativamente conforme descrito a seguir.

Considere os dados de entrada $(\mathbf{x}(i), d(i))$, para $i = 0, 1, \dots, k$, e seja

$$\mathcal{H}(i) = \{ \mathbf{w} \in \mathbb{R}^{N+1} : |d(i) - \mathbf{w}^T \mathbf{x}(i)| \leq \bar{\gamma} \} \quad (2.28)$$

o conjunto restrito de soluções em uma dada iteração i . Pode-se obter um conjunto $\psi(k)$, denominado de *exact membership set*, que é uma estimativa de Θ , da seguinte forma

$$\psi(k) = \bigcap_{i=0}^k \mathcal{H}(i) \quad (2.29)$$

É claro que, como $\psi(k)$ não leva em consideração todos os possíveis pares de entrada, temos $\Theta \subset \psi(k)$. Note que $\mathcal{H}(i)$ é uma região entre dois hiper-planos, e $\psi(k)$ é um poliedro formado pela interseção de todos esses hiper-planos, até a iteração k . A interpretação geométrica do SMF é resumida na Figura 2.4.

Na prática, $\bar{\gamma}$ costuma ser escolhido de forma a modelar a incerteza que temos em torno de \mathbf{w}_o devido à presença de ruído $n(k)$. Portanto, costuma-se es-

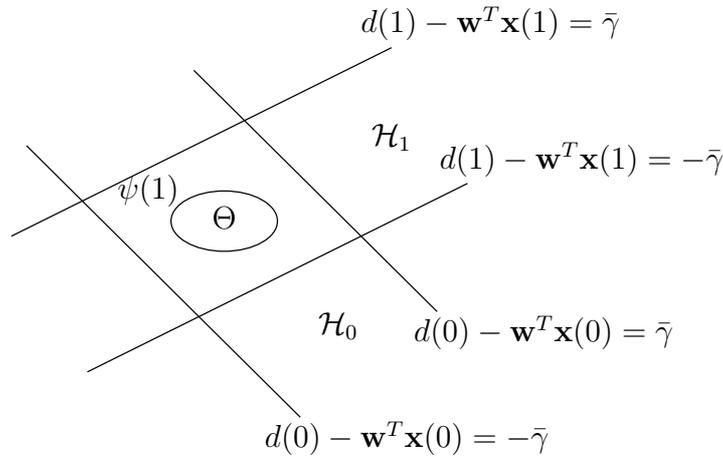


Figura 2.4: Interpretação geométrica do SMF no espaço dos coeficientes, considerando $k = 1$.

colhê-lo como uma função de σ_n^2 (variância do ruído), mais precisamente, $\bar{\gamma} = \sqrt{5\sigma_n^2}$.

2.3.1 SM-NLMS

O algoritmo SM-NLMS (*set-membership normalized least-mean-square*) foi proposto em [37], em 1998. Esse algoritmo mescla as características desejáveis do NLMS com a filosofia SMF.

O problema de otimização relacionado ao SM-NLMS é

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & \mathbf{w}(k+1) \in \mathcal{H}(k) \end{aligned} \quad (2.30)$$

de onde se conclui que:

- $\mathbf{w}(k) \in \mathcal{H}(k) \rightarrow \mathbf{w}(k+1) = \mathbf{w}(k)$, ou seja, não há atualização;
- $\mathbf{w}(k) \notin \mathcal{H}(k) \rightarrow \mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu(k)}{\|\mathbf{x}(k)\|^2} e(k) \mathbf{x}(k)$, onde $\mu(k) = 1 - \frac{\bar{\gamma}}{|e(k)|}$, isto é, $\mathbf{w}(k+1)$ é obtido através de uma projeção de $\mathbf{w}(k)$ em $\mathcal{H}(k)$.

Quando o SM-NLMS atualiza, a regra de formação de $\mathbf{w}(k+1)$ descrita acima minimiza a seguinte função objetivo [4]

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & |d(k) - \mathbf{w}^T(k+1)\mathbf{x}(k)| = \bar{\gamma} \end{aligned} \quad (2.31)$$

e a sua interpretação geométrica é descrita na Figura 2.5.

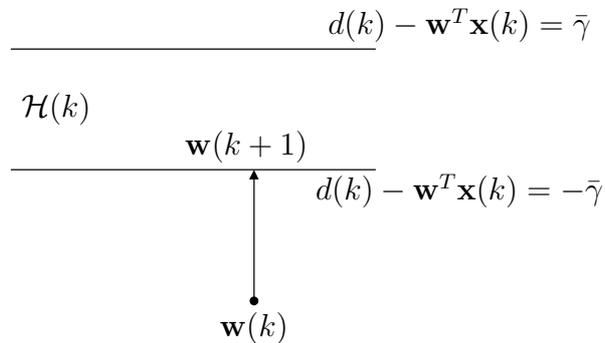


Figura 2.5: Interpretação geométrica da atualização do SM-NLMS no espaço dos coeficientes.

A seguir apresentamos o algoritmo SM-NLMS com fator de regularização δ [4]. Assim como no NLMS, o fator de regularização tem a finalidade de evitar instabilidade numérica que pode ocorrer quando $\|\mathbf{x}(k)\|^2 \rightarrow 0$, e é usualmente escolhido como uma constante pequena.

Algoritmo *Set-membership Normalized Least-Mean-Square*

Etapa 1. Inicialização:

$$\mathbf{x}(0) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$$

escolha $\bar{\gamma}$ próximo de $\sqrt{5\sigma_n^2}$, onde σ_n^2 é a variância do ruído $n(k)$

Etapa 2. Recursão:

Para $k \geq 0$ faça

$$\begin{aligned} e(k) &= d(k) - \mathbf{w}^T(k)\mathbf{x}(k) \\ \mu(k) &= \begin{cases} 1 - \frac{\bar{\gamma}}{|e(k)|} & \text{se } |e(k)| > \bar{\gamma} \\ 0 & \text{do contrário} \end{cases} \\ \mathbf{w}(k+1) &= \mathbf{w}(k) + \frac{\mu(k)}{\delta + \|\mathbf{x}(k)\|^2} e(k)\mathbf{x}(k) \end{aligned}$$

2.3.2 SM-AP

O algoritmo SM-AP (*set-membership affine projection*) surgiu em [13], em 2001. A idéia de Werner e Diniz era formular um algoritmo com reuso e seleção de dados, de forma que a velocidade de convergência do SM-NLMS fosse aumentada (relacionado ao reuso de dados, vide Subseção 2.2.3), porém realizando menos operações aritméticas e sem piorar o desajuste (abordagem SMF).

Portanto, analogamente ao AP, o SM-AP reutiliza L dados passados para acelerar sua convergência. Vamos re-escrever o *exact membership set* da seguinte forma [4]

$$\begin{aligned} \psi(k) &= \left(\bigcap_{i=0}^{k-L-1} \mathcal{H}(i) \right) \left(\bigcap_{j=k-L}^k \mathcal{H}(j) \right) \\ &= \psi_0^{k-L-1} \quad \bigcap \quad \psi_{k-L}^k \end{aligned} \quad (2.32)$$

onde

$$\psi_0^{k-L-1} = \bigcap_{i=0}^{k-L-1} \mathcal{H}(i)$$

representa a interseção dos $k-L$ primeiros conjuntos restritivos, enquanto

$$\psi_{k-L}^k = \bigcap_{j=k-L}^k \mathcal{H}(j)$$

representa a interseção dos $L + 1$ últimos conjuntos restritivos. Dessa forma, o objetivo do SM-AP é achar um $\mathbf{w}(k+1) \in \psi_{k-L}^k$. Dito isso, o problema de otimização relacionado com o SM-AP é

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & \mathbf{w}(k+1) \in \psi_{k-L}^k \end{aligned} \quad (2.33)$$

de onde se conclui que:

- $\mathbf{w}(k) \in \psi_{k-L}^k \rightarrow \mathbf{w}(k+1) = \mathbf{w}(k)$, ou seja, não há atualização;
- $\mathbf{w}(k) \notin \psi_{k-L}^k \rightarrow \mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k)]^{-1} [\mathbf{e}(k) - \boldsymbol{\gamma}(k)]$, onde $\boldsymbol{\gamma}(k)$ é o vetor de restrição e $\mathbf{w}(k+1)$ é obtido através de uma projeção de $\mathbf{w}(k)$ em ψ_{k-L}^k . O vetor de restrição $\boldsymbol{\gamma}(k)$ pode ser escolhido de infinitas maneiras, desde que respeite $|\gamma_i(k)| \leq \bar{\gamma}$, para $i = 0, 1, \dots, L$. Na prática, as principais escolhas são [4]: vetor de restrição com módulo constante baseado no erro (apresentado no início da Seção 4.4), vetor trivial (vide Seção 4.5) e o vetor de restrição simples (também conhecido como *simple-choice*), que não será abordado neste trabalho.

Quando o SM-AP atualiza, a regra de formação de $\mathbf{w}(k+1)$ descrita acima minimiza a seguinte função objetivo [4]

$$\begin{aligned} \min_{\mathbf{w}(k+1)} \quad & \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ \text{sujeito a} \quad & \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k+1) = \boldsymbol{\gamma}(k) \end{aligned} \quad (2.34)$$

e a sua interpretação geométrica, para um dado vetor $\boldsymbol{\gamma}(k)$ escolhido de forma arbitrária (i.e., suas componentes foram escolhidas ao acaso, porém respeitando $|\gamma_i(k)| \leq \bar{\gamma}$, para $i = 0, 1, \dots, L$), é descrita na Figura 2.6. Note que as variáveis aqui presentes são as mesmas definidas anteriormente para o AP, vide Subseção 2.2.3.

A seguir apresentamos o algoritmo SM-AP com fator de regularização δ [4].

Algoritmo *Set-membership Affine Projection*

Etapa 1. Inicialização:

$$\mathbf{x}(0) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$$

escolha $\bar{\gamma}$ próximo de $\sqrt{5\sigma_n^2}$

escolha o vetor $\boldsymbol{\gamma}(k)$ respeitando $|\gamma_i(k)| \leq \bar{\gamma}$, para $i = 0, 1, \dots, L$

Etapa 2. Recursão:

Para $k \geq 0$ faça

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k)$$

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1} [\mathbf{e}(k) - \boldsymbol{\gamma}(k)] & \text{se } |e_0(k)| > \bar{\gamma} \\ \mathbf{w}(k) & \text{do contrário} \end{cases}$$

2.3.3 Considerações Finais

O SM-AP engloba, como casos particulares, os algoritmos SM-BNLMS, SM-NLMS, AP com $\mu = 1$ (e os seus casos particulares, por exemplo o BNLMS e o NLMS, ambos com $\mu = 1$), dentre outros. Note que quando o SM-AP atualiza, a sua lei de formação de $\mathbf{w}(k+1)$ é bem parecida com a do AP e, portanto, deve ser possível utilizar uma abordagem similar à do FAP [33] para implementar o SM-AP de forma rápida.

O SM-AP possui os seguintes graus de liberdade: $\boldsymbol{\gamma}(k)$, $\bar{\gamma}$, L e δ . Existem muitos trabalhos visando otimizar esses parâmetros segundo algum critério. Em [43, 44] estuda-se a utilização de $\bar{\gamma}$ variante no tempo. Em [45] otimiza-se L de forma a aumentar a velocidade de convergência, aumentando L , e, após convergir, diminuir o L tornando o desajuste um pouco menor. Quanto à otimização de δ , valem as mesmas observações feitas no final da seção anterior (parte de considerações finais).

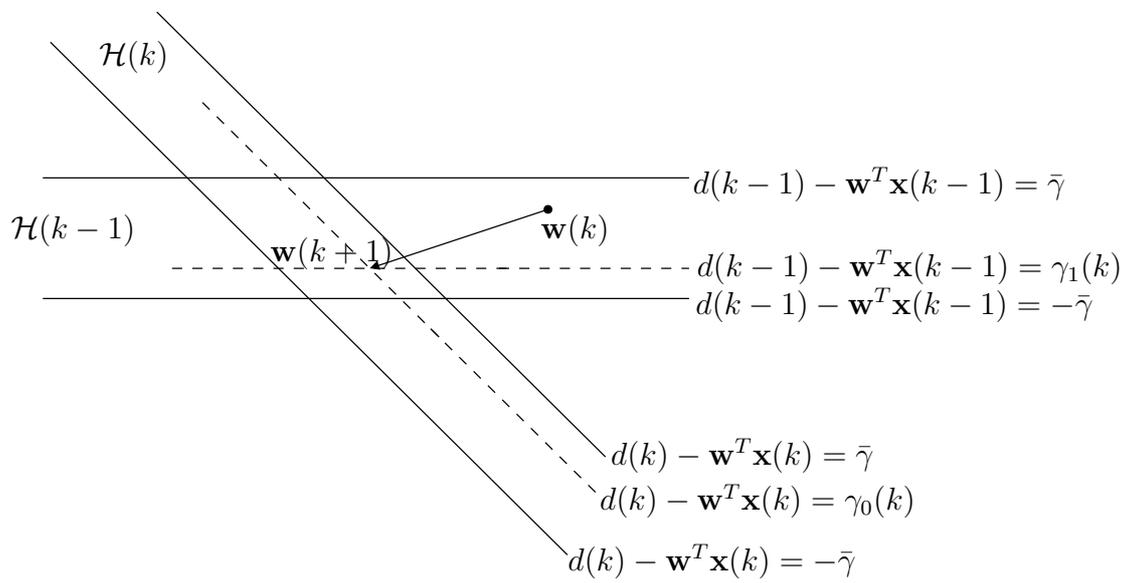


Figura 2.6: Interpretação geométrica da atualização do SM-AP no espaço dos coeficientes.

Capítulo 3

Principais Técnicas de Análise de EMSE

Este capítulo tem por finalidade apresentar os três principais métodos de análise de EMSE (*excess of mean-square error*) presentes na literatura. Dessa forma, as Seções 3.1, 3.2, e 3.3 correspondem aos métodos de análise por equação a diferenças (ED), por equação diferencial ordinária (EDO), e o método de conservação de energia, respectivamente. Após essa breve exposição, apresentamos, na Seção 3.4, os argumentos que nos levaram a utilizar o método de conservação de energia para analisar o algoritmo SM-AP.

3.1 Equação a Diferenças

O método de análise de EMSE por equação a diferenças (ED) se baseia na teoria da independência [46], vide Definição 1.

Definição 1 (Teoria da Independência). Assume-se que os vetores de entrada $\mathbf{x}(k)$ são independentes entre si. Enunciando de maneira mais formal, a teoria da independência diz que:

Dois vetores de entrada $\mathbf{x}(k_1)$ e $\mathbf{x}(k_2)$ são estatisticamente independentes para todo $k_1 \neq k_2$.

A teoria da independência, vide Definição 1, pode ser aplicada a quaisquer vetores de entrada independentemente de suas estruturas. Embora essa teoria não seja

válida para algoritmos que utilizem vetores de entrada com estrutura de *tapped-delay-line*, que é o caso dos algoritmos que nos interessam nesta dissertação, ela simplifica consideravelmente a análise, como veremos a seguir.

É claro que para vetores de entrada com estrutura de *tapped-delay-line*

Agora, vamos começar a análise de EMSE por ED. Primeiramente, o sinal de saída do filtro adaptativo em uma dada iteração k após o transiente é

$$\begin{aligned} y(k) &= \mathbf{w}^T(k)\mathbf{x}(k) \\ &= \mathbf{w}^T(k)\mathbf{x}(k) - \mathbf{w}_o^T\mathbf{x}(k) + \mathbf{w}_o^T\mathbf{x}(k) \\ &= \Delta\mathbf{w}^T(k)\mathbf{x}(k) + \mathbf{w}_o^T\mathbf{x}(k) \end{aligned} \quad (3.1)$$

onde

$$\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o \quad (3.2)$$

é o vetor de erro dos coeficientes, ao qual chamaremos, por simplicidade, de vetor desvio. Definindo o erro existente quando $\mathbf{w}(k) = \mathbf{w}_o$ como sendo

$$e_o(k) = d(k) - \mathbf{w}_o^T\mathbf{x}(k) \quad (3.3)$$

o MSE após a convergência $\xi(k)$ pode ser escrito como

$$\begin{aligned} \xi(k) &= \text{E} \left[e^2(k) \right] \\ &= \text{E} \left[(d(k) - y(k))^2 \right] \\ &= \text{E} \left[(d(k) - \mathbf{w}_o^T\mathbf{x}(k) - \Delta\mathbf{w}^T(k)\mathbf{x}(k))^2 \right] \\ &= \text{E} \left[(e_o(k) - \Delta\mathbf{w}^T(k)\mathbf{x}(k))^2 \right] \\ &= \text{E} \left[e_o^2(k) \right] - 2\text{E} \left[e_o(k)\Delta\mathbf{w}^T(k)\mathbf{x}(k) \right] + \text{E} \left[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k) \right] \end{aligned} \quad (3.4)$$

e, como consequência da teoria da independência (vide Definição 1) temos que $\Delta\mathbf{w}(k)$ é independente de $\mathbf{x}(k)$, pois $\Delta\mathbf{w}(k)$ depende da variável aleatória $\mathbf{w}(k)$ que por sua vez depende dos vetores de entrada anteriores a $\mathbf{x}(k)$. Dessa forma, podemos reescrever a equação acima como

$$\xi(k) = \text{E} \left[e_o^2(k) \right] - 2\text{E} \left[\Delta\mathbf{w}^T(k) \right] \text{E} \left[e_o(k)\mathbf{x}(k) \right] + \text{E} \left[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k) \right] \quad (3.5)$$

e, pelo princípio da ortogonalidade [4, 6], sabemos que $\text{E} \left[e_o(k)\mathbf{x}(k) \right] = 0$. Além disso, o termo $\text{E} \left[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k) \right]$ é maior ou igual a 0, onde a igualdade

é obtida quando $\mathbf{w}(k) = \mathbf{w}_o$. Nesse caso, tem-se o mínimo MSE após convergência ξ_{\min} , ou seja,

$$\xi_{\min} = \xi(k)|_{\mathbf{w}(k)=\mathbf{w}_o} = \mathbb{E} \left[e_o^2(k) \right] \quad (3.6)$$

Dessa forma, a equação (3.5) pode ser escrita como

$$\xi(k) = \xi_{\min} + \mathbb{E} \left[\Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k) \right] \quad (3.7)$$

Como a equação acima envolve apenas escalares, aplicaremos o operador traço no último termo e utilizaremos a propriedade $\text{tr}\{\mathbf{AB}\} = \text{tr}\{\mathbf{BA}\}$ para escrevermos

$$\begin{aligned} \xi(k) &= \xi_{\min} + \text{tr} \left\{ \mathbb{E} \left[\Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \Delta \mathbf{w}(k) \right] \right\} \\ &= \xi_{\min} + \mathbb{E} \left[\text{tr} \left\{ \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{x}(k) \mathbf{x}^T(k) \right\} \right] \\ &= \xi_{\min} + \text{tr} \left\{ \mathbb{E} \left[\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \right] \mathbb{E} \left[\mathbf{x}(k) \mathbf{x}^T(k) \right] \right\} \\ &= \xi_{\min} + \text{tr} \left\{ \mathbb{E} \left[\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \mathbf{R} \right] \right\} \\ &= \xi_{\min} + \mathbb{E} \left[\text{tr} \left\{ \Delta \mathbf{w}^T(k) \mathbf{R} \Delta \mathbf{w}(k) \right\} \right] \\ &= \xi_{\min} + \mathbb{E} \left[\Delta \mathbf{w}^T(k) \mathbf{R} \Delta \mathbf{w}(k) \right] \end{aligned} \quad (3.8)$$

onde novamente utilizou-se a teoria da independência para separar os valores esperados que dependiam de $\Delta \mathbf{w}(k)$ dos que dependiam de $\mathbf{x}(k)$.

Por definição, o EMSE avalia o quão longe o MSE está do MSE mínimo, ou seja, $\text{EMSE} = \xi(k) - \xi_{\min}$ e, de acordo com a equação acima

$$\text{EMSE} = \mathbb{E} \left[\Delta \mathbf{w}^T(k) \mathbf{R} \Delta \mathbf{w}(k) \right] \quad (3.9)$$

Como a média de $\Delta \mathbf{w}(k)$ é zero (isto é, estimador não é polarizado) para k suficientemente grande, a equação (3.9) pode ser escrita em função da matriz de covariância de $\Delta \mathbf{w}(k)$, $\text{cov}[\Delta \mathbf{w}(k)] = \mathbb{E} \left[\Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \right]$ aplicando, novamente, o operador traço

$$\begin{aligned} \text{EMSE} &= \mathbb{E} \left[\text{tr} \left\{ \mathbf{R} \Delta \mathbf{w}(k) \Delta \mathbf{w}^T(k) \right\} \right] \\ &= \text{tr} \left\{ \mathbf{R} \text{cov}[\Delta \mathbf{w}(k)] \right\} \end{aligned} \quad (3.10)$$

Utilizando a decomposição espectral [4, 22] para a matriz \mathbf{R} , podemos escrevê-la como $\mathbf{R} = \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^T$, onde $\mathbf{\Lambda} \in \mathbb{R}^{N+1 \times N+1}$ é uma matriz diagonal formada pelos autovalores de \mathbf{R} , e $\mathbf{Q} \in \mathbb{R}^{N+1 \times N+1}$ é uma matriz ortogonal cujas colunas são os autovetores de \mathbf{R} . Dessa forma, podemos reescrever a equação acima como

$$\text{EMSE} = \text{tr} \left\{ \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^T \text{cov}[\Delta \mathbf{w}(k)] \right\} \quad (3.11)$$

e como $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}$, podemos multiplicar $\mathbf{Q}\mathbf{Q}^T$ pela direita e criar uma nova variável definida como $\Delta\mathbf{w}'(k) = \mathbf{Q}^T \Delta\mathbf{w}(k)$, e com isso escrever

$$\begin{aligned} \text{EMSE} &= \text{tr} \left\{ \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T \text{cov} [\Delta\mathbf{w}(k)] \mathbf{Q}\mathbf{Q}^T \right\} \\ &= \text{tr} \left\{ \mathbf{Q}\mathbf{\Lambda} \text{cov} [\Delta\mathbf{w}'(k)] \mathbf{Q}^T \right\} \\ &= \text{tr} \left\{ \mathbf{\Lambda} \text{cov} [\Delta\mathbf{w}'(k)] \right\} \end{aligned} \quad (3.12)$$

Através da equação (3.12) conclui-se que apenas os elementos da diagonal de $\text{cov} [\Delta\mathbf{w}'(k)]$ influenciam no EMSE. Dessa forma, definindo o vetor $\mathbf{v}'(k) = \text{diag}\{\text{cov} [\Delta\mathbf{w}'(k)]\}$ e o vetor $\boldsymbol{\lambda} = \text{diag}\{\mathbf{\Lambda}\}$, podemos escrever essa equação da seguinte forma

$$\text{EMSE} = \boldsymbol{\lambda}^T \mathbf{v}'(k) = \sum_{i=0}^N \lambda_i v'_i(k) \quad (3.13)$$

Essa é a essência do método de análise por ED. Agora, o que nos falta para calcularmos o EMSE é escolher algum algoritmo para analisar. Basicamente, utilizamos a equação de atualização do algoritmo adaptativo escolhido para calcular o $\text{cov} [\Delta\mathbf{w}'(k)]$ e substituir em (3.12), ou então calcular diretamente o vetor $\mathbf{v}'(k)$ e substituí-lo diretamente em (3.13). Note que esse processo requer os mesmos passos que foram realizados nesta seção, isto é, invocar a teoria da independência, utilizar a decomposição espectral para \mathbf{R} , e realizar a mudança de variável. Ao fim do processo teremos uma equação onde $\text{cov} [\Delta\mathbf{w}'(k+1)]$ é escrito em função de $\text{cov} [\Delta\mathbf{w}'(k)]$ e, assumindo que após convergência vale $\text{cov} [\Delta\mathbf{w}'(k+1)] = \text{cov} [\Delta\mathbf{w}'(k)]$, podemos resolvê-la.

3.2 Equação Diferencial Ordinária

O método de análise de filtros adaptativos por meio de equação diferencial ordinária (EDO) foi desenvolvido por [47, 48], e uma de suas principais vantagens é não utilizar a teoria da independência (vide Definição 1). Em síntese, o método de EDO aproxima uma ED por uma EDO e, a partir daí, realiza a análise no domínio contínuo utilizando a grande variedade de resultados disponíveis para sistemas contínuos.

A análise por EDO, diferentemente da análise por ED, já parte da equação de atualização do algoritmo. Por conveniência, utilizaremos o LMS como exemplo, conforme realizado em [6]. Conforme vimos na Subseção 2.2.1, a equação de atualização

do LMS em uma dada iteração $k + P - 1$ é

$$\mathbf{w}(k + P) = \mathbf{w}(k + P - 1) + 2\mu e(k + P - 1)\mathbf{x}(k + P - 1) \quad (3.14)$$

Resolvendo essa recursão, podemos escrever a equação anterior em função de $\mathbf{w}(k)$

$$\begin{aligned} \mathbf{w}(k + P) &= \mathbf{w}(k + P - 1) + 2\mu e(k + P - 1)\mathbf{x}(k + P - 1) \\ &= \mathbf{w}(k + P - 2) + 2\mu e(k + P - 2)\mathbf{x}(k + P - 2) + 2\mu e(k + P - 1)\mathbf{x}(k + P - 1) \\ &\vdots \\ &= \mathbf{w}(k) + 2\mu \sum_{i=k}^{k+P-1} e(i)\mathbf{x}(i) \end{aligned} \quad (3.15)$$

e, após uma breve manipulação, temos

$$\begin{aligned} \frac{\mathbf{w}(k + P) - \mathbf{w}(k)}{2\mu} &= \sum_{i=k}^{k+P-1} e(i)\mathbf{x}(i) \\ \frac{\mathbf{w}(k + P) - \mathbf{w}(k)}{2\mu P} &= \frac{1}{P} \sum_{i=k}^{k+P-1} e(i)\mathbf{x}(i) \end{aligned} \quad (3.16)$$

Note que no lado direito da igualdade temos uma média temporal. Assumindo que os processos estocásticos são estacionários e ergódicos [21] e que o μ é suficientemente pequeno de forma que os coeficientes não variem muito no intervalo de iterações $[k, k + P - 1]$, a seguinte aproximação pode ser feita

$$\frac{1}{P} \sum_{i=k}^{k+P-1} e(i)\mathbf{x}(i) \approx \mathbb{E}[e(k)\mathbf{x}(k)] = \mathbf{p} - \mathbf{R}\mathbf{w}(k) \quad (3.17)$$

e, com isso, podemos aproximar a equação (3.16) como

$$\frac{\mathbf{w}(k + P) - \mathbf{w}(k)}{2\mu P} = \mathbf{p} - \mathbf{R}\mathbf{w}(k) \quad (3.18)$$

que, para $\mu \rightarrow 0$ pode ser aproximada por

$$\frac{d\mathbf{w}(t)}{dt} = \mathbf{p} - \mathbf{R}\mathbf{w}(t) \quad (3.19)$$

3.3 Conservação de Energia

O método de conservação de energia foi proposto nos trabalhos [49, 50], por Sayed e Rupp, e foi posteriormente utilizado para analisar diversos algoritmos. Em [6] encontram-se os principais resultados envolvendo/utilizando o método de conservação de energia. Ao contrário dos dois métodos anteriormente apresentados,

este método não utiliza a teoria da independência e nem requer um passo infinitesimal ($\mu \rightarrow 0$). Além disso, assim como o primeiro método apresentado, a derivação da relação de conservação de energia é bastante genérica, por exemplo, ela vale para a grande maioria dos algoritmos do tipo gradiente estocástico.

Inicialmente, vamos considerar um subconjunto de algoritmos pertencentes à família gradiente estocástico cuja equação de atualização pode ser escrita da seguinte forma

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\mathbf{x}(k)f[e(k)] \quad (3.20)$$

Note que, dentre os algoritmos que foram apresentados no capítulo 2, dependendo da escolha da função $f[e(k)]$, temos um algoritmo diferente:

- fazendo $f[e(k)] = e(k)$ temos o LMS;
- fazendo $f[e(k)] = \frac{1}{2} \frac{1}{\|\mathbf{x}(k)\|^2} e(k)$ temos o NLMS;
- fazendo $f[e(k)] = \frac{1}{2} \frac{1}{\delta + \|\mathbf{x}(k)\|^2} e(k)$ temos o NLMS regularizado;
- através de outras escolhas, podemos chegar a outros algoritmos que não foram abordados nesta dissertação, por exemplo, LMF ou o sign-error [4, 6].

A seguir, subtraímos \mathbf{w}_o dos dois lados da equação (3.20) gerando

$$\Delta\mathbf{w}(k+1) = \Delta\mathbf{w}(k) + 2\mu\mathbf{x}(k)f[e(k)] \quad (3.21)$$

onde $\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$ é o vetor desvio, o mesmo que foi utilizado na Subseção 3.1. Multiplicando por $\mathbf{x}^T(k)$ pela esquerda, a equação acima torna-se

$$-\tilde{\varepsilon}(k) = -\tilde{e}(k) + 2\mu\|\mathbf{x}(k)\|^2 f[e(k)] \quad (3.22)$$

onde o erro-sem-ruído *a priori* $\tilde{e}(k)$ e o erro-sem-ruído *a posteriori* $\tilde{\varepsilon}(k)$ são definidos como

$$\tilde{e}(k) = -\mathbf{x}^T(k)\Delta\mathbf{w}(k) = e(k) - n(k) \quad (3.23)$$

$$\tilde{\varepsilon}(k) = -\mathbf{x}^T(k)\Delta\mathbf{w}(k+1) = \varepsilon(k) - n(k) \quad (3.24)$$

onde $e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$ e $\varepsilon(k) = d(k) - \mathbf{w}^T(k+1)\mathbf{x}(k)$ são os erros *a priori* e *a posteriori*, respectivamente.

Podemos isolar $f[e(k)]$ obtendo

$$f[e(k)] = \frac{(\|\mathbf{x}(k)\|^2)^\dagger}{2\mu} [\tilde{e}(k) - \tilde{\varepsilon}(k)] \quad (3.25)$$

onde $(\|\mathbf{x}(k)\|^2)^\dagger$ é a pseudo-inversa de um escalar, cuja definição é [6]

$$(\|\mathbf{x}(k)\|^2)^\dagger = \begin{cases} \frac{1}{\|\mathbf{x}(k)\|^2} & \text{se } \mathbf{x}(k) \neq 0 \\ 0 & \text{do contrário} \end{cases} \quad (3.26)$$

Substituindo (3.25) em (3.21) temos

$$\Delta \mathbf{w}(k+1) = \Delta \mathbf{w}(k) + \mathbf{x}(k) (\|\mathbf{x}(k)\|^2)^\dagger [\tilde{e}(k) - \tilde{\varepsilon}(k)] \quad (3.27)$$

E, re-arranjando os termos

$$\Delta \mathbf{w}(k+1) - \mathbf{x}(k) (\|\mathbf{x}(k)\|^2)^\dagger \tilde{e}(k) = \Delta \mathbf{w}(k) - \mathbf{x}(k) (\|\mathbf{x}(k)\|^2)^\dagger \tilde{\varepsilon}(k) \quad (3.28)$$

Finalmente, tomando a norma de ambos os lados da equação anterior, a seguinte relação se revela

$$\|\Delta \mathbf{w}(k+1)\|^2 + (\|\mathbf{x}(k)\|^2)^\dagger |\tilde{e}(k)|^2 = \|\Delta \mathbf{w}(k)\|^2 + (\|\mathbf{x}(k)\|^2)^\dagger |\tilde{\varepsilon}(k)|^2 \quad (3.29)$$

que é a relação de conservação de energia. Note que todos os termos presentes nessa equação dizem respeito à energia associada a algum tipo de erro. Os $\Delta \mathbf{w}(k+1)$ e $\Delta \mathbf{w}(k)$ representam o erro presente nos coeficientes, isto é, o quão distante os coeficientes estão de \mathbf{w}_o . Os $\tilde{e}(k)$ e $\tilde{\varepsilon}(k)$ representam o erro-sem-ruído *a priori* e *a posteriori*, respectivamente, ou seja, só contabilizam os erros inerentes ao algoritmo.

3.4 Escolha do Método

Descartamos a utilização do método ED, pois desejamos evitar recorrer à teoria da independência, uma vez que ela não reflete a realidade, já que estamos utilizando uma estrutura de *tapped-delay line*¹. Além disso, para utilizarmos o método ED precisamos avaliar $\text{cov}[\Delta \mathbf{w}'(k+1)]$, que por sua vez depende de $\text{cov}[\Delta \mathbf{w}(k+1)] = \text{E} [\Delta \mathbf{w}(k+1) \Delta \mathbf{w}^T(k+1)]$, que pode ser computado a partir da equação de atualização do algoritmo adaptativo escolhido. Porém, avaliar

¹Na verdade, se considerarmos que $\mathbf{x}(k)$ e $\mathbf{w}(k)$ são dois processos conjuntamente estacionários, e que $\mathbf{w}(k)$ varia muito lentamente com $\mathbf{x}(k)$, então podemos considerar que esses dois processos são aproximadamente independentes [54].

$\text{cov}[\Delta\mathbf{w}(k+1)]$ é complicado para algoritmos que fazem reuso de dados, uma vez que a teoria da independência não pode ser aplicada para $\mathbf{X}(k)$ e $\Delta\mathbf{w}(k)$ com $L \neq 0$, considerando-se que $\mathbf{X}(k)$ guarda L vetores de entrada anteriores.

O método por EDO também foi descartado, pois, para o SM-AP, não há um parâmetro μ que possamos forçar que ele tenda a zero. Além disso, o método de EDO costuma facilitar o estudo da estabilidade do algoritmo, e não a análise de MSE (que é o que nos importa).

Portanto, nos resta o método de conservação de energia que já foi utilizado com sucesso para analisar o AP [51].

Capítulo 4

Análise de EMSE do SM-AP

No presente capítulo, derivaremos expressões para duas medidas de desempenho bem conhecidas, denominadas de excesso de MSE (EMSE) e desajuste, com a finalidade de descrever o comportamento de MSE em regime permanente (isto é, após a convergência) do algoritmo SM-AP em um ambiente estacionário.

Inicialmente, na Seção 4.1, apresentaremos um resumo das variáveis que serão constantemente utilizadas neste capítulo. O objetivo dessa seção é facilitar o leitor concentrando as definições importantes em um único lugar, e além disso, possibilitar que os capítulos anteriores possam ser pulados por pessoas já familiarizadas com o assunto. A Seção 4.2 destina-se a apresentar o ambiente estacionário, definir matematicamente o EMSE e desajuste, e apresentar o nosso modelo de análise. Em seguida, na Seção 4.3, derivamos expressões de conservação de energia para o nosso modelo. Por fim, nas Seções 4.4 e 4.5, escolhemos dois tipos de vetores de restrição $\gamma(k)$ que são comumente utilizados e derivamos as expressões para o EMSE e desajuste.

4.1 Resumo das Principais Variáveis

Como já foi dito, o SM-AP reutiliza dados antigos. Assumindo que dispomos dos últimos $L + 1$ pares de dados de vetores de entrada e de sinais desejados, podemos definir as principais variáveis presentes no algoritmo SM-AP em uma determinada

iteração k :

$$\begin{aligned}
\mathbf{X}(k) &= [\mathbf{x}(k) \quad \mathbf{x}(k-1) \quad \dots \quad \mathbf{x}(k-L)] && \text{(matriz de entrada)} \\
\mathbf{x}(k) &= [x(k) \quad x(k-1) \quad \dots \quad x(k-N)]^T && \text{(vetor de entrada)} \\
\mathbf{w}(k) &= [w_0(k) \quad w_1(k) \quad \dots \quad w_N(k)]^T && \text{(vetor de coeficientes)} \\
\mathbf{d}(k) &= [d(k) \quad d(k-1) \quad \dots \quad d(k-L)]^T && \text{(vetor desejado)} \\
\boldsymbol{\gamma}(k) &= [\gamma_0(k) \quad \gamma_1(k) \quad \dots \quad \gamma_L(k)]^T && \text{(vetor de restrição)} \\
\mathbf{e}(k) &= [e_0(k) \quad e_1(k) \quad \dots \quad e_L(k)]^T && \text{(vetor de erro } a \text{ priori)} \\
\boldsymbol{\varepsilon}(k) &= [\varepsilon_0(k) \quad \varepsilon_1(k) \quad \dots \quad \varepsilon_L(k)]^T && \text{(vetor de erro } a \text{ posteriori)}
\end{aligned} \tag{4.1}$$

onde a matriz de entrada $\mathbf{X}(k) \in \mathbb{R}^{(N+1) \times (L+1)}$ contém o vetor de entrada atual $\mathbf{x}(k) \in \mathbb{R}^{N+1}$ e os L vetores de entrada anteriores $\mathbf{x}(k-i)$ para $i = 1, \dots, L$; os coeficientes do filtro adaptativo são dados pelo vetor de coeficientes $\mathbf{w}(k) \in \mathbb{R}^{N+1}$; o vetor desejado $\mathbf{d}(k) \in \mathbb{R}^{L+1}$ contém o sinal desejado atual $d(k)$ e os L anteriores $d(k-i)$ para $i = 1, \dots, L$; o vetor de restrição é dado por $\boldsymbol{\gamma}(k) \in \mathbb{R}^{L+1}$; sendo que $\mathbf{e}(k), \boldsymbol{\varepsilon}(k) \in \mathbb{R}^{L+1}$ definidos como

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k) \tag{4.2}$$

$$\boldsymbol{\varepsilon}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k+1) \tag{4.3}$$

são os vetores de erro *a priori* e *a posteriori*, respectivamente.¹

O parâmetro $\bar{\gamma} \in \mathbb{R}_+$ estabelece um limite superior para o vetor de restrição, isto é, $|\gamma_i(k)| \leq \bar{\gamma}$ para $i = 0, \dots, L$ e é usualmente escolhido com base na variância do ruído σ_n^2 . O valor mais comumente utilizado é $\bar{\gamma} = \sqrt{5\sigma_n^2}$, uma vez que esse valor faz com que o algoritmo SM-NLMS atualize durante poucas iterações e tenha desajuste pequeno [44]. O vetor de ruído $\mathbf{n}(k) \in \mathbb{R}^{L+1}$ é definido como

$$\mathbf{n}(k) = [n(k) \quad n(k-1) \quad \dots \quad n(k-L)]^T \quad \text{(vetor de ruído)} \tag{4.4}$$

Note, a partir das definições acima, que os vetores de entrada, desejado e o de ruído tem uma estrutura de *tapped-delay line* e, portanto, existe uma relação de deslocamento entre os seus elementos em iterações próximas. Essa observação será utilizada posteriormente.

¹Não confundir $e_0(k)$, primeira entrada do vetor $\mathbf{e}(k)$ (vide equação (4.1)), com $e_o(k)$, erro existente no sistema quando os coeficientes do filtro adaptativo coincidem com a solução de Wiener, isto é, quando $\mathbf{w}(k) = \mathbf{w}_o$ (vide equação (3.3)).

4.2 Preliminares

Vamos começar descrevendo o modelo do ambiente que será considerado, conhecido como modelo estacionário (pag. 285 de [6]).

Definição 2 (Modelo Estacionário). Dizemos que os dados $\{d(k), \mathbf{x}(k), n(k)\}$ satisfazem o modelo estacionário se eles obedecem às seguintes condições:

- (a) $\exists \mathbf{w}_o \in \mathbb{R}^{N+1} : d(k) = \mathbf{w}_o^T \mathbf{x}(k) + n(k), \forall k;$
- (b) As variáveis aleatórias $d(k), n(k), \mathbf{x}(k)$ tem média 0, $\forall k;$
- (c) $\forall k, n(k)$ é i.i.d. com variância $\sigma_n^2 = E[n^2(k)];$
- (d) $\forall k_1, \forall k_2, n(k_1)$ é independente de $\mathbf{x}(k_2);$
- (e) A matriz de autocorrelação do sinal de entrada é $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)], \forall k;$
- (f) A condição inicial $\mathbf{w}(0)$ é independente de $\{d(k), \mathbf{x}(k), n(k)\}, \forall k.$

O modelo descrito acima recebe o nome de estacionário porque \mathbf{w}_o, σ_n^2 e \mathbf{R} não dependem de k . Como consequência da Definição 2, podemos escrever o erro *a priori* como

$$\begin{aligned} e(k) &= d(k) - y(k) \\ &= \mathbf{w}_o^T \mathbf{x}(k) + n(k) - \mathbf{w}^T(k)\mathbf{x}(k) \\ &= n(k) + \tilde{e}(k) \end{aligned} \tag{4.5}$$

onde $\tilde{e}(k) = (\mathbf{w}_o - \mathbf{w}(k))^T \mathbf{x}(k)$ é o sinal de erro-sem-ruído *a priori*. Elevando ao quadrado a equação acima temos

$$e^2(k) = n^2(k) + \tilde{e}^2(k) + 2n(k)\tilde{e}(k) \tag{4.6}$$

E em seguida, tomando o valor esperado chegamos em

$$\begin{aligned} E[e^2(k)] &= E[n^2(k)] + E[\tilde{e}^2(k)] \\ &= \sigma_n^2 + E[\tilde{e}^2(k)] \end{aligned} \tag{4.7}$$

onde utilizamos na primeira linha a Proposição 1 (enunciada logo abaixo) e na segunda linha invocamos a Definição 2.

Proposição 1 (Resultado de Independência). *Como consequência do modelo estacionário dado na Definição 2, o ruído $n(k)$ é independente de $\tilde{e}(k)$ (para maiores detalhes, ver Subseção 6.2.2 de [6]).*

Finalmente, assumindo que o algoritmo converge, podemos estudar o seu desempenho de MSE em regime estacionário através da análise de)

$$\lim_{k \rightarrow \infty} E[e^2(k)] = \sigma_n^2 + \text{EMSE} \quad (4.8)$$

onde

$$\text{EMSE} = \lim_{k \rightarrow \infty} E[\tilde{e}^2(k)] \quad (4.9)$$

e o desajuste (M, do inglês *misadjustment*) é dado por

$$M = \frac{\text{EMSE}}{\sigma_n^2} \quad (4.10)$$

Note que, até agora, as equações apresentadas neste capítulo valem para qualquer algoritmo adaptativo em ambiente estacionário, mesmo para aqueles que reutilizam dados. Entretanto, nós adaptaremos a notação apresentada aqui para a notação apresentada na Seção 4.1, pois ela elimina ambiguidades que podem ocorrer na notação dos algoritmos que reutilizam dados, através da inclusão do subscrito 0 com a finalidade de denotar que $\tilde{e}_0(k) = \tilde{e}(k)$ é o primeiro elemento do vetor erro-sem-ruído *a priori* $\tilde{\mathbf{e}}(k)$ que será definido posteriormente, veja equação (4.25). Dessa forma, a definição do EMSE torna-se

$$\text{EMSE} = \lim_{k \rightarrow \infty} E[\tilde{e}_0^2(k)] \quad (4.11)$$

Agora que introduzimos as quantidades que desejamos calcular, podemos começar a análise do algoritmo SM-AP utilizando uma abordagem baseada em argumentos de conservação de energia [51] apresentada na Seção 3.3.

Na verdade, consideraremos um algoritmo que é uma pequena generalização do SM-AP, pois adota os coeficientes de regularização (δ) e de relaxamento (μ)². A esse SM-AP com constante de relaxamento chamaremos de GenSM-AP (isto é, SM-AP generalizado) apenas para diferenciá-lo do SM-AP original, introduzido em [13].

A equação de atualização do GenSM-AP é dada por

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mu \mathbf{X}(k) [\mathbf{X}^T(k) \mathbf{X}(k) + \delta \mathbf{I}]^{-1} [\mathbf{e}(k) - \boldsymbol{\gamma}(k)] & \text{se } |e_0(k)| > \bar{\gamma} \\ \mathbf{w}(k) & \text{do contrário} \end{cases} \quad (4.12)$$

²A única razão de termos considerado o fator μ , também chamado de fator de convergência, é para haver correspondência direta entre as expressões de EMSE do GenSM-AP e do AP. Em outras palavras, o GenSM-AP com vetor de restrição $\boldsymbol{\gamma}(k) = \mathbf{0}$ recai no algoritmo AP e, como mostraremos na Seção 4.5, as expressões de EMSE desses dois algoritmos são idênticas.

A não-linearidade apresentada na equação (4.12) devido à etapa de *innovation check* torna a análise muito difícil de ser realizada. Portanto, nós utilizaremos um modelo simplificado, introduzido na Seção IV de [14], para superar esse problema. Assim, representaremos a equação (4.12) da seguinte forma

$$\mathbf{w}(k+1) = \mathbf{w}(k) + P_{\text{up}}(k)\mu\mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1} [\mathbf{e}(k) - \boldsymbol{\gamma}(k)], \quad (4.13)$$

onde $P_{\text{up}}(k) \in [0, 1]$ é uma função que representa a probabilidade de atualizar os coeficientes do filtro em uma dada iteração k , isto é, $P_{\text{up}}(k) = P[|e_0(k)| > \bar{\gamma}]$.

4.2.1 O Modelo de Análise

Pode-se argumentar que o modelo de análise dado pela equação (4.13) não reflete a prática, dada pela equação (4.12), uma vez que a não-linearidade no erro é trocada por um fator escalar. Em outras palavras, o processo de *innovation check* é trocado por um parâmetro $P_{\text{up}}(k)$ que funciona da mesma forma que um coeficiente de relaxamento. Em resumo, o nosso modelo de análise, à primeira vista, parece não ser válido.

Isso seria verdade se (4.12) e (4.13) fossem equações a diferenças determinísticas. Entretanto, todas as variáveis presentes nessas equações são variáveis aleatórias, com exceção a μ e $P_{\text{up}}(k)$ que são determinísticas. Portanto, temos que encarar essas equações como equações estocásticas a diferenças. Além disso, as métricas que desejamos calcular (EMSE e desajuste) representam um comportamento médio, isto é, teremos que tomar o valor esperado das variáveis aleatórias em algum momento durante a nossa análise. Por fim, olhando para o valor esperado dessas duas equações veremos que elas representam a mesma relação. Vamos mostrar isso com mais detalhes.

Seja $\mathbf{p}(k) \in \mathbb{R}^{N+1}$ uma variável aleatória representando a perturbação aplicada a $\mathbf{w}(k)$ quando $|e_0(k)| > \bar{\gamma}$, i.e,

$$\mathbf{p}(k) = \mu\mathbf{X}(k) [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1} [\mathbf{e}(k) - \boldsymbol{\gamma}(k)]$$

Podemos escrever a equação (4.12) como

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mathbf{p}(k) & \text{se } |e_0(k)| > \bar{\gamma} \\ \mathbf{w}(k) & \text{do contrário} \end{cases} \quad (4.14)$$

Agora, seja $\mathbf{p}'(k)$ um vetor aleatório definido por

$$\mathbf{p}'(k) = \begin{cases} \mathbf{p}(k) & \text{com probabilidade } P_{\text{up}}(k) \\ \mathbf{0} & \text{com probabilidade } 1 - P_{\text{up}}(k) \end{cases}$$

onde $P_{\text{up}}(k) = P[|e_0(k)| > \bar{\gamma}]$. Com essa definição, a equação (4.14) pode ser reescrita como

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{p}'(k) \quad (4.15)$$

Em seguida, tomando o valor esperado de (4.15) obtemos

$$E[\mathbf{w}(k+1)] = E[\mathbf{w}(k)] + E[\mathbf{p}'(k)] \quad (4.16)$$

Calculando o valor esperado de $\mathbf{p}'(k)$, temos

$$\begin{aligned} E[\mathbf{p}'(k)] &= E[\mathbf{p}'(k) \mid |e_0(k)| \leq \bar{\gamma}] (1 - P_{\text{up}}(k)) + E[\mathbf{p}'(k) \mid |e_0(k)| > \bar{\gamma}] P_{\text{up}}(k) \\ &= \mathbf{0}(1 - P_{\text{up}}(k)) + E[\mathbf{p}(k) \mid |e_0(k)| > \bar{\gamma}] P_{\text{up}}(k) \\ &= E[\mathbf{p}(k) \mid |e_0(k)| > \bar{\gamma}] P_{\text{up}}(k) \end{aligned}$$

e, assumindo que $\mathbf{p}(k)$ e $|e_0(k)| > \bar{\gamma}$ são fracamente dependentes³, podemos escrever

$$E[\mathbf{p}'(k)] = E[\mathbf{p}(k)] P_{\text{up}}(k) \quad (4.17)$$

e agora é direto verificar que o valor esperado da equação (4.12), dado pelas equações (4.16) e (4.17), leva ao mesmo resultado que o valor esperado de (4.13).

Em resumo, (4.13) reflete (4.12), na média. Como estamos interessados no comportamento médio do algoritmo, precisaremos tomar valores esperados do nosso modelo e, desse momento em diante, o modelo de análise refletirá perfeitamente a prática.

4.3 Expressões de Conservação de Energia

O modelo de análise, dado pela equação (4.13), pode ser escrito como

$$\mathbf{w}(k+1) = \mathbf{w}(k) + P_{\text{up}}(k) \mu \mathbf{X}(k) \hat{\mathbf{S}}(k) [\mathbf{e}(k) - \boldsymbol{\gamma}(k)] \quad (4.18)$$

³Note que essa assunção será tão melhor quanto maior for $P_{\text{up}}(k)$, pois dessa forma a média de $\mathbf{p}(k)$ é dominada pela influência do termo $E[\mathbf{p}(k) \mid |e_0(k)| > \bar{\gamma}]$. Consequentemente, o modelo adotado se torna melhor conforme L cresce (vide gráficos de $P_{\text{up}}(k)$ apresentados no Capítulo 5) e $\bar{\gamma}$ decresce.

onde, com a finalidade de manter a notação compacta, definimos

$$\hat{\mathbf{R}}(k) = \mathbf{X}^T(k)\mathbf{X}(k) \quad (4.19)$$

$$\hat{\mathbf{S}}(k) = [\hat{\mathbf{R}}(k) + \delta\mathbf{I}]^{-1} = [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1} \quad (4.20)$$

Conforme já foi dito anteriormente (veja Definição 2), o sinal desejado é dado por

$$d(k) = \mathbf{w}_o^T \mathbf{x}(k) + n(k) \quad (4.21)$$

Subtraindo \mathbf{w}_o , os coeficientes ótimos do filtro, dos dois lados da equação (4.18) obtemos

$$\Delta\mathbf{w}(k+1) = \Delta\mathbf{w}(k) + P_{\text{up}}(k)\mu\mathbf{X}(k)\hat{\mathbf{S}}(k)[\mathbf{e}(k) - \gamma(k)] \quad (4.22)$$

onde $\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$ é o vetor desvio.

Multiplicando a equação (4.22) por $\mathbf{X}^T(k)$ pela esquerda, e usando (4.19) temos

$$-\tilde{\boldsymbol{\varepsilon}}(k) = -\tilde{\mathbf{e}}(k) + P_{\text{up}}(k)\mu\hat{\mathbf{R}}(k)\hat{\mathbf{S}}(k)[\mathbf{e}(k) - \gamma(k)] \quad (4.23)$$

onde

$$\tilde{\boldsymbol{\varepsilon}}(k) = -\mathbf{X}^T(k)\Delta\mathbf{w}(k+1) = \boldsymbol{\varepsilon}(k) - \mathbf{n}(k) \quad (4.24)$$

$$\tilde{\mathbf{e}}(k) = -\mathbf{X}^T(k)\Delta\mathbf{w}(k) = \mathbf{e}(k) - \mathbf{n}(k) \quad (4.25)$$

i.e., $\tilde{\boldsymbol{\varepsilon}}(k) = [\tilde{\varepsilon}_0(k) \ \tilde{\varepsilon}_1(k) \ \dots \ \tilde{\varepsilon}_L(k)]^T$, $\tilde{\mathbf{e}}(k) = [\tilde{e}_0(k) \ \tilde{e}_1(k) \ \dots \ \tilde{e}_L(k)]^T \in \mathbb{R}^{L+1}$, são os vetores erro-sem-ruído *a posteriori* e *a priori*, respectivamente. Lembre-se que precisamos do primeiro elemento do vetor $\tilde{\mathbf{e}}(k)$ para calcularmos o EMSE, vide equação (4.11).

Assumindo que $\hat{\mathbf{R}}(k)$ é inversível (isto é, que $\mathbf{X}(k)$ tem posto completo por colunas) podemos resolver a equação (4.23) da seguinte forma

$$\hat{\mathbf{R}}^{-1}(k)[\tilde{\mathbf{e}}(k) - \tilde{\boldsymbol{\varepsilon}}(k)] = P_{\text{up}}(k)\mu\hat{\mathbf{S}}(k)[\mathbf{e}(k) - \gamma(k)] \quad (4.26)$$

Usando a equação (4.26) na equação (4.22)

$$\Delta\mathbf{w}(k+1) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) = \Delta\mathbf{w}(k) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\boldsymbol{\varepsilon}}(k) \quad (4.27)$$

Teorema 1 (Relação de Conservação de Energia para o GenSM-AP). *Avaliando as energias dos dois lados da equação (4.27), podemos mostrar a seguinte relação*

$$\|\Delta\mathbf{w}(k+1)\|^2 + [\tilde{\boldsymbol{\varepsilon}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k)] = \|\Delta\mathbf{w}(k)\|^2 + [\tilde{\boldsymbol{\varepsilon}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\boldsymbol{\varepsilon}}(k)] \quad (4.28)$$

que envolve as energias dos vetores desvio, e dos vetores de erro-sem-ruído *a priori* e *a posteriori*.

A prova do Teorema 1 é realizada no Apêndice A. Note que esse teorema vale para outros algoritmos, por exemplo, o AP.

Aplicando o operador valor esperado à equação (4.28) obtemos

$$\mathbb{E} [\|\Delta \mathbf{w}(k+1)\|^2] + \mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k)] = \mathbb{E} [\|\Delta \mathbf{w}(k)\|^2] + \mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k)]$$

que é conhecida na literatura como a relação de variância das energias [51].

Assumindo que em uma dada iteração k o algoritmo já tenha convergido, então $\mathbb{E} [\|\Delta \mathbf{w}(k+1)\|^2] = \mathbb{E} [\|\Delta \mathbf{w}(k)\|^2]$, e portanto

$$\mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k)] = \mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k)] \quad (4.29)$$

vale em regime permanente, ou seja, as energias dos erro-sem-ruído *a priori* e *a posteriori* se conservam, em regime permanente.

Agora, utilizamos a equação (4.23) para eliminar a dependência do erro-sem-ruído *a posteriori*, $\tilde{\mathbf{e}}(k)$, da equação acima e usamos o fato de que tanto $\hat{\mathbf{R}}(k)$ quanto $\hat{\mathbf{S}}(k)$ são matrizes simétricas para obter, depois de uma manipulação matemática simples, a relação

$$P_{\text{up}}(k) \mu \mathbb{E} [(\mathbf{e}(k) - \boldsymbol{\gamma}(k))^T \hat{\mathbf{S}}(k) \hat{\mathbf{R}}(k) \hat{\mathbf{S}}(k) (\mathbf{e}(k) - \boldsymbol{\gamma}(k))] = 2 \mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{S}}(k) (\mathbf{e}(k) - \boldsymbol{\gamma}(k))] \quad (4.30)$$

onde usamos a relação $\tilde{\mathbf{e}}^T(k) \hat{\mathbf{S}}(k) (\mathbf{e}(k) - \boldsymbol{\gamma}(k)) = (\mathbf{e}(k) - \boldsymbol{\gamma}(k))^T \hat{\mathbf{S}}(k) \tilde{\mathbf{e}}(k)$, uma vez que os dois termos envolvidos são escalares e $\hat{\mathbf{S}}(k)$ é simétrica.

Pela equação (4.25) temos $\mathbf{e}(k) = \tilde{\mathbf{e}}(k) + \mathbf{n}(k)$. Substituindo essa relação em (4.30) temos

$$\begin{aligned} P_{\text{up}}(k) \mu \mathbb{E} [(\tilde{\mathbf{e}}(k) + \mathbf{n}(k) - \boldsymbol{\gamma}(k))^T \hat{\mathbf{S}}(k) \hat{\mathbf{R}}(k) \hat{\mathbf{S}}(k) (\tilde{\mathbf{e}}(k) + \mathbf{n}(k) - \boldsymbol{\gamma}(k))] \\ = 2 \mathbb{E} [\tilde{\mathbf{e}}^T(k) \hat{\mathbf{S}}(k) (\tilde{\mathbf{e}}(k) + \mathbf{n}(k) - \boldsymbol{\gamma}(k))] \end{aligned} \quad (4.31)$$

Definindo

$$\underline{\tilde{\mathbf{e}}}(k) = \tilde{\mathbf{e}}(k) - \boldsymbol{\gamma}(k) \quad (4.32)$$

a equação (4.31) pode ser reescrita como

$$P_{\text{up}}(k) \mu \mathbb{E} [(\underline{\tilde{\mathbf{e}}}(k) + \mathbf{n}(k))^T \hat{\mathbf{S}}(k) \hat{\mathbf{R}}(k) \hat{\mathbf{S}}(k) (\underline{\tilde{\mathbf{e}}}(k) + \mathbf{n}(k))] = 2 \mathbb{E} [(\underline{\tilde{\mathbf{e}}}(k) + \boldsymbol{\gamma}(k))^T \hat{\mathbf{S}}(k) (\underline{\tilde{\mathbf{e}}}(k) + \mathbf{n}(k))]$$

Expandindo a equação acima obtemos

$$\begin{aligned} P_{\text{up}}(k)\mu\text{E} \left[\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\hat{\mathbf{R}}(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \mathbf{n}^T(k)\hat{\mathbf{S}}(k)\hat{\mathbf{R}}(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) + 2\mathbf{n}^T(k)\hat{\mathbf{S}}(k)\hat{\mathbf{R}}(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) \right] \\ = 2\text{E} \left[\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) + \boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) \right] \end{aligned}$$

Uma vez que o fator de regularização é uma constante pequena ($\delta \ll 1$) usada apenas para evitar instabilidade numérica devido à inversão de matriz, podemos assumir $\hat{\mathbf{S}}(k) = \hat{\mathbf{R}}^{-1}(k)$ para simplificar a expressão acima

$$\begin{aligned} P_{\text{up}}(k)\mu\text{E} \left[\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \mathbf{n}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) + 2\mathbf{n}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) \right] \\ = 2\text{E} \left[\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) + \boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) + \boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) \right] \end{aligned}$$

Re-arranjando os termos e utilizando a relação $\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) = \mathbf{n}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k)$, uma vez que eles são escalares e $\hat{\mathbf{S}}(k)$ é simétrica, nós temos

$$\begin{aligned} (2 - P_{\text{up}}(k)\mu)\text{E} \left[\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) \right] + 2(1 - P_{\text{up}}(k)\mu)\text{E} \left[\mathbf{n}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) \right] + 2\text{E} \left[\boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\underline{\tilde{\mathbf{e}}}(k) \right] \\ = P_{\text{up}}(k)\mu\text{E} \left[\mathbf{n}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) \right] - 2\text{E} \left[\boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k)\mathbf{n}(k) \right] \end{aligned}$$

Aplicando o operador traço na equação acima e utilizando a propriedade $\text{tr}\{\mathbf{AB}\} = \text{tr}\{\mathbf{BA}\}$, podemos escrever

$$\begin{aligned} (2 - P_{\text{up}}(k)\mu)\text{E} \left[\text{tr} \left\{ \underline{\tilde{\mathbf{e}}}(k)\underline{\tilde{\mathbf{e}}}^T(k)\hat{\mathbf{S}}(k) \right\} \right] + 2(1 - P_{\text{up}}(k)\mu)\text{E} \left[\text{tr} \left\{ \underline{\tilde{\mathbf{e}}}(k)\mathbf{n}^T(k)\hat{\mathbf{S}}(k) \right\} \right] \\ + 2\text{E} \left[\text{tr} \left\{ \underline{\tilde{\mathbf{e}}}(k)\boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k) \right\} \right] \\ = P_{\text{up}}(k)\mu\text{E} \left[\text{tr} \left\{ \mathbf{n}(k)\mathbf{n}^T(k)\hat{\mathbf{S}}(k) \right\} \right] - 2\text{E} \left[\text{tr} \left\{ \mathbf{n}(k)\boldsymbol{\gamma}^T(k)\hat{\mathbf{S}}(k) \right\} \right] \end{aligned}$$

Considerando que $\hat{\mathbf{S}}(k)$ é estatisticamente independente do ruído e do erro-sem-ruído *a priori* após a convergência, podemos escrever

$$\begin{aligned} (2 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \text{E} \left[\underline{\tilde{\mathbf{e}}}(k)\underline{\tilde{\mathbf{e}}}^T(k) \right] \text{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + 2(1 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \text{E} \left[\underline{\tilde{\mathbf{e}}}(k)\mathbf{n}^T(k) \right] \text{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\ + 2\text{tr} \left\{ \text{E} \left[\underline{\tilde{\mathbf{e}}}(k)\boldsymbol{\gamma}^T(k) \right] \text{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\ = P_{\text{up}}(k)\mu\text{tr} \left\{ \text{E} \left[\mathbf{n}(k)\mathbf{n}^T(k) \right] \text{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2\text{tr} \left\{ \text{E} \left[\mathbf{n}(k)\boldsymbol{\gamma}^T(k) \right] \text{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \end{aligned}$$

Por fim, eliminamos a dependência de $\underline{\tilde{\mathbf{e}}}(k)$ através da substituição de (4.32) na

equação acima, e obtemos

$$\begin{aligned}
& (2 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + 2(1 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad - P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& - (2 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\boldsymbol{\gamma}(k)\tilde{\mathbf{e}}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2(1 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\boldsymbol{\gamma}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad = P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\mathbf{n}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2\text{tr} \left\{ \mathbb{E} \left[\mathbf{n}(k)\boldsymbol{\gamma}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\}
\end{aligned} \tag{4.33}$$

Na próxima seção, com a finalidade de chegarmos em fórmulas fechadas para o EMSE do SM-AP, assumiremos que as matrizes de correlação presentes na equação acima são predominantemente diagonais. Note que, sob essa assunção, as aproximações $\mathbb{E} \left[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k) \right] \approx \mathbb{E} \left[\boldsymbol{\gamma}(k)\tilde{\mathbf{e}}^T(k) \right]$ e $\mathbb{E} \left[\boldsymbol{\gamma}(k)\mathbf{n}^T(k) \right] \approx \mathbb{E} \left[\mathbf{n}(k)\boldsymbol{\gamma}^T(k) \right]$ são válidas, e a equação acima é reduzida a

$$\begin{aligned}
& (2 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + 2(1 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad - P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2(1 - P_{\text{up}}(k)\mu)\text{tr} \left\{ \mathbb{E} \left[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad \quad \quad + 2P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\boldsymbol{\gamma}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad \quad \quad = P_{\text{up}}(k)\mu\text{tr} \left\{ \mathbb{E} \left[\mathbf{n}(k)\mathbf{n}^T(k) \right] \mathbb{E} \left[\hat{\mathbf{S}}(k) \right] \right\}
\end{aligned} \tag{4.34}$$

Nesta seção derivamos uma importante relação que é válida para o GenSM-AP com qualquer escolha possível para o vetor $\boldsymbol{\gamma}(k)$. Agora, prosseguiremos nossa análise assumindo uma escolha específica para esse vetor.

4.4 Vetor de Restrição com Módulo Constante Baseado no Erro

Nesta seção trabalharemos apenas com o vetor de restrição $\boldsymbol{\gamma}(k)$ do GenSM-AP escolhido conforme a Definição 3. Nas subseções que se seguem, calcularemos cada uma das matrizes necessárias para computarmos o EMSE em (4.34).

Definição 3 (Vetor de Restrição com Módulo Constante Baseado no Erro). Todo vetor de restrição $\boldsymbol{\gamma}(k)$ que é escolhido conforme

$$\gamma_i(k) = \bar{\gamma} \text{ sign} [e_i(k)], \text{ para } i = 0, 1, \dots, L \tag{4.35}$$

é dito um vetor de restrição com módulo constante baseado no erro.

4.4.1 Aproximação para $E[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k)]$

Examinando a $\{i + 1\}$ -ésima linha das equações (4.24) e (4.25) temos

$$\tilde{\varepsilon}_i(k) = -\mathbf{x}^T(k - i)\Delta\mathbf{w}(k + 1) \quad (4.36)$$

e

$$\tilde{\varepsilon}_i(k) = -\mathbf{x}^T(k - i)\Delta\mathbf{w}(k) = e_i(k) - n(k - i) \quad (4.37)$$

para $i = 0, \dots, L$. Usando na equação (4.23) o fato de que $\hat{\mathbf{R}}(k)\hat{\mathbf{S}}(k) \approx \mathbf{I}$ para δ pequeno, temos

$$-\tilde{\boldsymbol{\varepsilon}}(k) = -\tilde{\mathbf{e}}(k) + P_{\text{up}}(k)\mu[\mathbf{e}(k) - \boldsymbol{\gamma}(k)] \quad (4.38)$$

cuja $\{i + 1\}$ -ésima linha é dada por

$$\tilde{\varepsilon}_i(k) = \tilde{e}_i(k) - P_{\text{up}}(k)\mu[e_i(k) - \gamma_i(k)] \quad (4.39)$$

Usando a relação $e_i(k) = \tilde{e}_i(k) + n(k - i)$, veja (4.37), a equação acima pode ser escrita como

$$\tilde{\varepsilon}_i(k) = (1 - P_{\text{up}}(k)\mu)\tilde{e}_i(k) - P_{\text{up}}(k)\mu(n(k - i) - \gamma_i(k)) \quad (4.40)$$

E elevando ao quadrado, temos

$$\begin{aligned} \tilde{\varepsilon}_i^2(k) &= (1 - P_{\text{up}}(k)\mu)^2\tilde{e}_i^2(k) + (P_{\text{up}}(k)\mu)^2(n(k - i) - \gamma_i(k))^2 \\ &\quad - 2P_{\text{up}}(k)\mu(1 - P_{\text{up}}(k)\mu)\tilde{e}_i(k)(n(k - i) - \gamma_i(k)) \end{aligned} \quad (4.41)$$

Note que o vetor de erro-sem-ruído *a posteriori* é relacionado ao erro-sem-ruído *a priori* através da relação (4.42)

$$\begin{aligned} \tilde{\varepsilon}_i(k - 1) &= -\mathbf{x}^T(k - 1 - i)\Delta\mathbf{w}(k) \\ &= -\mathbf{x}^T(k - (i + 1))\Delta\mathbf{w}(k) \\ &= \tilde{\varepsilon}_{i+1}(k) \end{aligned} \quad (4.42)$$

Então, considerando a equação (4.41) na iteração $k - 1$, substituindo (4.42) na mesma, e tomando o valor esperado, obtemos

$$\begin{aligned} E[\tilde{\varepsilon}_{i+1}^2(k)] &= (1 - P_{\text{up}}\mu)^2E[\tilde{\varepsilon}_i^2(k - 1)] + (P_{\text{up}}\mu)^2E[(n(k - 1 - i) - \gamma_i(k - 1))^2] \\ &\quad - 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu)E[\tilde{\varepsilon}_i(k - 1)(n(k - 1 - i) - \gamma_i(k - 1))] \end{aligned} \quad (4.43)$$

onde consideramos que, em regime permanente, $P_{\text{up}}(k)$ é uma constante P_{up} .

Expandindo a equação acima temos

$$\begin{aligned}
\mathbb{E} [\tilde{e}_{i+1}^2(k)] &= (1 - P_{\text{up}}\mu)^2 \mathbb{E} [\tilde{e}_i^2(k-1)] + (P_{\text{up}}\mu)^2 \mathbb{E} [n^2(k-1-i)] + (P_{\text{up}}\mu)^2 \mathbb{E} [\gamma_i^2(k-1)] \\
&\quad - 2(P_{\text{up}}\mu)^2 \mathbb{E} [n(k-1-i)\gamma_i(k-1)] \\
&\quad - 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \mathbb{E} [\tilde{e}_i(k-1)n(k-1-i)] \\
&\quad + 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \mathbb{E} [\tilde{e}_i(k-1)\gamma_i(k-1)]
\end{aligned} \tag{4.44}$$

que pode ser simplificada usando a equação (4.35), usando a relação $\mathbb{E} [\gamma_i^2(k-1)] = \bar{\gamma}^2$, para $i = 0, 1, \dots, L$, e usando $\mathbb{E} [n^2(k)] = \sigma_n^2, \forall k$ (veja 2), levando a

$$\begin{aligned}
\mathbb{E} [\tilde{e}_{i+1}^2(k)] &= (1 - P_{\text{up}}\mu)^2 \mathbb{E} [\tilde{e}_i^2(k-1)] + (P_{\text{up}}\mu)^2 \sigma_n^2 + (P_{\text{up}}\mu)^2 \bar{\gamma}^2 \\
&\quad - 2(P_{\text{up}}\mu)^2 \bar{\gamma} \mathbb{E} [n(k-1-i) \text{sign}[e_i(k-1)]] \\
&\quad - 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \mathbb{E} [\tilde{e}_i(k-1)n(k-1-i)] \\
&\quad + 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \bar{\gamma} \mathbb{E} [\tilde{e}_i(k-1) \text{sign}[e_i(k-1)]]
\end{aligned} \tag{4.45}$$

Teorema 2 (Teorema de Price). *Considere duas variáveis aleatórias a e \mathbf{b} . Se assumirmos que essas variáveis são conjuntamente Gaussianas [21], a seguinte aproximação é válida*

$$\mathbb{E} [\text{sign}[a]\mathbf{b}] \approx \sqrt{\frac{2}{\pi\sigma_a^2}} \mathbb{E} [\mathbf{b}a] \tag{4.46}$$

onde σ_a^2 é a variância de a .

A prova do Teorema 2 pode ser encontrada em [52, 21] e, por não ser central em nossa discussão, será omitida.

Assumindo que $n(k-1-i)$ e $e_i(k-1)$, assim como $\tilde{e}_i(k-1)$ e $e_i(k-1)$ são duas sequências aleatórias conjuntamente gaussianas, podemos aplicar o Teorema 2 na equação (4.45) para obtermos a seguinte relação

$$\begin{aligned}
\mathbb{E} [\tilde{e}_{i+1}^2(k)] &= (1 - P_{\text{up}}\mu)^2 \mathbb{E} [\tilde{e}_i^2(k-1)] + (P_{\text{up}}\mu)^2 \sigma_n^2 + (P_{\text{up}}\mu)^2 \bar{\gamma}^2 \\
&\quad - 2(P_{\text{up}}\mu)^2 \bar{\gamma} \rho_i(k-1) \mathbb{E} [n(k-1-i)e_i(k-1)] \\
&\quad - 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \mathbb{E} [\tilde{e}_i(k-1)n(k-1-i)] \\
&\quad + 2P_{\text{up}}\mu(1 - P_{\text{up}}\mu) \bar{\gamma} \rho_i(k-1) \mathbb{E} [\tilde{e}_i(k-1)e_i(k-1)]
\end{aligned} \tag{4.47}$$

onde

$$\rho_i(k) = \sqrt{\frac{2}{\pi \mathbb{E} [e_i^2(k)]}} \tag{4.48}$$

Agora, utilizamos a relação $e_i(k-1) = \tilde{e}_i(k-1) + n(k-1-i)$ com a finalidade de remover a dependência do sinal de erro *a priori* da equação acima e, após re-arranjar os termos temos

$$\begin{aligned} \mathbb{E} [\tilde{e}_{i+1}^2(k)] &= [1 - P_{\text{up}}\mu + 2P_{\text{up}}\mu\bar{\gamma}\rho_i(k)] (1 - P_{\text{up}}\mu) \mathbb{E} [\tilde{e}_i^2(k)] \\ &\quad + [\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\rho_i(k)\sigma_n^2] (P_{\text{up}}\mu)^2 \end{aligned} \quad (4.49)$$

onde assumimos que o algoritmo já atingiu a convergência de forma que podemos trocar $\mathbb{E} [\tilde{e}_i^2(k-1)]$ e $\mathbb{E} [e_i^2(k-1)]$ por $\mathbb{E} [\tilde{e}_i^2(k)]$ e $\mathbb{E} [e_i^2(k)]$, respectivamente. Além disso, também desprezamos os termos dependentes de $\mathbb{E} [n(k-1-i)\tilde{e}_i(k-1)]$, pois eles são muito menores que os demais, por exemplo, para $i = 0$ temos $\mathbb{E} [n(k-1)\tilde{e}_0(k-1)] = 0$, devido à Definição 2 e Proposição 1.

Faremos agora a seguinte assunção

$$\rho_i(k) \approx \rho_0(k) , \text{ para } i = 0, 1, \dots, L \quad (4.50)$$

com a finalidade de simplificar as manipulações matemáticas posteriores. Usando essa assunção, podemos reescrever a recursão dada pela equação (4.49) como

$$\mathbb{E} [\tilde{e}_{i+1}^2(k)] = a\mathbb{E} [\tilde{e}_i^2(k)] + b(P_{\text{up}}\mu)^2 \quad (4.51)$$

onde

$$a = [1 - P_{\text{up}}\mu + 2P_{\text{up}}\mu\bar{\gamma}\rho_0(k)] (1 - P_{\text{up}}\mu) \quad (4.52)$$

$$b = [\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\rho_0(k)\sigma_n^2] \quad (4.53)$$

Agora, vamos resolver a recursão acima. Para $i = 0$ temos

$$\mathbb{E} [\tilde{e}_1^2(k)] = a\mathbb{E} [\tilde{e}_0^2(k)] + b(P_{\text{up}}\mu)^2 \quad (4.54)$$

Para $i = 1$ temos

$$\begin{aligned} \mathbb{E} [\tilde{e}_2^2(k)] &= a\mathbb{E} [\tilde{e}_1^2(k)] + b(P_{\text{up}}\mu)^2 \\ &= a^2\mathbb{E} [\tilde{e}_0^2(k)] + (1+a)b(P_{\text{up}}\mu)^2 \end{aligned} \quad (4.55)$$

Por indução, é fácil mostrar que

$$\mathbb{E} [\tilde{e}_{i+1}^2(k)] = a^{(i+1)}\mathbb{E} [\tilde{e}_0^2(k)] + \left(\sum_{l=0}^i a^l \right) b(P_{\text{up}}\mu)^2 \quad (4.56)$$

Então, assumindo que $E[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k)]$ é predominantemente diagonal, que é o que costuma ocorrer na prática, podemos escrever

$$E[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k)] = \mathbf{A}_1 E[\tilde{e}_0^2(k)] + \mathbf{A}_2 b (P_{\text{up}}\mu)^2 \quad (4.57)$$

com

$$\mathbf{A}_1 = \begin{bmatrix} 1 & & & & \\ & a & & \mathbf{0} & \\ & & a^2 & & \\ & \mathbf{0} & & \ddots & \\ & & & & a^L \end{bmatrix}$$

e

$$\mathbf{A}_2 = \begin{bmatrix} 0 & & & & \\ & 1 & & \mathbf{0} & \\ & & 1+a & & \\ & \mathbf{0} & & \ddots & \\ & & & & \sum_{l=0}^{L-1} a^L \end{bmatrix}$$

4.4.2 Aproximação para $E[\mathbf{n}(k)\mathbf{n}^T(k)]$

Assumindo que o ruído é branco e usando a Definição 2 temos

$$E[\mathbf{n}(k)\mathbf{n}^T(k)] = \sigma_n^2 \mathbf{I} \quad (4.58)$$

onde $E[\mathbf{n}(k)\mathbf{n}^T(k)] \in \mathbb{R}^{L+1 \times L+1}$.

4.4.3 Aproximação para $E[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k)]$

Como os elementos da matriz $E[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k)]$ são bem menores que os demais, consideraremos

$$E[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k)] \approx \mathbf{0} \quad (4.59)$$

4.4.4 Aproximação para $E[\boldsymbol{\gamma}(k)\mathbf{n}^T(k)]$

Usando a assunção de diagonal dominante (ADD) para a matriz $E[\boldsymbol{\gamma}(k)\mathbf{n}^T(k)]$, utilizando a definição de $\boldsymbol{\gamma}(k)$, o Teorema 2, a relação $\mathbf{e}(k) = \tilde{\mathbf{e}}(k) + \mathbf{n}(k)$, e a

aproximação dada pela equação (4.59), temos

$$\mathbf{E} [\boldsymbol{\gamma}(k)\mathbf{n}^T(k)] \approx \bar{\gamma}\sigma_n^2\mathbf{C} + \bar{\gamma}\mathbf{E} [\tilde{\mathbf{e}}(k)\mathbf{n}^T(k)] \mathbf{C} \approx \bar{\gamma}\sigma_n^2\mathbf{C}$$

onde

$$\mathbf{C} = \begin{bmatrix} \rho_0(k) & & & & & \\ & \rho_1(k) & & & & \\ & & \rho_2(k) & & & \\ & & & \ddots & & \\ & & & & \rho_L(k) & \\ & & & & & \end{bmatrix}$$

e, se considerarmos a assunção (4.50), podemos simplificar a expressão acima para

$$\mathbf{E} [\boldsymbol{\gamma}(k)\mathbf{n}^T(k)] \approx \bar{\gamma}\sigma_n^2\rho_0(k)\mathbf{I}_{L+1} \quad (4.60)$$

4.4.5 Aproximação para $\mathbf{E}[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k)]$

Novamente, usando a ADD para a matriz $\mathbf{E}[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k)]$, temos

$$\mathbf{E} [\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k)] \approx \bar{\gamma}^2\mathbf{I}_{L+1} \quad (4.61)$$

4.4.6 Aproximação para $\mathbf{E}[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k)]$

Por fim, invocando mais uma vez a ADD para a matriz $\mathbf{E}[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k)]$, utilizando a definição de $\boldsymbol{\gamma}(k)$, o Teorema 2, a relação $\mathbf{e}(k) = \tilde{\mathbf{e}}(k) + \mathbf{n}(k)$, e considerando a assunção (4.50), temos

$$\begin{aligned} \mathbf{E} [\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k)] &\approx \bar{\gamma}\mathbf{C}\mathbf{E} [\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k)] \\ &\approx \bar{\gamma}\rho_0(k) \left\{ \mathbf{A}_1\mathbf{E} [\tilde{\mathbf{e}}_0(k)] + \mathbf{A}_2b(P_{\text{up}}\mu)^2 \right\} \end{aligned} \quad (4.62)$$

4.4.7 EMSE para SM-AP

Agora, por conveniência, começamos reescrevendo a relação de energia (4.34), considerando que o algoritmo já está operando em regime permanente de forma que

possa substituir $P_{\text{up}}(k)$ por P_{up} , levando a

$$\begin{aligned}
& (2 - P_{\text{up}}\mu)\text{tr} \left\{ \mathbf{E} \left[\tilde{\mathbf{e}}(k)\tilde{\mathbf{e}}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + 2(1 - P_{\text{up}}\mu)\text{tr} \left\{ \mathbf{E} \left[\tilde{\mathbf{e}}(k)\mathbf{n}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& - P_{\text{up}}\mu\text{tr} \left\{ \mathbf{E} \left[\boldsymbol{\gamma}(k)\boldsymbol{\gamma}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2(1 - P_{\text{up}}\mu)\text{tr} \left\{ \mathbf{E} \left[\tilde{\mathbf{e}}(k)\boldsymbol{\gamma}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& \quad + 2P_{\text{up}}\mu\text{tr} \left\{ \mathbf{E} \left[\boldsymbol{\gamma}(k)\mathbf{n}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& = P_{\text{up}}\mu\text{tr} \left\{ \mathbf{E} \left[\mathbf{n}(k)\mathbf{n}^T(k) \right] \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\}
\end{aligned}$$

Note que todas as matrizes de correlação presentes nessa equação já foram calculadas nas subseções anteriores. Portanto, substituindo as equações (4.57), (4.58), (4.59), (4.60), (4.61), e (4.62) na equação acima, temos

$$\begin{aligned}
& (2 - P_{\text{up}}\mu)\mathbf{E} \left[\tilde{\epsilon}_0^2(k) \right] \text{tr} \left\{ \mathbf{A}_1 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + (2 - P_{\text{up}}\mu)b(P_{\text{up}}\mu)^2 \text{tr} \left\{ \mathbf{A}_2 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& - P_{\text{up}}\mu\bar{\gamma}^2 \text{tr} \left\{ \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k)\mathbf{E} \left[\tilde{\epsilon}_0^2(k) \right] \text{tr} \left\{ \mathbf{A}_1 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k)b(P_{\text{up}}\mu)^2 \text{tr} \left\{ \mathbf{A}_2 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} + 2P_{\text{up}}\mu\bar{\gamma}\sigma_n^2\rho_0(k)\text{tr} \left\{ \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \\
& = P_{\text{up}}\mu\sigma_n^2 \text{tr} \left\{ \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \quad (4.63)
\end{aligned}$$

Conforme sabemos, $0 \leq P_{\text{up}} \leq 1$ e μ deve ser escolhido tal que $0 < \mu \leq 1$. Então, considerando que $P_{\text{up}} \ll 1$, ou $\mu \ll 1$, ou o produto $P_{\text{up}}\mu \ll 1$, os termos que dependem de $(P_{\text{up}}\mu)^2$ são muito menores que os outros e podem ser desprezados, levando a

$$\begin{aligned}
& [(2 - P_{\text{up}}\mu) - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k)] \text{tr} \left\{ \mathbf{A}_1 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \mathbf{E} \left[\tilde{\epsilon}_0^2(k) \right] \\
& = P_{\text{up}}\mu \left[\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\sigma_n^2\rho_0(k) \right] \text{tr} \left\{ \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\} \quad (4.64)
\end{aligned}$$

Re-arranjando a equação acima e considerando que os elementos da diagonal principal de $\mathbf{E} \left[\hat{\mathbf{S}}(k) \right]$ são iguais, conseguimos obter uma expressão para o EMSE do GenSM-AP

$$\begin{aligned}
\text{EMSE}_{\text{GenSM-AP}} & = \mathbf{E} \left[\tilde{\epsilon}_0^2(k) \right] \\
& = \frac{[\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\sigma_n^2\rho_0(k)] P_{\text{up}}\mu \text{tr} \left\{ \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\}}{[(2 - P_{\text{up}}\mu) - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k)] \text{tr} \left\{ \mathbf{A}_1 \mathbf{E} \left[\hat{\mathbf{S}}(k) \right] \right\}} \\
& = \frac{(L+1) [\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\sigma_n^2\rho_0(k)] P_{\text{up}}\mu}{[(2 - P_{\text{up}}\mu) - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k)]} \left(\frac{1-a}{1-a^{L+1}} \right) \quad (4.65)
\end{aligned}$$

onde $a = [1 - P_{\text{up}}\mu + 2P_{\text{up}}\mu\bar{\gamma}\rho_0(k)](1 - P_{\text{up}}\mu)$ conforme definido em (4.52). O de-

sajuste do GenSM-AP é dado por

$$\begin{aligned} M_{\text{GenSM-AP}} &= \frac{\text{EMSE}_{\text{GenSM-AP}}}{\sigma_n^2} \\ &= \frac{(L+1) \left[1 + \frac{\bar{\gamma}^2}{\sigma_n^2} - 2\bar{\gamma}\rho_0(k) \right] P_{\text{up}}\mu}{\left[(2 - P_{\text{up}}\mu) - 2(1 - P_{\text{up}}\mu)\bar{\gamma}\rho_0(k) \right]} \left(\frac{1-a}{1-a^{L+1}} \right) \end{aligned} \quad (4.66)$$

Fazendo $\mu = 1$, a equação (4.65) se torna o EMSE do algoritmo SM-AP com vetor de restrição com módulo fixo baseado no erro

$$\begin{aligned} \text{EMSE}_{\text{SM-AP}} &= \text{E} \left[\tilde{e}_0^2(k) \right]_{\mu=1} \\ &= \frac{(L+1) \left[\sigma_n^2 + \bar{\gamma}^2 - 2\bar{\gamma}\sigma_n^2\rho_0(k) \right] P_{\text{up}}}{\left[(2 - P_{\text{up}}) - 2(1 - P_{\text{up}})\bar{\gamma}\rho_0(k) \right]} \left(\frac{1-\bar{a}}{1-\bar{a}^{L+1}} \right) \end{aligned} \quad (4.67)$$

onde $\bar{a} = [1 - P_{\text{up}} + 2P_{\text{up}}\bar{\gamma}\rho_0(k)](1 - P_{\text{up}})$. O desajuste do SM-AP é dado por

$$M_{\text{SM-AP}} = \frac{(L+1) \left[1 + \frac{\bar{\gamma}^2}{\sigma_n^2} - 2\bar{\gamma}\rho_0(k) \right] P_{\text{up}}}{\left[(2 - P_{\text{up}}) - 2(1 - P_{\text{up}})\bar{\gamma}\rho_0(k) \right]} \left(\frac{1-\bar{a}}{1-\bar{a}^{L+1}} \right) \quad (4.68)$$

e, uma vez que consideramos $\mu = 1$, a acurácia das aproximações dadas pelas equações (4.67) e (4.68) irá aumentar conforme P_{up} se aproxima de zero.

Note que as expressões de EMSE e Desajuste estão em função de P_{up} e $\rho_0(k)$ e, até o presente momento, nada foi dito a respeito dessas variáveis. Elas serão estudadas na próxima subseção.

4.4.8 Eurísticas sobre $\rho_0(k)$ e P_{up}

Escolha de $\rho_0(k)$:

Pela equação (4.48), temos que

$$\rho_0(k) = \sqrt{\frac{2}{\pi \text{E}[e_0^2(k)]}} \quad (4.69)$$

Portanto, arbitrar $\rho_0(k)$ significa arbitrar $\text{E}[e_0^2(k)]$. Contudo, sabendo $\text{E}[e_0^2(k)]$, descobrir o $\text{E}[\tilde{e}_0^2(k)]$ é imediato. E mais, sendo matematicamente rigoroso, as equações (4.67) e (4.68) deveriam ser resolvidas em função de $\text{E}[\tilde{e}_0^2(k)]$ e as suas soluções serem analisadas como possíveis candidatas ao EMSE e ao desajuste do SM-AP, respectivamente. Esse é, de fato, um trabalho que ainda pretendemos investigar.

No entanto, em [15], os autores optaram por fazer uma análise bastante rigorosa, do ponto de vista matemático, do SM-NLMS e descobrir o MSE através da solução de

uma equação. Porém, para derivar essa equação, aproximações fortes, possivelmente inválidas para a maior parte dos casos, foram utilizadas. Consequentemente, o resultado final de [15] apresenta inconsistência com os resultados práticos, como será verificado no próximo capítulo. Esse foi o principal motivo de termos optado por fazer uma assunção baseada em experimentos, sendo mais coerente com o modelo e com as aproximações realizadas ao longo da derivação, ao invés de tentar resolver a equação.

Além disso, durante a derivação da expressão para o EMSE do SM-AP, utilizamos a aproximação (4.50) que, em geral, não é válida para $L > 0$. Isso reforça a idéia de que a simples solução da equação (4.67) talvez leve a um resultado inconsistente com a prática. Sendo assim, $E[e_0^2(k)]$ foi escolhido da seguinte forma:

$$E[e_0^2(k)] = 2\sigma_n^2 + \frac{1}{L+1}\bar{\gamma}^2 \quad (4.70)$$

A aproximação dada acima é um refinamento da aproximação utilizada em [16] para uma versão simplificada do algoritmo SM-AP. Com base em observações experimentais, notamos que o algoritmo SM-AP com vetor de restrição com módulo constante baseado no erro é mais sensível ao fator de reuso de memória L que a versão do SM-AP considerada em [16] (*simple-choice*) e, portanto, resolvemos incorporá-lo na expressão.

Escolha de P_{up} :

Pelo teorema central do limite [21, 53], sabemos que, para um k suficientemente grande, o sinal de erro $e_0(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$ terá distribuição gaussiana, pois ele é formado pela soma de várias variáveis aleatórias independentes⁴, até o instante k . Além disso, devido ao modelo estacionário (vide Definição 2), ele terá média 0. Portanto, a probabilidade do filtro adaptativo atualizar, em uma dada iteração k , é dada por

$$P_{\text{up}}(k) = P[|e_0(k)| > \gamma] \quad (4.71)$$

⁴Essa afirmação é válida para o caso em que o sinal de entrada é branco e, quanto mais colorido for o sinal de entrada, menos válida será essa assunção de distribuição gaussiana para a variável erro.

e, após a convergência, essa probabilidade de atualização é dada por

$$P_{\text{up}} = 2Q\left(\frac{\gamma}{\sigma_e}\right) \quad (4.72)$$

onde σ_e^2 é a variância de $e_0(k)$, e $Q(\cdot)$ é a função distribuição cumulativa complementar da Gaussiana, definida como

$$Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \quad (4.73)$$

Recorrendo à teoria da independência, sabemos que a variância do erro pode ser escrita da seguinte forma

$$\sigma_e^2 = \sigma_n^2 + \text{E} \left[\Delta \mathbf{w}^T(k) \mathbf{R} \Delta \mathbf{w}(k) \right] \quad (4.74)$$

Como não queremos assumir um modelo para o sinal de entrada do filtro adaptativo, utilizaremos o quociente de Rayleigh [4, 22] para descobrir a região de possíveis valores para σ_e^2

$$\sigma_n^2 + \lambda_{\min} \text{E} \left[\|\Delta \mathbf{w}(k)\|^2 \right] \leq \sigma_e^2 \leq \sigma_n^2 + \lambda_{\max} \text{E} \left[\|\Delta \mathbf{w}(k)\|^2 \right] \quad (4.75)$$

onde λ_{\min} e λ_{\max} são o menor e o maior autovalor de \mathbf{R} , respectivamente.

O modelo proposto para σ_e^2 é

$$\begin{aligned} \sigma_e^2 &= \sigma_n^2 + \lambda_g \alpha \sigma_n^2 \\ &= (1 + \lambda_g \alpha) \sigma_n^2 \end{aligned} \quad (4.76)$$

onde

$$\lambda_g = \sqrt{\lambda_{\min} \lambda_{\max}} \quad (4.77)$$

$$\alpha \quad \text{constante próxima de 0} \quad (4.78)$$

Tabela 4.1: Escolha de α em função da L e da razão de autovalores.

$L \setminus$ entrada	branca	$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$	$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$
0	0,08	0,07	0,06
1	0,08	0,25	0,25
2	0,08	0,25	0,25
3	0,08	0,25	0,25
4	0,08	0,25	0,25

Notamos, por meio de simulações, que quando alterávamos somente α , as características da curva de desajuste teórico eram conservadas, porém ela sofria um translado vertical, i.e., se deslocava para cima ou para baixo. Com base nessa observação, chegamos nos valores de α apresentados na Tabela 4.1. É interessante notar que, para $L \neq 0$, os valores de α dependem apenas da razão de autovalores.

Por fim, a aproximação para P_{up} dada em (4.72), não leva em consideração que a probabilidade de atualização em regime permanente aumenta junto com L . Dessa forma, para chegarmos em um valor de P_{up} adequado para o modelo precisamos somar uma constante, o que nos levou à seguinte expressão:

$$P_{\text{up}} = 2Q \left(\frac{\gamma}{\sigma_e} \right) + P_{\text{min}} \quad (4.79)$$

onde P_{min} é uma estimativa do menor valor que P_{up} assume em função de $\bar{\gamma}$.

A Tabela 4.2 resume os valores de P_{min} que foram utilizados nos experimentos. Esses valores foram obtidos a partir de várias simulações em diferentes cenários. Além disso, a aproximação (4.79) não é muito sensível a variações pequenas de P_{min} .

4.5 Vetor Restrição Trivial

Outra escolha interessante para o vetor de restrição $\gamma(k)$ é

$$\gamma(k) = \mathbf{0} \quad (4.80)$$

que é conhecido como vetor de restrição trivial.

Assumindo um vetor $\gamma(k)$ trivial, o algoritmo GenSM-AP dado pela equação (4.12) se torna o AP [4].

Tabela 4.2: Escolha de P_{min} em função de L e da razão de autovalores.

$L \setminus$ entrada	branca	$\frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} = 20$	$\frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} = 80$
0	0	0	0
1	0, 15	0, 20	0, 20
2	0, 30	0, 38	0, 38
3	0, 35	0, 43	0, 43
4	0, 40	0, 50	0, 50

Note que o vetor trivial pode ser mapeado em um vetor de restrição com módulo constante baseado no erro através da escolha $\bar{\gamma} = 0$ (vide equação (4.35)) e, portanto, os resultados derivados na seção anterior também se aplicam aqui. Note também que quando $\bar{\gamma} = 0$ temos $P_{\text{up}} = 1$ (vide equação (4.72)), e a equação (4.65) se torna

$$\text{EMSE}_{\text{APA}} = \text{E} \left[\tilde{e}_0^2(k) \right]_{\bar{\gamma}=0, P_{\text{up}}=1} = \frac{(L+1)\mu}{(2-\mu)} \left(\frac{1 - (1-\mu)^2}{1 - (1-\mu)^{2(L+1)}} \right) \sigma_n^2 \quad (4.81)$$

que é o EMSE do AP (vide equação (4.123) de [4]). O desajuste do AP é dado por

$$M_{\text{APA}} = \frac{(L+1)\mu}{(2-\mu)} \left(\frac{1 - (1-\mu)^2}{1 - (1-\mu)^{2(L+1)}} \right) \quad (4.82)$$

que também está de acordo com [4] (vide equação (4.124) dessa referência).

Capítulo 5

Resultados

Neste capítulo, a validade das expressões (4.67) e (4.68) será testada, considerando-se diferentes situações. Para todos os cenários, consideraremos o problema de identificação de sistema onde o sistema desconhecido é um filtro FIR (resposta ao impulso de duração finita), e o filtro adaptativo é também um filtro FIR com a mesma ordem do sistema desconhecido.

5.1 Cenário 1

Nesta seção, consideraremos um sistema desconhecido cuja resposta ao impulso é

$$[0,1 \ 0,3 \ 0 \ -0,2 \ -0,4 \ -0,7 \ -0,4 \ -0,2]^T$$

Nas próximas subseções mostraremos os resultados de MSE e desajuste para diferentes valores de L . Para cada L , consideraremos três tipos de entrada:

- razão de autovalores 1 (entrada branca);
- razão de autovalores 20 e
- razão de autovalores 80.

Os sinais de entrada que são coloridos foram gerados a partir de um processo autorregressivo de primeira ordem. Em todos os três casos citados acima a variância do sinal de entrada é 1 e a variância do ruído Gaussiano é 10^{-3} . Além disso, a média de ambas as variáveis aleatórias é 0, respeitando o modelo estacionário.

Para gerar os resultados experimentais de MSE em regime permanente utilizamos o seguinte procedimento. Para cada realização do processo estocástico rodamos o algoritmo adaptativo durante 10^4 iterações. Cabe ressaltar que o algoritmo, para o cenário em questão, converge antes de 500 iterações para todo valor de L . Os resultados experimentais de MSE em regime permanente são obtidos realizando-se uma média temporal das 9×10^3 últimas iterações de cada realização, e em seguida calculando-se a média estatística (das médias temporais de cada realização) ao longo de 100 experimentos realizados de forma independente.

5.1.1 SM-NLMS ($L = 0$)

Nesta subseção apresentamos resultados de simulações para o SM-AP com vetor de restrição com módulo constante baseado no erro e considerando $L = 0$. Para essa configuração, o SM-AP coincide com o SM-NLMS [4].

As Figuras 5.1, 5.2, e 5.3 apresentam, respectivamente, o P_{up} , MSE em regime permanente, e desajuste do SM-NLMS considerando um sinal de entrada branco.

Identificamos com as Figuras 5.4, 5.5, e 5.6, respectivamente, o P_{up} , MSE em regime permanente, e desajuste do SM-NLMS considerando um sinal de entrada colorido, cuja razão de autovalores é 20.

As Figuras 5.7, 5.8, e 5.9 apresentam, para uma entrada colorida com razão de autovalores 80, o P_{up} , MSE em regime permanente, e desajuste do SM-NLMS, respectivamente.

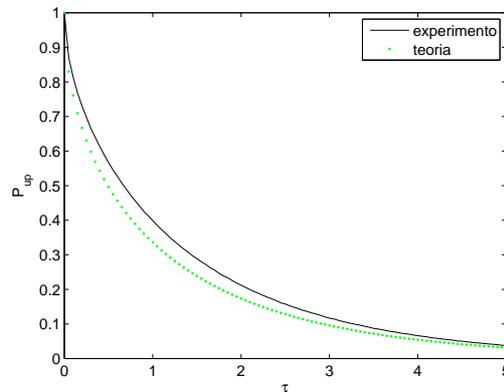


Figura 5.1: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 0$).

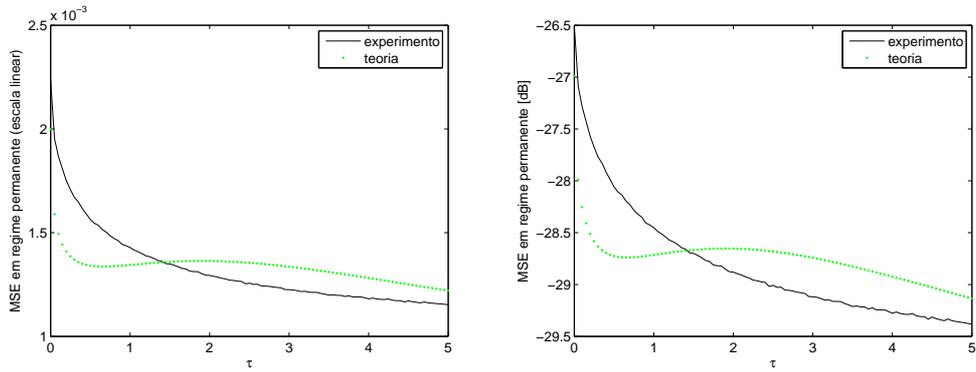


Figura 5.2: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada branca ($L = 0$).

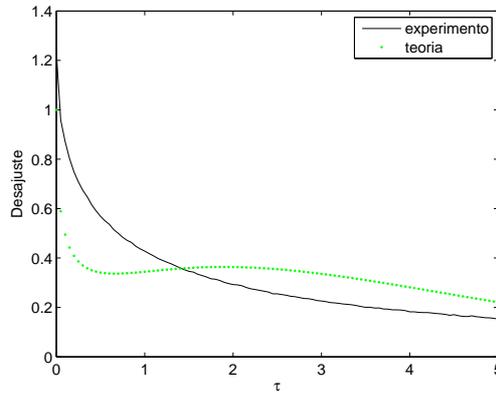


Figura 5.3: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada branca ($L = 0$).

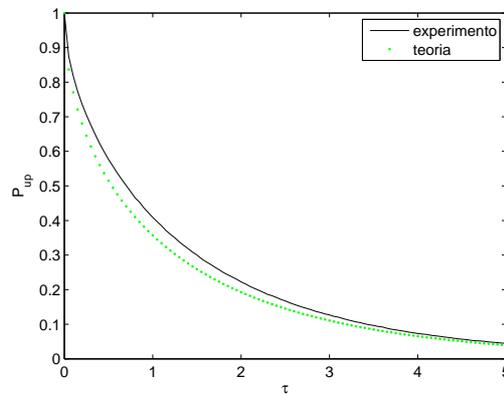


Figura 5.4: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 20 ($L = 0$).

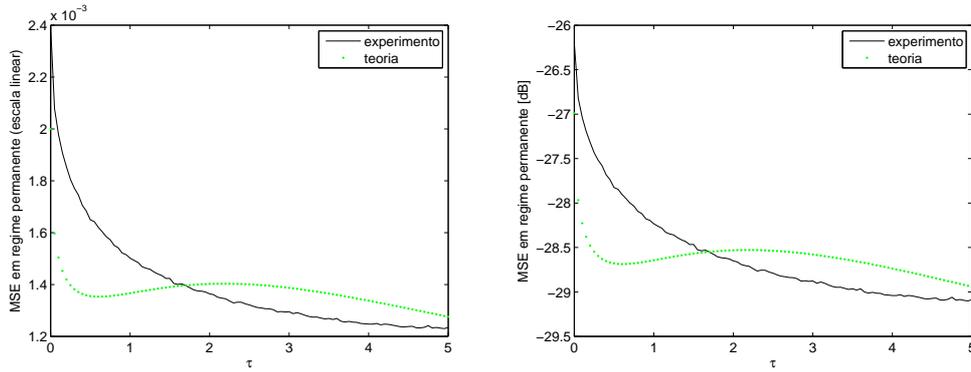


Figura 5.5: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 20 ($L = 0$).

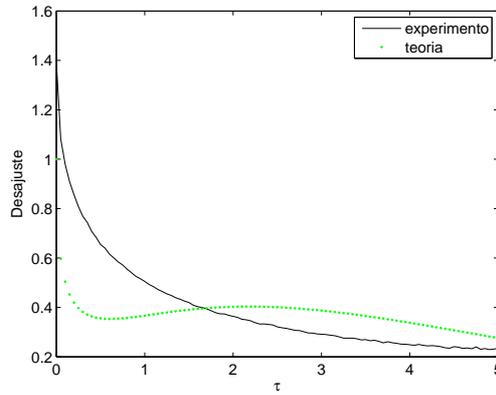


Figura 5.6: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 20 ($L = 0$).

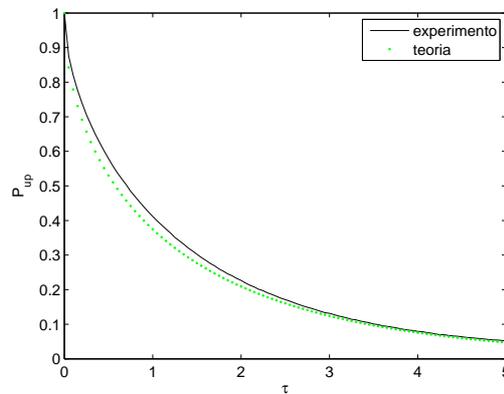


Figura 5.7: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 80 ($L = 0$).

Note, observando as figuras relativas ao desajuste do SM-NLMS, que o desajuste prático desse algoritmo não tende a zero quando $\bar{\gamma} \rightarrow \infty$. Em outras palavras, o MSE em regime permanente não tende a σ_n^2 quando $\bar{\gamma} \rightarrow \infty$, contrariando o resultado obtido em [15]. A Figura 5.10 elimina qualquer dúvida que possa existir a respeito disso, pois ela mostra que o desajuste aumenta a partir de um determinado valor de τ (assinalado com um quadrado na curva prática). O comportamento da curva de desajuste prático confere com a nossa intuição à respeito do algoritmo SM-AP, pois para valores baixos de τ temos o problema de amplificação do ruído (*noise enhancement*) que vai sendo atenuado conforme τ cresce, justificando a parte decrescente da curva e, em seguida, a curva passa a ser crescente com τ , pois com o aumento de $\bar{\gamma}$, maior é o módulo do erro tolerável sem que o algoritmo atualize os

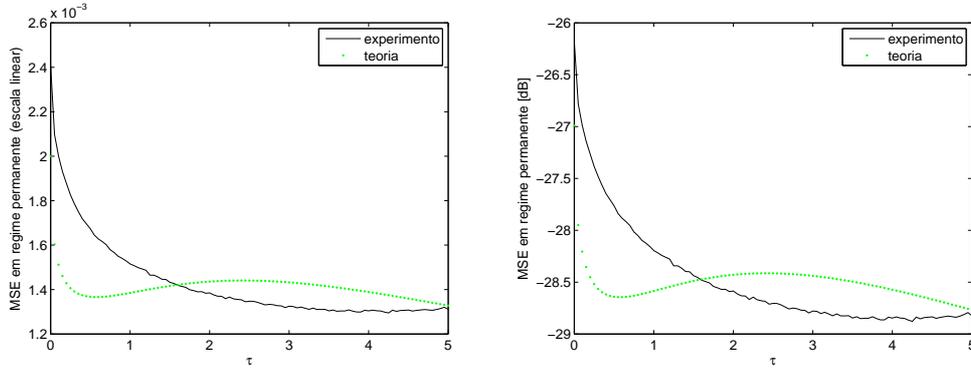


Figura 5.8: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 0$).

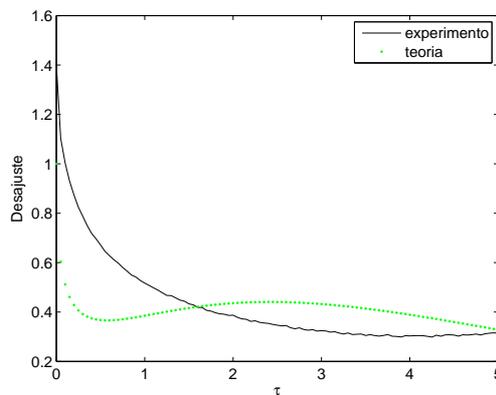


Figura 5.9: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 0$).

coeficientes do filtro adaptativo e, portanto, maior será o MSE.

Analizando os resultados obtidos nesta subseção, conforme pode ser visto nas figuras apresentadas, as curvas teóricas e práticas estão bem próximas. A distância entre o desajuste prático e teórico, no intervalo $\tau \in [0, 5]$ não ultrapassa 0,2. Contudo, percebe-se uma diferença de tendência (derivada) entre as curvas. Logo, nossa curva teórica se distancia da prática para $\tau > 5$. Na prática, para $\tau = 10$ a diferença entre as duas curvas de desajuste chegou a 0,5 no pior caso, i.e., quando a razão de autovalores do sinal de entrada é igual a 80. A causa desse problema de tendência entre as curvas é, possivelmente, o modelo de análise que consideramos, pois quando $L = 0$ e $\bar{\gamma}$ é grande, as hipóteses assumidas para que o modelo de análise seja uma boa aproximação da prática são desrespeitadas, vide Subseção 4.2.1.

Definindo $\tau_o \in [0, 5]$ como sendo o valor de τ para o qual o desajuste é mínimo, temos, na Tabela 5.1, um resumo dos τ_o para os três tipos de sinais de entrada. Pode-se notar que os valores práticos e teóricos são bem próximos.

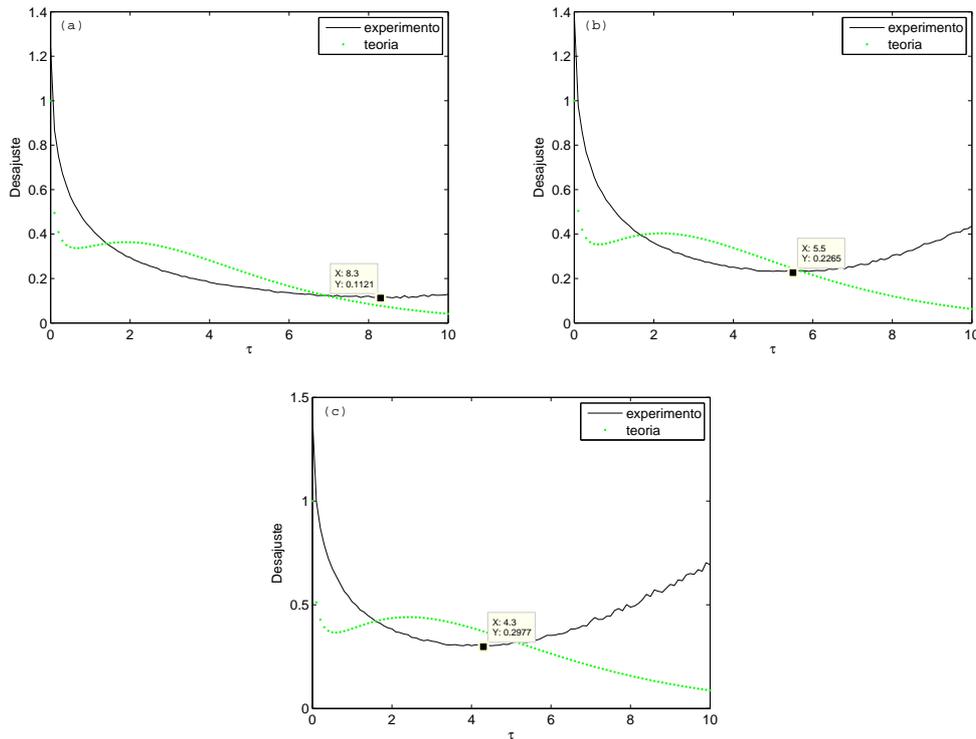


Figura 5.10: Desajuste do SM-NLMS em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – (a) entrada branca (b) entrada com razão de autovalores 20 (c) entrada com razão de autovalores 80 .

Por fim, a Tabela 5.2 mostra alguns resultados de desajuste para quatro escolhas particulares de $\bar{\gamma}$. Pode-se perceber que a escolha usual $\bar{\gamma} = \sqrt{5\sigma_n^2}$ nem sempre é a melhor, em termos de gerar um desajuste menor para o SM-NLMS. A escolha $\bar{\gamma} = \sqrt{4\sigma_n^2}$, por exemplo, seria uma opção melhor no caso em que a razão de autovalores é 80.

5.1.2 SM-BNLMS ($L = 1$)

Nesta subseção, consideraremos o algoritmo SM-BNLMS [4], que é um caso particular do algoritmo SM-AP com vetor de restrição com módulo constante baseado no erro quando fazemos $L = 1$.

As Figuras 5.11, 5.12, e 5.13 apresentam, respectivamente, o P_{up} , MSE em regime permanente, e desajuste do SM-BNLMS considerando um sinal de entrada branco.

Analogamente, as Figuras 5.14, 5.15, e 5.16 mostram, para um sinal de entrada colorido cuja razão de autovalores é 20, o P_{up} , MSE em regime permanente e desajuste do SM-BNLMS.

Considerando um sinal de entrada colorido com razão de autovalores 80, as Figuras 5.17, 5.18, e 5.19 ilustram o P_{up} , MSE em regime permanente, e desajuste do SM-BNLMS, respectivamente.

Assim como na Subseção 5.1.1, os resultados obtidos nesta subseção estão con-

Tabela 5.1: Valores de τ_o considerando $L = 0$.

resultado \ entrada	branca	$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$	$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$
experimental	5,0	4,9	4,3
teórico	5,0	5,0	5,0

Tabela 5.2: Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 0$).

$\bar{\gamma} \setminus$ entrada	branca		$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$		$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$	
	(exp.)	(teo.)	(exp.)	(teo.)	(exp.)	(teo.)
$\sqrt{1\sigma_n^2}$	0,4277	0,3444	0,5054	0,3666	0,5170	0,3851
$\sqrt{2\sigma_n^2}$	0,2925	0,3634	0,3636	0,4025	0,3873	0,4358
$\sqrt{4\sigma_n^2}$	0,1821	0,2813	0,2487	0,3376	0,3033	0,3890
$\sqrt{5\sigma_n^2}$	0,1525	0,2207	0,2342	0,2759	0,3145	0,3280

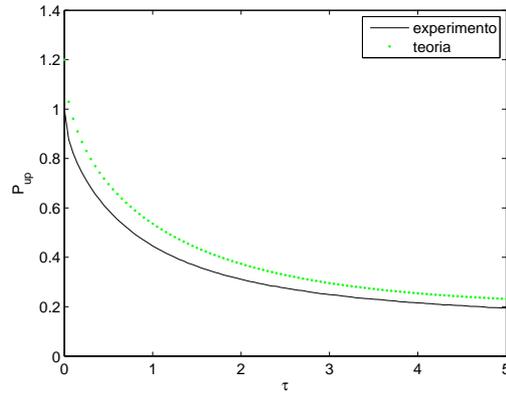


Figura 5.11: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).

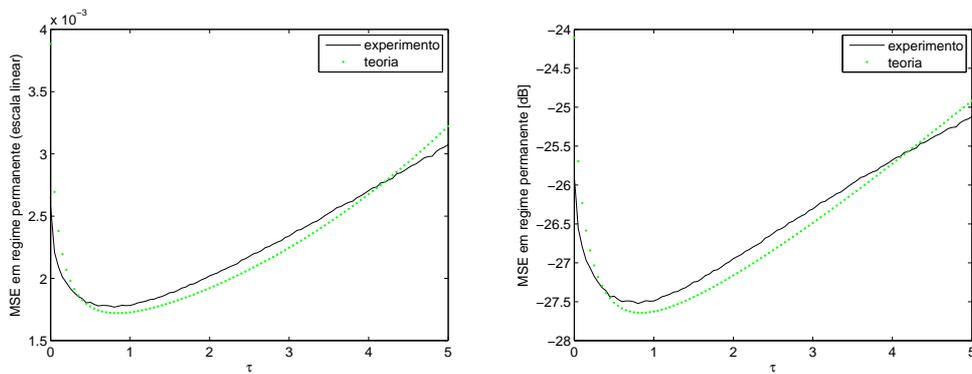


Figura 5.12: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).

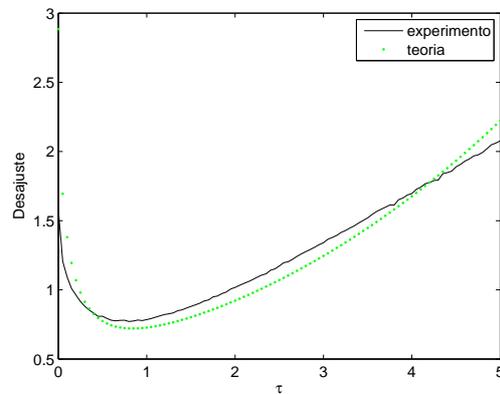


Figura 5.13: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 1$).

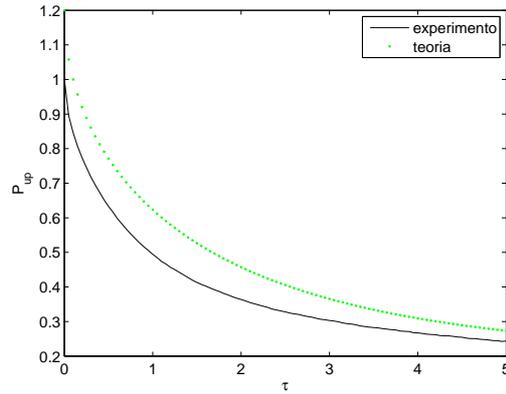


Figura 5.14: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).

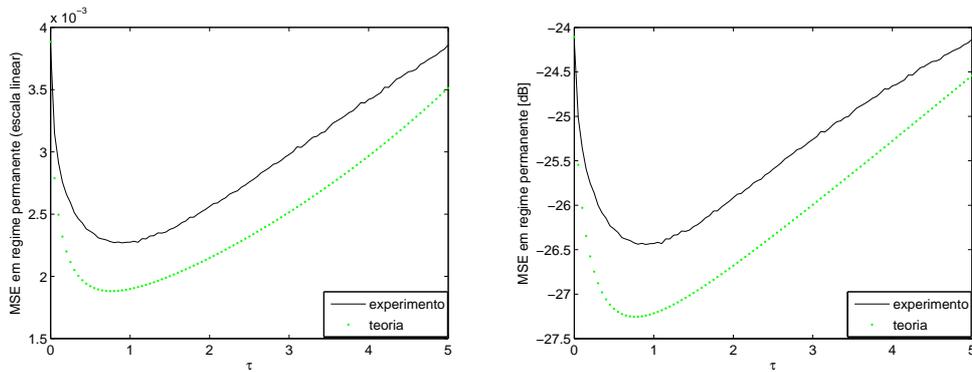


Figura 5.15: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).

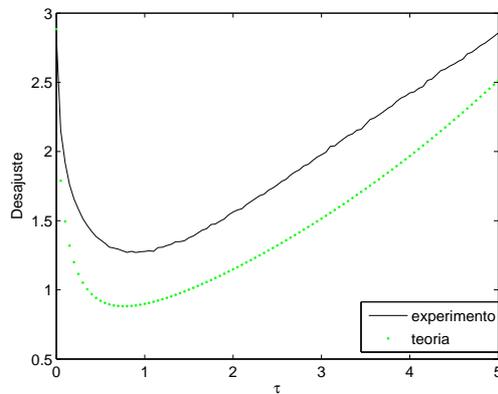


Figura 5.16: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 1$).

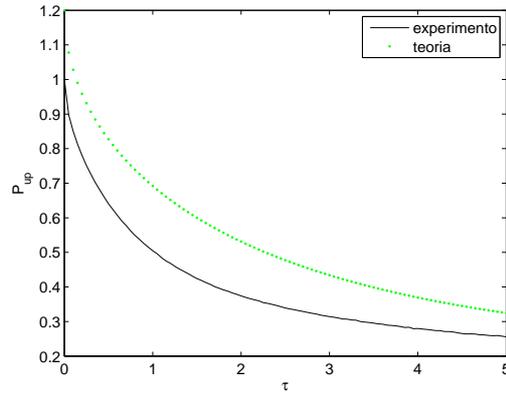


Figura 5.17: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).

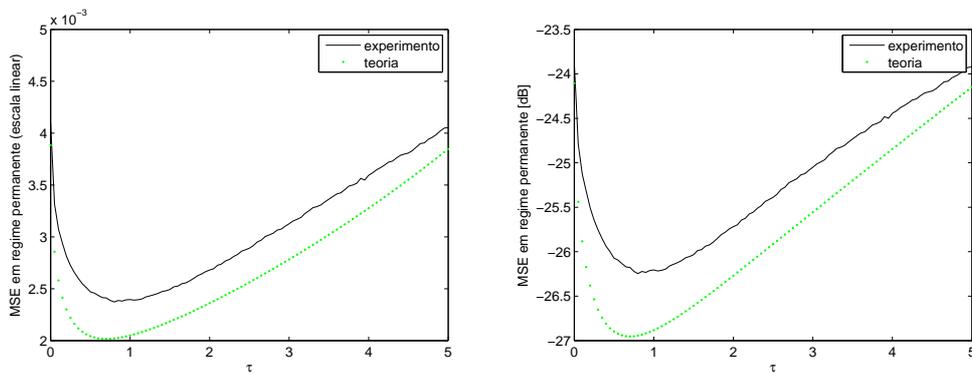


Figura 5.18: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).

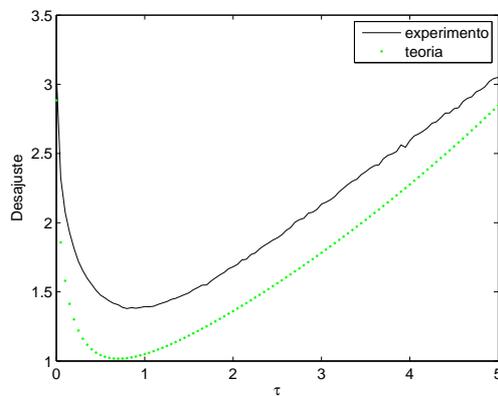


Figura 5.19: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 80 ($L = 1$).

dizentes com a prática. Além disso, pode-se notar que o problema de tendência das curvas de desajuste existente para o SM-NLMS não existe para o caso em que $L = 1$ (SM-BNLMS). A Tabela 5.3 resume os valores de τ_o para os três tipos de sinais de entrada. Novamente, percebemos que os valores teóricos estão bem próximos dos práticos.

Por fim, a Tabela 5.4 mostra alguns resultados de desajuste para quatro escolhas particulares de $\bar{\gamma}$. Pode-se perceber que a escolha usual $\bar{\gamma} = \sqrt{5\sigma_n^2}$ está longe de ser a melhor, em termos de gerar um desajuste menor para o SM-BNLMS. Para essa finalidade, a escolha $\bar{\gamma} = \sqrt{1\sigma_n^2}$ mostrou-se bem superior.

5.1.3 $L = 2$

De forma análoga ao que foi feito nas subseções anteriores, mostraremos, nesta subseção, resultados para o MSE em regime permanente considerando o algoritmo SM-AP com vetor de restrição com módulo constante baseado no erro e com $L = 2$.

As Figuras 5.20, 5.21, e 5.22 apresentam, respectivamente, o P_{up} , MSE em regime permanente, e desajuste do algoritmo considerando um sinal de entrada branco.

Resultados para o P_{up} , MSE em regime permanente, e desajuste do algoritmo considerando um sinal de entrada colorido, cuja razão de autovalores é 20, são exibidos nas Figuras 5.23, 5.24, e 5.25, respectivamente.

Tabela 5.3: Valores de τ_o considerando $L = 1$.

resultado \ entrada	branca	$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$	$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$
experimental	0,80	0,90	0,80
teórico	0,85	0,75	0,75

Tabela 5.4: Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 1$).

$\bar{\gamma} \setminus$ entrada	branca		$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$		$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$	
	(exp.)	(teo.)	(exp.)	(teo.)	(exp.)	(teo.)
$\sqrt{1\sigma_n^2}$	0,7845	0,7273	1,2780	0,8986	1,3940	1,0500
$\sqrt{2\sigma_n^2}$	1,0170	0,9220	1,5630	1,1480	1,6810	1,3610
$\sqrt{4\sigma_n^2}$	1,6960	1,6760	2,4240	1,9680	2,5930	2,2770
$\sqrt{5\sigma_n^2}$	2,0770	2,2230	2,8570	2,5130	3,0510	2,8470

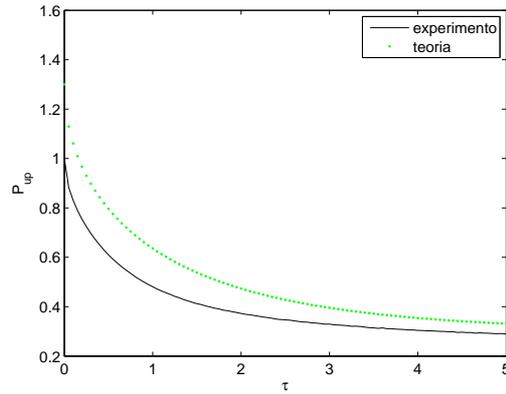


Figura 5.20: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).

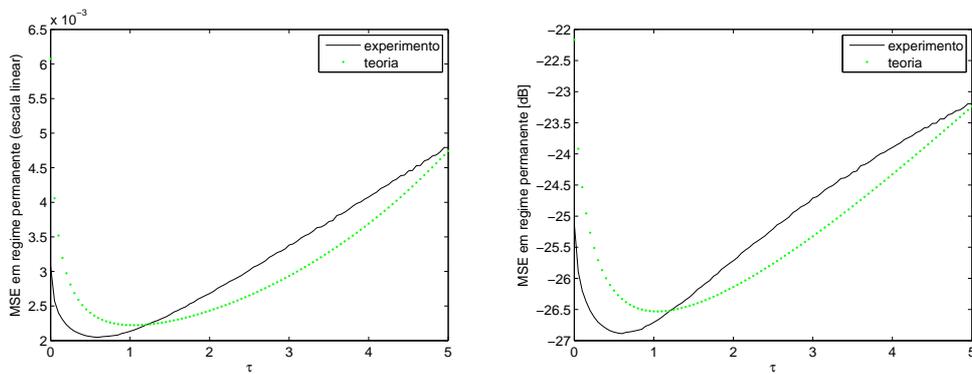


Figura 5.21: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).

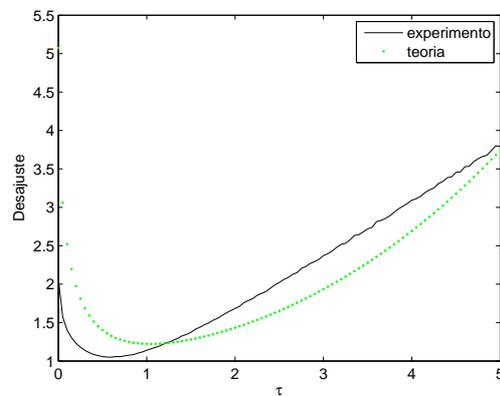


Figura 5.22: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 2$).

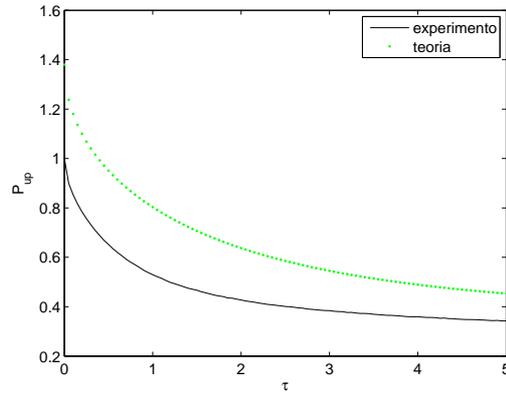


Figura 5.23: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).

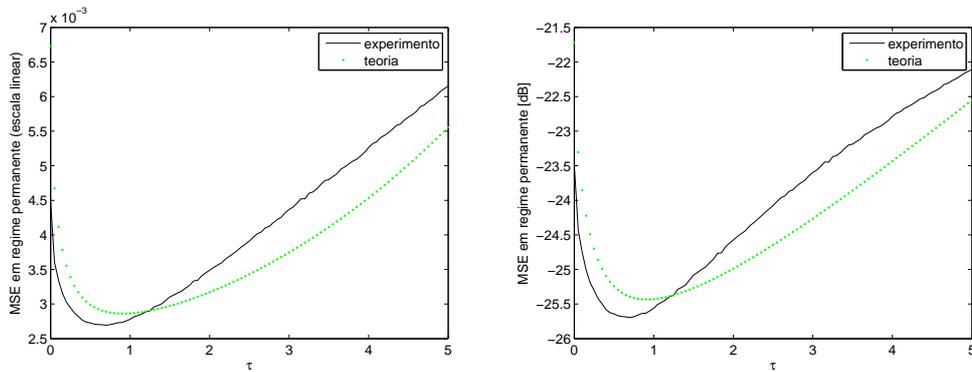


Figura 5.24: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).

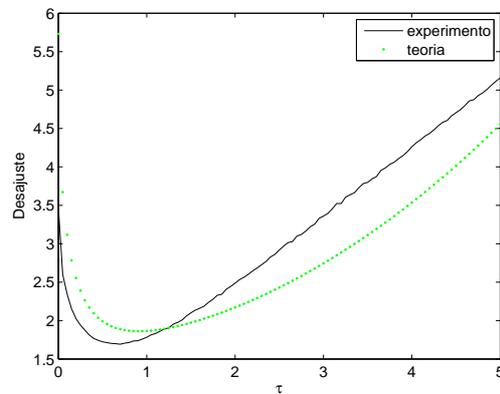


Figura 5.25: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores 20 ($L = 2$).

As Figuras 5.26, 5.27, e 5.28 ilustram, respectivamente, o P_{up} , MSE em regime permanente, e desajuste do algoritmo considerando um sinal de entrada colorido, cuja razão de autovalores é 80.

Assim como na Subseção 5.1.1, os resultados obtidos nesta subseção estão con-
dizentes com a prática. Além disso, pode-se notar que o problema de tendência das
curvas de desajuste existente para o SM-NLMS também não existe para o caso em
que $L = 2$. A Tabela 5.5 resume os valores de τ_o para os três tipos de sinais de
entrada. Novamente, percebemos que os valores teóricos estão bem próximos dos
práticos.

Por fim, a Tabela 5.6 exhibe alguns resultados de desajuste para quatro escolhas
particulares de $\bar{\gamma}$. Pode-se perceber que a escolha usual $\sqrt{5\sigma_n^2}$, novamente, não é a

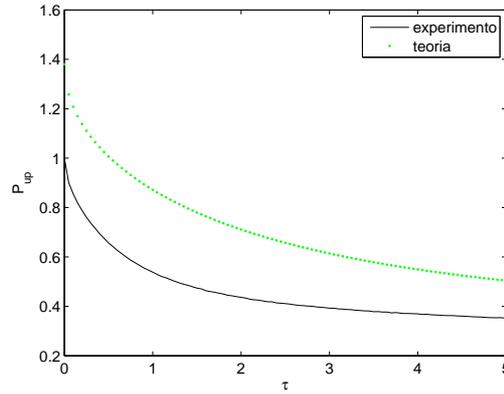


Figura 5.26: Percentual de atualização do filtro adaptativo em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 80 ($L = 2$).

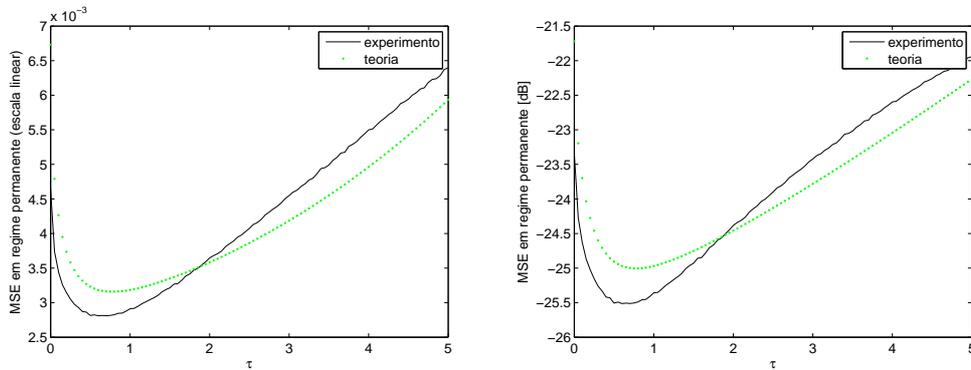


Figura 5.27: MSE após convergência em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 80 ($L = 2$).

melhor, no que diz respeito a gerar um desajuste menor para o SM-AP com $L = 2$.

5.1.4 $L = 4$

Com a finalidade de ilustrar que o problema de tendência da curva teórica de desajuste é um problema particular do caso em que $L = 0$, resolvemos mostrar um gráfico de desajuste para $L = 4$ e entrada branca, vide Figura 5.29.

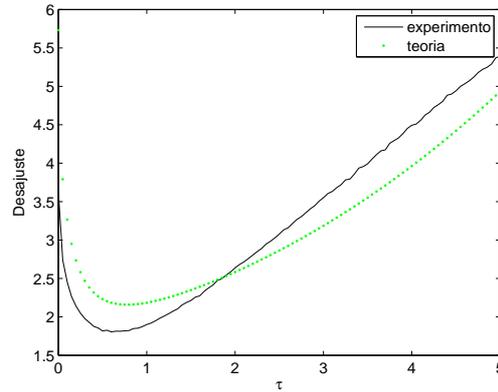


Figura 5.28: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2} \times \tau$ – entrada com razão de autovalores 80 ($L = 2$).

Tabela 5.5: Valores de τ_o considerando $L = 2$.

resultado \ entrada	branca	$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$	$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$
	experimental	0,60	0,70
teórico	1,00	0,90	0,75

Tabela 5.6: Valores de desajuste para diferentes valores de $\bar{\gamma}$ ($L = 2$).

$\bar{\gamma} \setminus$ entrada	branca		$\frac{\lambda_{\max}}{\lambda_{\min}} = 20$		$\frac{\lambda_{\max}}{\lambda_{\min}} = 80$	
	(exp.)	(teo.)	(exp.)	(teo.)	(exp.)	(teo.)
$\sqrt{1\sigma_n^2}$	1,140	1,223	1,782	1,865	1,900	2,184
$\sqrt{2\sigma_n^2}$	1,684	1,433	2,486	2,171	2,638	2,583
$\sqrt{4\sigma_n^2}$	3,092	2,692	4,262	3,536	4,493	3,963
$\sqrt{5\sigma_n^2}$	3,789	3,743	5,158	4,555	5,393	4,934

5.2 Cenário 2

Nesta seção, consideraremos o mesmo cenário descrito na Seção 5.1, porém com a variância do ruído igual a 10^{-2} . O nosso objetivo aqui é verificar o quanto nossas aproximações para o desajuste são sensíveis à uma piora na razão sinal-ruído. Escolheu-se exibir, na Figura 5.30, o resultado do desajuste do SM-AP com $L = 1$ e entrada com razão de autovalores igual a 80.

Comparando as Figuras 5.30 (considera $\sigma_n^2 = 10^{-2}$) e 5.19 (considera $\sigma_n^2 = 10^{-3}$) nota-se que as curvas geradas em ambos os casos são praticamente idênticas.

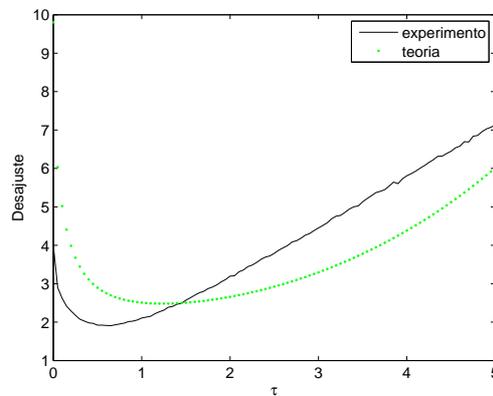


Figura 5.29: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada branca ($L = 4$).

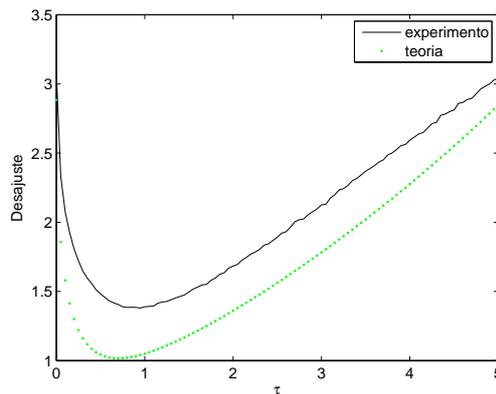


Figura 5.30: Desajuste em função de τ , onde τ é a constante que define $\bar{\gamma} = \sqrt{\sigma_n^2 \times \tau}$ – entrada com razão de autovalores igual a 80 ($L = 1$) e $\sigma_n^2 = 10^{-2}$.

5.3 Resumo dos resultados

Neste capítulo mostramos que os resultados obtidos a partir da análise do algoritmo fornecem uma boa aproximação do comportamento real de MSE em regime permanente do SM-AP. Além disso, mostrou-se que a escolha usual $\tau = 5$ em $\bar{\gamma} = \sqrt{\tau\sigma_n^2}$ nem sempre é boa (em termos de gerar um desajuste pequeno), verificando que conforme L aumenta, mais próximo τ_o estará de 0. Mais do que isso, mostramos que o desajuste do SM-AP é uma curva convexa em τ e, consequentemente, podemos utilizar técnicas de otimização convexa para descobrir o τ_o , isto é, podemos otimizar a curva teórica do desajuste e, com isso, obter uma boa aproximação do τ_o prático. Essa observação não vale para as curvas teóricas em que $L = 0$ devido a uma diferença no formato dessas curvas. Contudo, para um intervalo de $\tau \in [0, 5]$ (que é o intervalo de escolhas usuais para τ), os desajustes teórico e prático são muito próximos.

Mostramos também que os resultados obtidos em [15] não representam uma boa aproximação para o SM-NLMS, uma vez que a sua expressão de MSE após convergência é monotonicamente decrescente com $\bar{\gamma}$ e decai até σ_n^2 , ou seja, o EMSE e desajuste obtidos em [15] tendem a 0 conforme $\bar{\gamma}$ cresce. Mostramos, nos experimentos realizados para o SM-NLMS, que isso não é verdade.

Capítulo 6

Conclusões

Nesta dissertação foi realizada, pela primeira vez na literatura, a análise de MSE em regime permanente do algoritmo SM-AP.

A abordagem utilizada nesta apresentação foi a de, primeiramente, apresentar o algoritmo SM-AP como uma espécie de generalização de outros algoritmos e ressaltar duas de suas características mais importantes: seleção de dados e reuso de dados. Em seguida, discutiu-se brevemente sobre os métodos de análise de algoritmos adaptativos enfatizando os pontos fortes e fracos de cada técnica. Com isso, a escolha pela utilização do método de conservação de energia foi justificada.

Em seguida, realizou-se a análise do algoritmo SM-AP, baseada no método de conservação de energia, e derivou-se expressões para o excesso de MSE e o desajuste desse algoritmo. É importante ressaltar que em momento algum de nossa análise escolhemos uma distribuição para o vetor de entrada.

Por fim, apresentam-se alguns resultados de MSE em regime permanente para o algoritmo SM-AP. Verificou-se experimentalmente que as expressões teóricas derivadas fornecem uma boa aproximação para o comportamento real desse algoritmo após convergência. Atenção especial foi dada aos casos em que $L = 0$ e $L = 1$, pois eles recaem nos algoritmos SM-NLMS e SM-BNLMS, respectivamente. Cabe ressaltar que para o algoritmo SM-NLMS já existe uma análise de MSE em regime permanente [15], porém ela apresenta inconsistência com a prática, uma vez que o seu MSE é monotonicamente decrescente com $\bar{\gamma}$, enquanto que, na prática, verificou-se que existe uma região de valores de τ na qual o MSE é monotonicamente decrescente com τ e uma outra região na qual o MSE é monotonicamente crescente com τ .

Além disso, deve-se destacar que as expressões propostas para EMSE e desajuste foram testadas em vários outros cenários e sempre proporcionaram boas estimativas do EMSE e desajuste reais do algoritmo. Portanto, podemos concluir que as aproximações e eurísticas propostas na análise não são particulares para o cenário considerado nesta dissertação.

Capítulo 7

Trabalhos Futuros

Segue-se alguns trabalhos que ficaram em aberto e podem ser realizados em futuras investigações:

- calcular o EMSE de forma exata, utilizando algum método numérico para resolver a equação (4.67), na variável $E[\tilde{e}_0^2(k)]$;
- verificar o problema de tendência da curva de desajuste para $L = 0$;
- analisar outras características do SM-AP, tais como velocidade de convergência, comportamento em um ambiente não-estacionário, estabilidade, dentre outras.

Referências Bibliográficas

- [1] DINIZ, P. S. R., DA SILVA, E. A. B., NETTO, S. L., *Digital Signal Processing: Systems Analysis and Design*. Cambridge, UK, Cambridge University Press, 2002.
- [2] ANTONIOU, A., *Digital Signal Processing: Signals, Systems, and Filters*. New York, NY, McGraw Hill, 2006.
- [3] PROAKIS, J. G., MANOLAKIS, D. G., *Digital Signal Processing*. 4 ed. Englewood Cliffs, NJ, Prentice Hall, 2007.
- [4] DINIZ, P. S. R., *Adaptive Filtering: Algorithms and Practical Implementation*. 3 ed. New York, NY, Springer, 2008.
- [5] BISPO, B. C., *Cancelamento de eco elétrico em redes telefônicas e acústico em sistemas de teleconferência*, Master's Thesis, COPPE/UFRJ, 2008.
- [6] SAYED, A. H., *Fundamentals of Adaptive Filtering*. Hoboken, NJ, John Wiley & Sons, 2003.
- [7] HAYKIN, S., *Adaptive Filter Theory*. 4 ed. Englewood Cliffs, NJ, Prentice Hall, 2002.
- [8] DINIZ, P. S. R., BRAGA, R. P., WERNER, S., “Set-membership affine projection algorithm for echo cancellation”, *Proc. IEEE International Symp. on Circuits and Systems*, pp. 405–408, 2006.
- [9] DE LAMARE, R. C., DINIZ, P. S. R., “Set-membership adaptive algorithms based on time-varying error bounds for CDMA interference suppression”, *IEEE Trans. Vehicular Technology*, v. 58, pp. 644–654, 2009.

- [10] DINIZ, P. S. R., LIMA, M. V. S., MARTINS, W. A., “Semi-blind data-selective algorithms for channel equalization”, *Proc. IEEE International Symp. on Circuits and Systems*, pp. 53–56, 2008.
- [11] MARTINS, W. A., LIMA, M. V. S., DINIZ, P. S. R., “Semi-blind data-selective equalizers for QAM”, *Proc. IEEE Signal Processing Advances for Wireless Communications*, pp. 501–505, 2008.
- [12] CARINI, A., SICURANZA, G. L., “Analysis of a multichannel filtered-x set-membership affine projection algorithm”, *Proc. IEEE International Conf. on Acoustic, Speech, and Signal Processing*, v. 3, pp. 193–196, 2006.
- [13] WERNER, S., DINIZ, P. S. R., “Set-membership affine projection algorithm”, *IEEE Signal Processing Lett.*, v. 8, pp. 231–235, 2001.
- [14] DINIZ, P. S. R., WERNER, S., “Set-membership binormalized data reusing LMS algorithms”, *IEEE Trans. Signal Processing*, v. 51, pp. 124–134, 2003.
- [15] TAKAHASHI, N., YAMADA, I., “Steady-state mean-square performance analysis of a relaxed set-membership NLMS algorithm by the energy conservation argument”, *IEEE Trans. Signal Processing*, v. 57, pp. 3361–3372, 2009.
- [16] DINIZ, P. S. R., “Analysis of a set-membership affine projection algorithm in nonstationary environment”, *Proc. European Signal Processing Conference*, pp. 1–5, 2009.
- [17] HAYES, M. H., *Statistical Digital Signal Processing*. New York, NY, John Wiley & Sons, 1996.
- [18] ANTONIOU, A., LU, W.-S., *Practical Optimization: Algorithms and Engineering Applications*. New York, NY, Springer, 2007.
- [19] BOYD, S., VANDENBERGHE, L., *Convex Optimization*. Cambridge, UK, Cambridge University Press, 2004.

- [20] LUENBERGER, D. G., *Linear and Nonlinear Programming*. 2 ed. Reading, MA, Addison-Wesley, 1984.
- [21] PAPOULIS, A., *Probability, Random Variables, and Stochastic Processes*. 3 ed. New York, NY, McGraw Hill, 1991.
- [22] STRANG, G., *Linear Algebra and Its Applications*. 3 ed. San Diego, CA, Harcourt Brace Jovanovich, 1988.
- [23] NASCIMENTO, V. H., BERMUDEZ, J. C. M., “Probability of divergence for the least-mean fourth algorithm”, *IEEE Trans. Signal Processing*, v. 54, pp. 1376–1385, 2006.
- [24] WIDROW, B., HOFF, M. E., “Adaptive switching circuits”, *IRE WESCOM Conv. Rec.*, v. 4, pp. 96–104, 1960.
- [25] WIDROW, B., MCCOOL, J. M., LARIMORE, M. G., et al., “Stationary and nonstationary learning characteristics of the LMS adaptive filters”, *Proceedings of the IEEE*, v. 64, pp. 1151–1162, 1976.
- [26] WIDROW, B., STEARNS, S. D., *Adaptive Signal Processing*. Englewood Cliffs, NJ, Prentice Hall, 1985.
- [27] NAGUMO, J. I., NODA, A., “A learning method for system identification”, *IEEE Trans. Automatic Control*, v. AC-12, pp. 282–287, 1967.
- [28] ALBERT, A. E., GARDNER JR., L. S., *Stochastic approximation and nonlinear regression*. Cambridge, MA, MIT Press, 1967.
- [29] BITMEAD, R. R., ANDERSON, B. D. O., “Lyapunov techniques for the exponential stability of linear difference equations with random coefficients”, *IEEE Trans. Automatic Control*, v. AC-25, pp. 782–787, 1980.
- [30] BITMEAD, R. R., ANDERSON, B. D. O., “Performance of adaptive estimation algorithms in dependent random environments”, *IEEE Trans. Automatic Control*, v. AC-25, pp. 788–794, 1980.

- [31] OZEKI, K., UMEDA, T., “An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties”, *Electronics and Communications in Japan*, v. 67-A, pp. 19–27, 1984.
- [32] SANKARAN, S. G., BEECH, A. A. L., “Convergence behavior of affine projection algorithms”, *IEEE Trans. Signal Processing*, v. 48, pp. 1086–1096, 2000.
- [33] GAY, S. L., TAVATHIA, S., “The fast affine projection algorithm”, *Proc. IEEE International Conf. on Acoustic, Speech, and Signal Processing*, v. 5, pp. 3023–3026, 1995.
- [34] CIOFFI, J. M., KAILATH, T., “Windowed fast transversal filters adaptive algorithms with normalization”, *IEEE Trans. Acoustics, Speech, Signal Processing Magazine*, v. 33, pp. 607–625, 1985.
- [35] SHIN, H.-C., SAYED, A. H., SONG, W.-J., “Variable step-size NLMS and affine projection algorithms”, *IEEE Signal Processing Lett.*, v. 11, pp. 132–135, 2004.
- [36] YIN, W., MEHR, A. S., “A variable regularization method for affine projection algorithm”, *IEEE Trans. Circuits and Systems II*, 2009 (to appear).
- [37] GOLLAMUDI, S., NAGARAJ, S., KAPOOR, S., et al., “Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size”, *IEEE Signal Processing Lett.*, v. 5, pp. 111–114, 1998.
- [38] FOGEL, E., HUANG, Y.-F., “On the value of information in system identification - bounded noise case”, *Automatica*, v. 18, pp. 229–238, 1982.
- [39] DELLER, J. R., “Set-membership identification in digital signal processing”, *IEEE Acoust., Speech, Signal Processing Magazine*, v. 6, pp. 4–20, 1989.
- [40] RAO, A. K., HUANG, Y.-H., “Tracking characteristics of an OBE parameter estimation algorithm”, *IEEE Trans. Signal Processing*, v. 33, pp. 1140–1148, 1993.

- [41] NAYERI, M., LIU, M. S., DELLER, J. R., “Do interpretable optimal bounding ellipsoids converge? - Part I and II”, *Proc. 10th IFAC Symp. on System Identification*, v. 3, pp. 389–400, 1994.
- [42] WERNER, S., DE CAMPOS, M. L. R., DINIZ, P. S. R., “Partial-update NLMS algorithm with data-selective updating”, *IEEE Trans. Signal Processing*, v. 52, pp. 938–949, 2004.
- [43] GUO, L., HUANG, Y.-F., “Set-membership adaptive filtering with parameter-dependent error bound tuning”, *Proc. IEEE International Conf. on Acoustic, Speech, and Signal Processing*, v. 4, pp. 369–372, 2005.
- [44] GALDINO, J. F., APOLINÁRIO JR., J. A., DE CAMPOS, M. L. R., “A set-membership NLMS algorithm with time-varying error bound”, *Proc. IEEE International Symp. on Circuits and Systems*, pp. 277–280, 2006.
- [45] WERNER, S., DINIZ, P. S. R., MOREIRA, J. E. W., “Set-membership affine projection algorithm with variable data-reuse factor”, *Proc. IEEE International Symp. on Circuits and Systems*, pp. 261–264, 2006.
- [46] MAZO, J. E., “On the independence theory of equalizer convergence”, *The Bell System Technical Journal*, v. 58, pp. 963–993, 1979.
- [47] LJUNG, L., “Analysis of recursive stochastic algorithms”, *IEEE Trans. Automatic Control*, v. 22, pp. 551–575, 1977.
- [48] LJUNG, L., “On positive real transfer functions and the convergence of some recursive schemes”, *IEEE Trans. Automatic Control*, v. 22, pp. 539–551, 1977.
- [49] SAYED, A. H., RUPP, M., “A time-domain feedback analysis of adaptive algorithms via the small gain theorem”, *Proc. SPIE*, v. 2563, pp. 458–469, 1995.
- [50] SAYED, A. H., RUPP, M., “Error-energy bounds for adaptive gradient algorithms”, *IEEE Trans. Signal Processing*, v. 44, pp. 1982–1989, 1996.

- [51] SHIN, H.-C., SAYED, A. H., “Mean-square performance of a family of affine projection algorithms”, *IEEE Trans. Signal Processing*, v. 52, pp. 90–102, 2004.
- [52] PRICE, R., “A useful theorem for nonlinear devices having Gaussian Inputs”, *IEEE Trans. Information Theory*, v. IT-4, pp. 69–72, 1958.
- [53] JAMES, B. R., *Probabilidade: um curso em nível intermediário*. 3 ed. Rio de Janeiro, RJ, IMPA, 2006.
- [54] DE ALMEIDA, S. J. M., BERMUDEZ, J. C. M., BERSHAD, N. J., “A stochastic model for a pseudo affine projection algorithm”, *IEEE Trans. Signal Processing*, v. 57, pp. 107–118, 2009.
- [55] APOLINÁRIO JR., J. A., DE CAMPOS, M. L. R., DINIZ, P. S. R., “The bi-normalized data-reusing LMS algorithm”, *IEEE Trans. Signal Processing*, v. 48, pp. 3235–3242, 2000.
- [56] YOUSEF, N. R., SAYED, A. H., “A unified approach to the steady-state and tracking analyses of adaptive filters”, *IEEE Trans. Signal Processing*, v. 49, pp. 314–324, 2001.
- [57] GOLLAMUDI, S., NAGARAJ, S., KAPOOR, S., et al., “Set-membership adaptive equalization and updatator-shared implementation for multiple channel communications systems”, *IEEE Trans. Signal Processing*, v. 46, pp. 2372–2384, 1998.
- [58] NAGARAJ, S., GOLLAMUDI, S., KAPOOR, S., et al., “BEACON: An adaptive set-membership filtering technique with sparse updates”, *IEEE Trans. Signal Processing*, v. 47, pp. 2928–2941, 1999.
- [59] SCHWEPPE, F. C., “Recursive state estimate: Unknown but bounded errors and system inputs”, *IEEE Trans. Automatic Control*, v. 13, pp. 22–28, 1968.

Apêndice A

Prova do Teorema 1

Começamos re-escrevendo a equação (4.27), nosso ponto de partida para provar o teorema 1

$$\Delta \mathbf{w}(k+1) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) = \Delta \mathbf{w}(k) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k)$$

Calculando a norma Euclidiana dos dois lados da equação acima obtemos

$$\|\Delta \mathbf{w}(k+1) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k)\|^2 = \|\Delta \mathbf{w}(k) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k)\|^2$$

Expandindo ambos os lados da igualdade

$$\begin{aligned} & \left[\Delta \mathbf{w}(k+1) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \left[\Delta \mathbf{w}(k+1) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right] = \\ & \left[\Delta \mathbf{w}(k) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \left[\Delta \mathbf{w}(k) - \mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right] \end{aligned}$$

Computando os produtos internos, a equação acima torna-se

$$\begin{aligned} \Delta \mathbf{w}^T(k+1)\Delta \mathbf{w}(k+1) - \Delta \mathbf{w}^T(k+1)\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) - \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \Delta \mathbf{w}(k+1) \\ + \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right] = \\ \Delta \mathbf{w}^T(k)\Delta \mathbf{w}(k) - \Delta \mathbf{w}^T(k)\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) - \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \Delta \mathbf{w}(k) \\ + \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right]^T \left[\mathbf{X}(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \right] \end{aligned}$$

Como $\tilde{\mathbf{e}}(k) = -\mathbf{X}^T(k)\Delta \mathbf{w}(k+1)$ e $\tilde{\mathbf{e}}(k) = -\mathbf{X}^T(k)\Delta \mathbf{w}(k)$

$$\begin{aligned} \|\Delta \mathbf{w}(k+1)\|^2 + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) = \\ \|\Delta \mathbf{w}(k)\|^2 + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) + \tilde{\mathbf{e}}^T(k)\hat{\mathbf{R}}^{-1}(k)\tilde{\mathbf{e}}(k) \end{aligned}$$

onde utilizamos o fato de que a inversa de uma matrix simétrica é também simétrica.

Removendo os termos iguais dos dois lados da última equação, a seguinte igualdade é válida

$$\|\Delta \mathbf{w}(k+1)\|^2 + \tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k) = \|\Delta \mathbf{w}(k)\|^2 + \tilde{\mathbf{e}}^T(k) \hat{\mathbf{R}}^{-1}(k) \tilde{\mathbf{e}}(k)$$

e a prova do teorema 1 está concluída. Repare que ao longo da demonstração nenhuma aproximação foi utilizada.