



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

**UMA METODOLOGIA DE AVALIAÇÃO E PROJETO DE
ALGORITMOS ADAPTATIVOS DE PASSO VARIÁVEL**

CÁSSIO GUIMARÃES LOPES

Florianópolis, Novembro de 1999



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

UMA METODOLOGIA DE AVALIAÇÃO E PROJETO DE
ALGORITMOS ADAPTATIVOS DE PASSO VARIÁVEL

Dissertação submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Mestre em Engenharia Elétrica

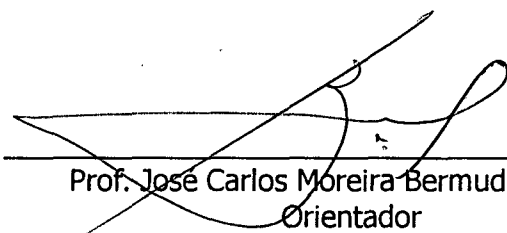
CÁSSIO GUIMARÃES LOPES

Florianópolis, Novembro de 1999


UMA METODOLOGIA DE AVALIAÇÃO E PROJETO DE ALGORITMOS ADAPTATIVOS DE PASSO VARIÁVEL

Cássio Guimarães Lopes

' Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Engenharia Elétrica, Área de Concentração em Processamento de Sinais, e aprovada em sua forma final pelo Curso de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.'

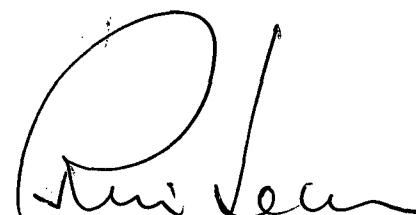


Prof. José Carlos Moreira Bermudez, Ph. D.
Orientador

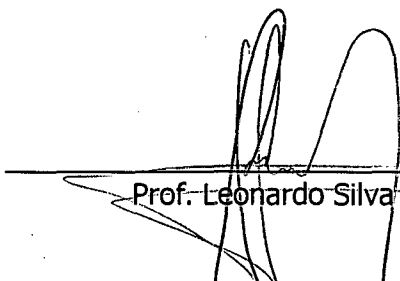


Prof. Idemar Cassana Decker, D. Sc.
Coordenador do Curso de Pós-Graduação em Engenharia Elétrica

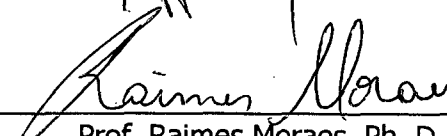
Banca Examinadora:



Prof. Rui Seara, Dr. Ing.



Prof. Leonardo Silva Resende, Dr.



Prof. Raimes Moraes, Ph. D.

*Aos Meus Avós, José e Meta, e ao Meu Tio, Renato. Uma Homenagem Póstuma.
Aos Meus Avós, Ademar e Délia. Exemplos de vida.
Aos Meus Pais, Paulo e Maria Elena. A gênese de tudo.*

Agradecimentos

Aos meus pais, Paulo e Maria Elena, e às minhas irmãs, Ana Paula e Karina, pelo apoio e carinho incondicionais, nem sempre reconhecidos à altura.

Ao amigo-irmão Humberto, pela amizade onipresente, por toda a ajuda e, sobretudo, pela paciência de Jó.

Ao Prof. Bermudez, amigo e orientador, pela confiança depositada e pela ímpar oportunidade de docência ainda nos tempos de formação. Onde havia um estudante, algo mais hoje há.

À CAPES, tímida esperança de independência brasileira, pelo suporte financeiro.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Mestre em Engenharia Elétrica

UMA METODOLOGIA DE AVALIAÇÃO E PROJETO DE ALGORITMOS ADAPTATIVOS DE PASSO VARIÁVEL

Cássio Guimarães Lopes

Novembro/1999

Orientador: José Carlos Moreira Bermudez, PhD

Área de Concentração: Processamento de Sinais

Palavras-chave: Filtros adaptativos, Algoritmos de Passo Variável, Processamento Digital de Sinais

Número de Páginas: 132

A operação do algoritmo LMS com passo variável pode oferecer vantagens em relação à operação com passo fixo. A adaptação do passo pode acelerar significativamente a dinâmica do algoritmo em ambientes estacionários. Na presença de não-estacionaridades, pode aumentar a capacidade de rastreamento da solução ótima. Utilizando indicativos absolutos de desempenho da convergência com passo variável e uma caracterização mais eficiente do processo adaptativo, derivou-se uma metodologia de avaliação e projeto de uma ampla classe dos inúmeros algoritmos propostos na literatura. Vários exemplos demonstram a aplicação da metodologia na avaliação e projeto de algoritmos existentes, ilustrando sua eficiência quando comparada a outras metodologias, mesmo em ambientes sujeitos a certos tipos de não-estacionaridades. Revela-se a potencialidade das estruturas de controle do passo baseadas na filtragem de alguma variável do processo adaptativo. Quando projetadas pela metodologia proposta, apresentam desempenho superior a algoritmos mais complexos. Em decorrência da análise do caso com passo variável, derivou-se uma nova expressão para o passo de adaptação que maximiza a velocidade de convergência em implementações com passo fixo. Comparado aos resultados clássicos, o passo calculado pela nova expressão acelera a convergência e fornece desajustes menores em regime permanente.

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

AN ASSESSMENT AND DESIGN METHODOLOGY OF VARIABLE STEP-SIZE ADAPTIVE ALGORITHMS

Cássio Guimarães Lopes

November/1999

Advisor: José Carlos Moreira Bermudez, PhD

Area of Concentration: Signal Processing

Keywords: Adaptive Filtering, Variable Step-Size Algorithms, Digital Signal Processing

Number of Pages: 132

The operation of the LMS algorithm with variable step-size can offer advantages as compared to the fixed step-size operation. The step-size adaptation may significantly accelerate the algorithm dynamics in stationary environments. In the presence of nonstationarities, it is possible to improve the tracking capabilities of the algorithm. Using absolute convergence performance indicators and a more efficient characterization of the adaptive process, an assessment and design methodology is developed, which can be applied to a broad class of algorithms proposed in literature. Several examples in different circumstances demonstrate the applicability of the methodology for assessing and designing the existing algorithms. The new methodology is efficient as compared to existing ones, even in certain nonstationary environments. It is shown that if well designed, very simple algorithms (based upon the filtering of some internal variable) can outperform much more complex algorithms. Originated from the variable step-size analysis, a new expression for the adaptation step-size is derived to maximize the convergence ratio for fixed step-size implementations. Compared to classic results, the new step-size provides faster convergence with less steady-state misadjustment.

Sumário

Capítulo 1 – Introdução Geral

1.1.	Contexto	1
1.2.	Estratégias Determinísticas.....	8
1.3.	Estratégias Baseadas em Estruturas de Filtragem	9
1.4.	Estratégias de Busca Automática	11
1.5.	Estratégias Direcionais	13
1.6.	Lógica Difusa no Controle do Passo	15
1.7.	Avaliação e Projeto de Algoritmos	15
1.8.	Estrutura do Trabalho	17

Capítulo 2 – O Algoritmo LMS

2.1.	Introdução	19
2.2.	Filtragem Ótima de Wiener	20
2.3.	A Solução Adaptativa	25
2.4.	O Algoritmo LMS com Passo Fixo	27
2.5.	Estabilidade e Desempenho do Algoritmo LMS com Passo Fixo	29
2.6.	O Algoritmo LMS Operando com Passo Variável	34
2.7.	Algumas Estratégias de Adaptação do Passo	36
2.8.	Conclusão	38

Capítulo 3 - Metodologia de Avaliação e Projeto de Algoritmos Adaptativos de Passo Variável

3.1.	Introdução	40
3.2.	O Modelo Analítico do Algoritmo LMS com Passo Fixo	41
3.3.	Evolução Ótima do Algoritmo LMS	45
3.4.	Caracterização Paramétrica de Algoritmos de Passo Variável	49
3.5.	O Plano de Aprendizado	51
3.6.	Geração e Utilização do Plano de Aprendizado.....	54

3.6.1.	A Trajetória Ótima	54
3.6.2.	A Curva de Estabilização	55
3.6.3.	Geração de Trajetórias de Algoritmos de Passo Variável no Plano de Aprendizado	59
3.6.4.	A Região de Estabilização e Heurísticas para Avaliação de Trajetórias no Plano de Aprendizado	61
3.7.	Conclusão	63

Capítulo 4 – Aplicações da Metodologia Desenvolvida

4.1.	Introdução	64
4.2.	Projeto de Algoritmos de Passo Variável para Identificação de Sistemas	65
4.2.1.	Passo Fixo Ótimo LMS – Um Novo Resultado	69
4.2.2.	Um Algoritmo Simples e Projetável: VSS	75
4.2.3.	Comparando Algoritmos: MVSS Versus VSS	81
4.3.	Projeto da Dinâmica dos Algoritmos ou Busca Automática ?	95
4.4.	Um Algoritmo Determinístico para Controle do Passo	100
4.5.	Caso Não-Estacionário	102
4.5.1.	Algoritmo NASS versus Algoritmo VSS	103
4.6.	Conclusão	106

Capítulo 5 – Conclusões, Críticas, Conjecturas e Sugestões para Trabalhos Futuros

5.1.	Conclusões	108
5.2.	Críticas	113
5.3.	Algumas Conjecturas (ou “Quase Conclusões”)	115
5.4.	Sugestões para Trabalhos Futuros	117

Referências Bibliográficas	120
---	------------

Apêndice – Modelo Analítico Recursivo da Matriz $K(n)$ com Passo Variável	124
---	------------

Capítulo 1

Introdução Geral

"Para entender uma ciência é necessário conhecer sua história"

Auguste Comte

1.1.Contexto

Em diversas áreas da engenharia e das ciências aplicadas, freqüentemente é necessário estimar sinais que só podem ser observados na presença de ruído aditivo. Este problema reflete-se em inúmeras aplicações, como exploração geofísica, radar, sonar, cancelamento de ecos, controle de ruído acústico e vibrações estruturais, astrofísica, sismologia, processamento de sinais biomédicos, identificação e modelamento de sistemas físicos, entre outras [1].

Quando se tem acesso a um sinal de referência, pode-se estimar o sinal desejado através de uma estrutura de filtragem apropriada, apresentando ao filtro estimador amostras do sinal de referência, que deve ser correlacionado com o sinal desejado. O filtro processa o sinal de referência, fornecendo em sua saída uma estimativa do sinal desejado.

Inerente à aproximação oferecida pelo filtro há um erro de estimação, definido como a diferença entre o sinal desejado e a resposta do filtro ao sinal de referência. O erro de estimação representa uma medida do desempenho alcançado pelo filtro na aproximação do sinal desejado, devendo ser minimizado. Uma vez que, *a priori*, não se tem controle sobre os sinais envolvidos, o procedimento natural é projetar o filtro de maneira a minimizar o erro de estimação. A Figura 1.1 ilustra a situação.

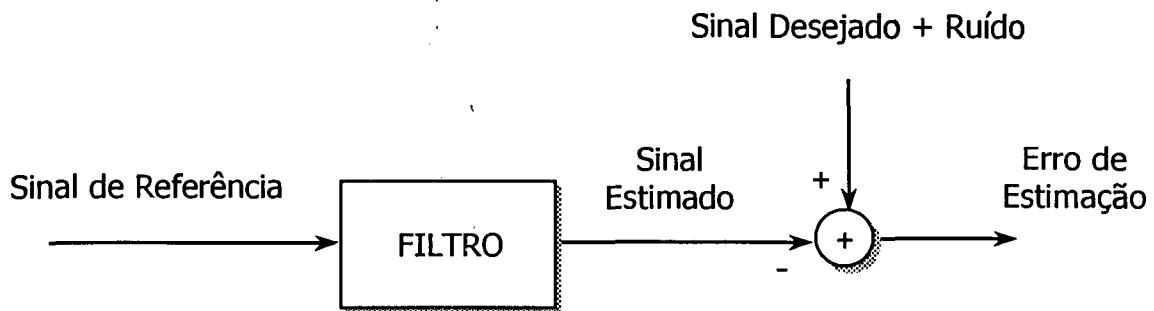


Fig. 1.1 – O Problema da Estimação de Sinais

Na abordagem clássica do problema, os sinais, em princípio de natureza desconhecida, são assumidos como realizações de processos estocásticos. O erro é então minimizado em algum sentido estatístico, considerando a estrutura de filtragem escolhida e o critério de otimização empregado.

Visando a *tratabilidade matemática* e outras considerações de ordem prática, as estruturas lineares de filtragem são normalmente preferidas na implementação do filtro estimador, e o critério de otimização empregado é a minimização do erro de estimação no sentido médio quadrático [1].

Os primeiros estudos utilizando estimação linear no sentido médio quadrático em processos estocásticos foram feitos por Kolmogorov, Krein e Wiener, durante as décadas de 30 e 40 [1], e encontram parentesco nos trabalhos do matemático alemão Cari Friederich Gauss, que criou em 1795, aos 18 anos, o método dos mínimos quadrados para estudar o movimento de corpos celestes. Os trabalhos de Kolmogorov e Krein eram independentes do trabalho de Wiener, e embora houvesse alguma similaridade nos resultados, suas abordagens eram consideravelmente diferentes.

Wiener, entre outras contribuições, abordou o problema da estimação de um processo estocástico corrompido por outro processo aditivo "ruidoso", derivando uma fórmula explícita para a estimativa ótima. Solucionando uma equação integral, conhecida como equação de *Wiener-Hopf*, era possível determinar a resposta ao impulso do filtro estimador linear ótimo [1], [2].

Trabalhando com processos estocásticos discretos, Levinson formulou

em 1947 o problema de Wiener para tempo discreto, utilizando uma estrutura de filtragem linear transversal, na qual o sinal de interesse é estimado através da combinação linear de observações do sinal de referência. O propósito era ajustar os coeficientes do filtro transversal de maneira a produzir a melhor estimativa possível do processo de interesse, dada alguma medida de desempenho. Seguindo a formulação original de Wiener, a qual herdou o critério de Gauss, projetava-se os coeficientes do filtro para minimizar o erro de estimação no sentido médio quadrático.

Com as premissas de linearidade e resposta ao impulso de duração finita, intrínsecas à estrutura transversal, determina-se uma *superfície de desempenho parabólica*, traduzindo o comportamento do erro médio quadrático em função dos coeficientes do filtro. Essa geometria é bastante conveniente e define unicamente a solução ótima, cujo vetor de coeficientes associado é conhecido na literatura como *filtro de Wiener* [1], [3]. No caso discreto, com os dados estendendo-se a um passado finito, a fórmula integral original do caso contínuo assume uma forma matricial, determinando uma expressão explícita para o vetor de coeficientes do filtro estimador ótimo [1], [3], [4], [5]:

$$RW_0 = P$$

onde W_0 é o vetor de coeficientes do filtro de Wiener, R é a matriz de autocorrelação do sinal de referência, e P é o vetor de correlação cruzada entre o sinal de referência do filtro e o sinal desejado. Se a matriz R é não-singular, a solução ótima para o filtro estimador é:

$$W_0 = R^{-1}P$$

Constituindo um dos resultados mais simples e elegantes em processamento de sinais discretos, a solução de Wiener traça os objetivos da estrutura transversal linear para a operação ótima. Contudo, a aparente simplicidade esconde alguns problemas. A sua utilização depende de informações geralmente não conhecidas *a priori*, além de envolver uma complexidade

computacional normalmente proporcional ao cubo da ordem do filtro estimador empregado (N^3) [6], o que pode acarretar problemas para aplicações em tempo real, nas quais o tempo de processamento é limitado. Quando os dados são estacionários e procedentes de uma *linha de retardo*, por exemplo, a matriz R é *Toeplitz* [4], e a complexidade computacional pode então ser amenizada para a ordem N^2 [6].

Além da complexidade computacional inerente, a utilização da expressão de Wiener também encontra dificuldades nas aplicações que envolvem sinais não-estacionários, onde a variação temporal das propriedades estatísticas torna problemática a estimação das matrizes R e P .

Um outro aspecto negativo, remanescente dos aspectos computacionais, pode manifestar-se em situações nas quais a matriz R apresenta um mau condicionamento numérico, gerando problemas para realizar a sua inversão, presente na expressão de Wiener.

As dificuldades de utilização da solução de Wiener na sua forma matricial foram contornadas com o surgimento das *estratégias adaptativas* - em torno do filtro é estruturado um *algoritmo de aprendizado* que extrai informações disponíveis dos dados e adapta recursivamente um conjunto inicial e arbitrário de coeficientes, visando a solução de Wiener. O filtro estimador é treinado simultaneamente à operação de filtragem num contexto de aprendizado, constituindo um processo de otimização iterativo.

Uma vez que a otimização é feita iterativamente, os algoritmos de aprendizado são costumeiramente avaliados pela evolução do erro médio quadrático em função do número de iterações, denominada *curva de aprendizado*. A curva de aprendizado revela o quão rápido um algoritmo pode adaptar-se a um ambiente, representando a evolução do aprendizado do filtro.

Com a convergência, é necessário avaliar a qualidade da solução oferecida pelo algoritmo adaptativo. Um critério bastante utilizado então é o *desajuste*, definido como a razão entre o erro médio quadrático em excesso e o erro médio quadrático mínimo, dado pela solução de Wiener. O erro em excesso,

por sua vez, é definido como a diferença entre o erro médio quadrático atingido pelo algoritmo em regime permanente e o erro médio quadrático mínimo.

As estruturas de filtragem que empregam a topologia adaptativa são denominadas *filtros adaptativos*. Além de computacionalmente mais eficientes, as estruturas adaptativas apresentam algumas características especiais, quando comparadas aos filtros “fixos” [3]:

- *São estruturas auto-projetáveis, podendo extrapolar o espaço de conhecimento e responder a novas situações após um pequeno conjunto de treinamento;*
- *São sistemas robustos, capazes de operar satisfatoriamente mesmo na presença de certos tipos de defeitos ou limitações;*
- *Adaptam-se automaticamente às mudanças do ambiente, oferecendo a possibilidade de rastreamento da solução ótima quando sujeitos a não-estacionaridades.*

O surgimento de sistemas adaptativos conferiu uma nova dimensão à área de processamento de sinais, cativando o interesse de muitos pesquisadores com o advento da era digital.

Um dos primeiros trabalhos abordando filtros adaptativos pode ser reconhecido ainda nos anos 50, quando pesquisadores trabalhavam independentemente em diferentes aplicações para os sistemas adaptativos [1]. Deste pioneirismo, emergiu o algoritmo LMS (“Least-Mean-Square”), um simples mas eficiente algoritmo de aprendizado para operação de filtros adaptativos. Concebido por Widrow e Hoff em 1959 na “Stanford University”, Califórnia, o algoritmo foi originalmente empregado numa estrutura linear de reconhecimento de padrões conhecida como ADALINE (“Adaptive Linear Element”) [1], [3], [8].

O algoritmo LMS foi popularizado principalmente pela simplicidade e robustez, apresentando uma complexidade computacional de ordem N , contrastante com aquelas já comentadas anteriormente. Algumas vezes é

preterido por apresentar uma convergência relativamente lenta, quando confrontado com outros algoritmos adaptativos existentes, tais como o algoritmo RLS ("Recursive Least-Squares") ou o filtro de Kalman [1], [9]. No entanto, é consideravelmente mais simples, encontrando uma gama enorme de aplicações em telecomunicações, teoria de controle e processamento de sinais.

O princípio do algoritmo LMS é simples e assemelha-se ao *método do gradiente-descendente*, introduzido pelo matemático francês Augustin-Louis Cauchy, em 1847 [10]. Partindo de uma estimativa inicial para a solução ótima, o LMS adapta os coeficientes do filtro guiando-se no sentido contrário ao gradiente da superfície de desempenho, estimado instantaneamente dos dados a cada iteração. O gradiente de uma função fornece a direção e sentido de maior crescimento da mesma no ponto em questão. Assim, caminhar no sentido contrário ao gradiente leva o algoritmo LMS a buscar o ponto de gradiente nulo, que é a própria solução de Wiener, dada a unimodalidade da superfície de desempenho. Essa característica leva muitos algoritmos adaptativos a buscarem iterativamente a solução ótima guiados pelo gradiente da função de desempenho [3].

Fruto da estimação instantânea do gradiente, o LMS imprime flutuações estocásticas nos coeficientes do filtro durante o processo adaptativo, motivo pelo qual é também conhecido como *algoritmo do gradiente estocástico* [1], [9]. No sentido médio dos coeficientes, comporta-se de maneira idêntica a sua "versão" determinística, o algoritmo do gradiente-descendente. Todavia, o algoritmo LMS estabiliza-se num patamar de erro médio quadrático *sempre superior* ao valor correspondente à solução de Wiener – é o preço a ser pago pela aproximação instantânea do gradiente.

A evolução do LMS é controlada por uma constante que determina o ajuste incremental a que são submetidos os coeficientes do filtro, conduzindo o algoritmo da estimativa inicial até a solução de regime permanente. O valor desta constante deve ser tal que garanta a estabilidade do algoritmo [6], [11].

A *constante de aprendizado* ou *passo de cálculo*, tal como é conhecido

o parâmetro utilizado no controle da adaptação, determina a velocidade de convergência e as características de regime permanente, influenciando o comportamento global do algoritmo.

Passos de cálculo “grandes” determinam ajustes grandes nos coeficientes do filtro a cada iteração, levando o algoritmo a uma rápida convergência, mas a um erro médio quadrático excessivo em regime, dado que as flutuações estocásticas são proporcionais ao valor do passo empregado (citar a seção onde isto é provado ou uma referência). Por outro lado, passos de cálculo “pequenos” minimizam as flutuações levando a patamares de erro menores em regime permanente, mas sacrificando a velocidade de convergência.

Na literatura, trabalhos visando a otimização do passo de cálculo desdobram-se em duas abordagens distintas [6], [11], [12], [13]. Uma procura maximizar a velocidade de convergência, resultando em passos grandes. Outra procura minimizar o erro médio quadrático em regime permanente, determinando passos pequenos. Esta última abordagem é freqüentemente preferida na prática, desde que após a convergência inicial, que é um fenômeno temporário, deve-se conviver com o erro em excesso no regime permanente [13], [14]. Bershad considerou em [13] a escolha de um passo de cálculo que minimiza o erro médio quadrático ao final de um dado número de iterações.

Infelizmente, a aspiração de todo algoritmo adaptativo - *velocidade de convergência aliada a desajustes pequenos em regime permanente* - não é possível para a formulação clássica do algoritmo LMS com passo de cálculo fixo. A utilização de passo fixo não consegue atender simultaneamente às exigências do regime transitório e do regime permanente no que concerne o erro médio quadrático: *uma boa convergência global envolve uma escolha conflitante para o passo de cálculo.*

Levando em conta que as exigências para o passo ocorrem em períodos diferentes da adaptação, alguns projetistas usualmente utilizam dois passos de cálculo distintos em implementações práticas. Um passo inicial “grande” para rápida convergência, comutando posteriormente para outro passo menor, que

determinará o erro médio quadrático em regime permanente. O ponto crítico é então a escolha do *instante de transição* [14].

Essa estratégia primitiva de variação do passo pode ser generalizada, conferindo ao processo adaptativo uma flexibilidade antes inexistente. O emprego de uma seqüência de passos diferentes a cada iteração diminui o compromisso existente entre velocidade de convergência e erro médio quadrático em regime, além de geralmente tornar o algoritmo LMS mais eficiente quando sujeito a não-estacionaridades [15].

Existem diversas estratégias para gerar uma seqüência de passos que melhore o desempenho do algoritmo do gradiente estocástico. Variando em complexidade e características, as diferentes estratégias procuram basicamente diminuir o compromisso existente entre velocidade de convergência e desajuste em regime permanente, também preocupando-se em alguns casos com a capacidade de rastreamento, necessária para operação do filtro adaptativo em ambientes não-estacionários.

1.2. Estratégias Determinísticas

O algoritmo LMS operando com passo variável está intimamente relacionado à formulação original da *aproximação estocástica*, desenvolvida por Robins e Monro em 1951 na estimação de parâmetros estatísticos [1]. Originalmente, empregava-se passos inversamente proporcionais ao número de iterações n , ou a uma potência qualquer desse número, determinando uma seqüência de passos determinística e sempre decrescente, que evolui independentemente do estado da adaptação.

Seguindo a mesma filosofia, algumas estratégias determinísticas para variação do passo foram propostas e analisadas na literatura [12], [14]. Dirk Slock propõe em [14] uma estratégia de variação do passo também inversamente proporcional às iterações decorridas, necessitando o ajuste de alguns parâmetros de controle. O autor aplicou a estratégia para controlar o algoritmo LMS *normalizado* (NLMS), uma variante do LMS, comparando-o posteriormente com o

algoritmo LMS convencional. Macchi analisou uma técnica muito semelhante em [12].

A estratégia de decrescimento progressivo do passo, implementado nas estruturas descritas, pode aprimorar a convergência do algoritmo LMS, mas é extremamente limitada em ambientes não-estacionários, estagnando a resposta do filtro às possíveis perturbações ocorridas no meio.

1.3. Estratégias Baseadas em Estruturas de Filtragem

Uma estratégia mais eficiente e bastante explorada na literatura consiste em vincular o passo de cálculo a alguma variável do processo adaptativo que reflita o estado do aprendizado, ajustando-se o valor do passo de acordo com as necessidades correntes da adaptação [16], [17], [18], [19]. Além da flexibilidade no caso estacionário, os algoritmos que empregam tais estratégias geralmente tornam o algoritmo LMS mais apto a responder a não-estacionariedades, relativos ao caso com passo fixo ou às estratégias determinísticas, por exemplo.

Mais eficientes e flexíveis, estas estratégias estimam a evolução da figura de mérito utilizada e a mapeiam num valor conveniente para o passo. As figuras de mérito utilizadas são diversas: erro médio quadrático [16], correlação entre o erro e o sinal de referência [18], auto-correlação do erro [17], dentre outras. As grandezas de interesse são estimadas filtrando aproximações instantâneas obtidas diretamente dos dados [16], [17], [18], [19], o que pode ser visto como uma aproximação da média estatística. Entretanto, essa abordagem pressupõe, ainda que implicitamente, *ergodicidade* dos sinais envolvidos, premissa esta incompatível com a natureza não-estacionária do processo adaptativo. Contudo, mesmo diante de uma estimativa possivelmente pouco precisa, a seleção apropriada dos parâmetros de controle intrínsecos às estruturas podem aprimorar significativamente a convergência do algoritmo LMS.

Shan e Kailath [18] propuseram em 1988 um dos primeiros algoritmos de controle do passo para operar o algoritmo LMS. O algoritmo gera uma

seqüência de passos determinada por estimativas do gradiente do erro, obtidas através de uma equação de diferenças de primeira ordem que filtra uma aproximação instantânea do gradiente [18]. A estimativa do gradiente e a seqüência de passos subjacente são reguladas por uma *constante de esquecimento*, cuja escolha determina o comportamento global do algoritmo LMS [18]. Computacionalmente simples, é uma estrutura *independente* da ordem do filtro estimador (N), mas que possui um só grau de liberdade, a constante de esquecimento. Isso pode limitar a flexibilidade da seqüência gerada.

Publicado em 1992, o algoritmo de controle do passo desenvolvido por Kwong e Johnston [16] utiliza uma versão filtrada do erro quadrático instantâneo para adaptar o passo, resultando numa equação de diferenças de primeira ordem. Computacionalmente independente da ordem do filtro N , trata-se de um algoritmo simples, com custo computacional adicional irrisório em relação ao caso com passo fixo. Possui duas constantes de projeto que controlam o comportamento do passo, em conjunto com o erro quadrático. Uma constante é escolhida empiricamente e a outra é ajustada correspondentemente para atingir o valor de erro médio quadrático desejado em regime permanente. Com dois graus de liberdade, um a mais do que o algoritmo proposto em [18], a seqüência gerada possui bastante flexibilidade.

Em algumas aplicações, como identificação de sistemas, a presença de ruído de medição pode contaminar severamente o erro de predição quadrático utilizado por Kwong e Johnston para o controle do passo. Criticando a possível sensibilidade deste algoritmo ao ruído de medição, Aboulnasr e Mayyas mantiveram a estrutura de adaptação original mas introduziram uma nova figura de mérito: o quadrado de uma estimativa da autocorrelação entre $e(n)$ e $e(n-1)$. Segundo os autores, esta figura de mérito reflete melhor o estado da adaptação na presença de *ruído de medição branco* [17]. A autocorrelação do erro é estimada por uma equação de diferenças auxiliar de primeira ordem, que filtra a quantidade instantânea $e(n) \cdot e(n-1)$. Esta estimativa define o critério utilizado pela equação de diferenças principal para adaptar o passo. Simples

computacionalmente, o algoritmo modificado também é independente da ordem N do filtro, possuindo três constantes de projeto a serem ajustadas. Uma delas controla a estimativa da autocorrelação do erro, as outras duas estão presentes na expressão de adaptação do passo, como no algoritmo original proposto em [16].

Seguindo a política de modificações, Okello *et al.* exploraram uma lacuna deixada por Aboulnasr e Mayyas, criando em [19] um algoritmo apresentado como uma melhoria do trabalho original em [17]. O algoritmo proposto é dito ser insensível a ruídos de medição correlacionados - uma limitação do algoritmo original - presentes em aplicações como cancelamento de eco ou teleconferência. O algoritmo estima o gradiente do erro filtrando os dados instantâneos, semelhante ao feito por Shan e Kailath e, embora tenha sido empregado para controlar o passo do algoritmo LMS normalizado, pode facilmente ser estendido para regular o passo do algoritmo LMS "convencional".

As estruturas de filtragem discutidas acima são potencialmente eficientes quando bem projetadas. Independentes computacionalmente da ordem do filtro adaptativo, característica bastante desejável principalmente em aplicações em tempo real, a principal limitação destas estruturas é a sensibilidade ao ambiente onde o filtro adaptativo está inserido, tornando o projeto mais complicado numa situação genérica.

1.4. Estratégias de Busca Automática

Numa tentativa de contornar a sensibilidade de estruturas existentes, algumas estratégias adotam uma abordagem filosoficamente diferente das já citadas. Enquanto a maioria dos algoritmos são baseados em técnicas *ad hoc*, apresentando desempenhos bastante influenciados por seus parâmetros intrínsecos, cuja escolha ótima em geral é dependente dos dados [15], algumas estratégias pretendem a otimização do passo a cada iteração. Para um dado desempenho em regime permanente, elas otimizariam automaticamente o comportamento dinâmico do algoritmo.

Alguns algoritmos de ajuste da constante de aprendizado adotam

o mesmo princípio do algoritmo LMS. Seguindo uma estratégia gradiente-descendente, evoluem a seqüência de passos guiando-se no sentido contrário ao gradiente do erro quadrático relativo ao passo de cálculo, atualizando recursivamente uma estimativa inicial para o mesmo. A cada iteração fornecem, em princípio, um valor conveniente para a constante de aprendizado [1], [21].

Em [1] é explanado um algoritmo que segue a abordagem descrita. Proposto originalmente por Benveniste *et al.* [1] e elaborado em Kushner e Yang [20], constitui um algoritmo relativamente complexo no aspecto computacional, com equações recursivas matriciais dependentes da ordem N do filtro e controladas por um parâmetro de aprendizado que determina a taxa de aprendizado do algoritmo de controle do passo.

Mathews e Xie analisaram mais detalhadamente o algoritmo proposto originalmente por Shin e Lee [21]. O algoritmo também segue a abordagem gradiente-descendente para ajustar o passo, possuindo uma estrutura diferente, computacionalmente mais simples, mas também dependente da ordem N do filtro. Conta com uma constante a ser ajustada, que influencia o comportamento global da seqüência gerada.

Outra estrutura que pode ser enquadrada na busca automática do passo foi sugerida por Koike em [22]. Abrangendo o algoritmo LMS e outros dele derivados, tais como os algoritmos SRA "Signed Regressor Algorithm", SSA "Sign-Sign Algorithm", o algoritmo LMF ("Least Mean Fourth") [23], dentre outros [24], Koike derivou um modelo analítico que descreve a seqüência de passos ótima, propondo um algoritmo de controle que implementa numericamente a expressão analítica, tão próximo quanto possível a cada iteração. A estrutura de controle emprega *acumuladores com perdas*, que nada mais são do que equações recursivas de filtragem utilizadas para estimar algumas quantidades necessárias à operação do algoritmo proposto. Relativamente complexo, o algoritmo é dependente da ordem N do filtro, envolvendo três equações matriciais de adaptação para o caso geral. Apresenta *divisão* numa das equações e necessita o ajuste de duas constantes que regulam a evolução do passo.

Embora pretendam o ajuste automático da constante de adaptação, de uma certa forma independentemente da natureza dos dados, os algoritmos de busca automática paradoxalmente também possuem constantes a serem ajustadas. Estas constantes influenciam o comportamento da seqüência de passos gerada, embora possivelmente de maneira menos significativa do que as constantes empregadas nas estratégias de filtragem.

1.5. Estratégias Direcionais

No algoritmo LMS, a estrutura da matriz R tem profundo impacto sobre seu comportamento. Quando o filtro adaptativo é submetido a entradas correlacionadas, a dispersão dos autovalores da matriz R correspondente influencia substancialmente o comportamento dinâmico do algoritmo LMS. Nestes ambientes, o vetor gradiente do erro não aponta, em geral, na direção da solução ótima, o que leva o algoritmo LMS a experimentar uma convergência de natureza direcional, descrevendo um caminho pelo domínio da superfície de desempenho que depende da estrutura da matriz R , da solução de Wiener, da inicialização e da seqüência de passos adotada [1], [14].

A *direcionalidade da convergência* do algoritmo LMS sob entradas correlacionadas é a explicação natural para a sua convergência relativamente lenta. Entretanto, a velocidade de convergência do LMS é maior em algumas direções e menor em outras. Dependendo da inicialização, a adaptação pode inclusive ser acelerada pela dispersão dos autovalores de R [1], [14]. Note-se que uma escolha displicente da seqüência de passos nessas circunstâncias poderia deteriorar significativamente a convergência do LMS.

Reconhecendo a importância da direcionalidade do algoritmo LMS para sinais de referência correlacionados, algumas estruturas empregam o *ajuste individual do passo para cada coeficiente do filtro adaptativo*. A utilização de passos diferentes para cada dimensão do filtro permitiria, teoricamente, ao algoritmo adaptativo aproveitar as direções de convergência mais rápida, dadas pelos autovetores com maiores autovalores associados.

Um dos trabalhos pioneiros nesse sentido foi proposto em 1986 por Harris *et al.* [8], empregando o ajuste individual do passo para cada coeficiente do filtro. Observando o comportamento do *sinal* do gradiente do erro quadrático instantâneo, cuja alternância serve como indicativo aproximado de convergência, o algoritmo de controle reduz ou aumenta o passo de acordo com duas constantes de controle, utilizadas para contar o número mudanças consecutivas e o número de vezes que o sinal do gradiente instantâneo permanece inalterado. Quando o gradiente em determinada direção alterna m_0 vezes consecutivas seu sinal, o algoritmo reduz o passo nessa direção dividindo-o por um fator constante $\alpha > 1$. Caso o gradiente permaneça por m_1 iterações consecutivas com seu sinal inalterado, o algoritmo aumenta o passo nessa direção, multiplicando-o pelo mesmo fator α .

É uma estrutura complexa, dependente da ordem N do filtro e necessitando a observação individual de cada componente do gradiente, com o emprego de dois contadores para cada dimensão do filtro adaptativo, comparados a cada iteração com as constantes de controle m_0 e m_1 para ajustar individualmente os passos.

Evans *et al.* [15] aprimoraram a análise do algoritmo proposto previamente por Harris *et al.* [8], descrevendo a evolução média e média quadrática do vetor de erro nos coeficientes do filtro, que são utilizados para calcular a evolução do erro médio quadrático.

Em [21], Mathews *et al.* também comentam a possibilidade de extensão do algoritmo de busca automática por eles estudado para ajustar individualmente o passo de cálculo para cada dimensão do filtro, mencionando uma ligeira superiodade frente a outros algoritmos com a estratégia direcional, o algoritmo proposto por Harris *et al.* inclusive, para aplicações multicanais e não-lineares de filtros adaptativos.

Justificáveis, em princípio, quando o algoritmo adaptativo opera com sinais de referência correlacionados, nos casos em que se tem ruído branco como sinal de referência (nas aplicações de identificação de sistemas, por exemplo) o

algoritmo LMS apresenta uma convergência não-direcional, com o gradiente apontando sempre na direção da solução ótima - a utilização de estruturas direcionais ficaria *descharacterizada*.

1.6. Lógica Difusa no Controle do Passo

Alegando uma natureza supostamente *lingüística* do passo de cálculo, Gan utilizou *lógica difusa* em [25] e [26] para controle da constante de aprendizado. Com um custo computacional grande, mas independente da ordem do filtro, o sistema de inferência difuso proposto emprega o erro quadrático instantâneo e o número de iterações como heurísticas, controlando a variável de saída, o passo de cálculo, com uma base de regras lógicas simples.

1.7. Avaliação e Projeto dos Algoritmos

Nas seções precedentes verifica-se a diversidade existente nas estratégias para adaptação do passo no algoritmo LMS. Com motivações e filosofias diferentes, todas as estruturas possuem parâmetros que influenciam o comportamento global do algoritmo LMS. Estes parâmetros devem ser ajustados para que a evolução do passo confira ao algoritmo adaptativo velocidade de convergência, desajustes pequenos em regime permanente e capacidade de rastreamento no caso não-estacionário.

Apesar do relativo sucesso alcançado em muitas situações, pode-se avaliar e, provavelmente, melhorar o desempenho dos algoritmos de controle do passo caso se tenha um indicativo do limite de desempenho do algoritmo LMS evoluindo em passo variável, observando que a flexibilidade proporcionada pela variação do passo configura um problema *adicional* de otimização dos parâmetros intrínsecos dos algoritmos de controle.

A questão que emerge quando se pensa na utilização de passo variável nos algoritmos adaptativos é: *qual seqüência de passos* deve ser implementada pelos algoritmos de controle do passo para obter operação ótima. A partir de algumas hipóteses comuns na análise do comportamento do algoritmo LMS, pode-

se determinar analiticamente *uma seqüência de passos ótima que leve o algoritmo LMS ao melhor desempenho possível num ambiente estacionário*, formalizando a sua adaptação em passo variável. A geração da seqüência ótima a partir do modelo analítico também foi abordada por Gardner em 1983 [9] e Koike em 1999 [22].

O resultado analítico derivado é de extrema importância. Porém, é negligenciado na maioria dos trabalhos publicados [16], [17], [21], [25], [26], [27], onde muitas vezes os algoritmos são justificados ou testados comparativamente com outros algoritmos, baseando-se em resultados de simulação. Isto pode levar, e realmente leva, a conclusões equivocadas acerca dos algoritmos e seus respectivos desempenhos, como se demonstra no Capítulo 4 deste trabalho.

A evolução ótima do erro médio quadrático e a seqüência de passos ótima correspondente seriam suficientes para avaliar os algoritmos de passo variável. No entanto, observar o impacto da seqüência de passos gerada pelos algoritmos de controle sobre o comportamento do erro médio quadrático nem sempre é uma tarefa trivial, quando realizada separadamente nas curvas do erro e do passo em função do número de iterações.

As informações do erro médio quadrático, do passo de cálculo e do número de iterações podem ser combinadas numa única curva no espaço, caracterizando a evolução em passo variável de forma mais completa. Por outro lado, a visualização de curvas no espaço tridimensional é difícil, conferindo pouco valor à representação espacial. Embora não seja de grande valia, a priori, a trajetória no espaço pode ser representada num lugar geométrico até então inexplorado, o qual foi denominado *plano de aprendizado*, onde o número de iterações n é um parâmetro implícito. As trajetórias implícitas em n reúnem informações *simultâneas* do passo de cálculo e do erro médio quadrático, o que permite avaliar a evolução dos algoritmos de passo variável de forma mais completa e compacta. O que antes era feito em curvas separadas, avaliando a evolução do passo e do erro em função do número de iterações, pode ser feito

conjuntamente no plano de aprendizado, comparando as trajetórias descritas com a trajetória ótima, obtida a partir da evolução ótima do algoritmo LMS.

Partindo de algumas hipóteses simplificadoras comuns na literatura, pode-se gerar modelos analíticos que permitem obter as trajetórias de algoritmos de passo variável no lugar geométrico descrito. Estes modelos, em conjunto com o resultado analítico ótimo derivado, constituem *uma metodologia de avaliação e projeto de algoritmos de passo variável*.

A avaliação dos algoritmos é sempre possível com a metodologia derivada. Contudo, a representação dos algoritmos pela maneira proposta no decorrer do trabalho exige informações acerca do ambiente adaptativo que não são conhecidas *a priori*. Isso restringe o projeto dos algoritmos de controle do passo e de seus parâmetros a casos onde o sinal de referência é ruído branco, como explanado no Capítulo 4. Como este é o caso de problemas de identificação de sistemas, comuns em inúmeras aplicações, a técnica de projeto proposta encontra larga aplicabilidade.

Descreve-se a seguir a organização geral do trabalho, explanando sucintamente os assuntos abordados em cada capítulo.

1.8. Estrutura do Trabalho

No Capítulo 2 formula-se o problema da filtragem ótima, utilizando filtros transversais lineares e derivando a solução ótima de Wiener. Apresenta-se o algoritmo LMS como um algoritmo simples e robusto para operação de filtros adaptativos, fazendo-se uma breve revisão de alguns resultados importantes como estabilidade, desajuste em regime, etc. . Originalmente proposto com passo fixo, motiva-se a operação do algoritmo LMS em passo variável, aprimorando seu desempenho global. Descreve-se rapidamente também algumas estruturas de adaptação do passo, ressaltando a necessidade de critérios para analisar a evolução dos algoritmos de passo variável.

O ponto de partida do Capítulo 3 é a busca de resultados analíticos que forneçam indicativos absolutos de desempenho do algoritmo LMS evoluindo com

passo variável, o que é obtido com o modelo analítico do erro médio quadrático: a evolução ótima do algoritmo LMS operando com passo variável. De posse do resultado analítico ótimo desenvolve-se uma metodologia de avaliação e projeto de algoritmos de passo variável, onde a evolução dos mesmos é caracterizada por trajetórias num lugar geométrico até então inexplorado, *o plano de aprendizado*. As trajetórias são implícitas em n , permitindo observar informações *simultâneas* do passo de cálculo e do erro médio quadrático. A avaliação da convergência antes feita em curvas separadas, na curva de aprendizado e na seqüência de passos, pode ser feita comparando uma única trajetória com a trajetória ótima no plano de aprendizado.

A metodologia de avaliação e projeto de algoritmos de passo variável desenvolvida é explorada no Capítulo 4, no qual se projeta e avalia alguns algoritmos da literatura, mostrando a importância de resultados analíticos numa área dominada muitas vezes por resultados de simulação. Fruto da teoria desenvolvida no Capítulo 3, derivou-se um novo resultado para otimização da velocidade de convergência em passo fixo. Além de levar a uma *convergência mais rápida*, o novo resultado fornece *erros em regime permanente menores* que os resultados clássicos. Compara-se também duas filosofias diferentes, a correção da dinâmica de algumas estruturas frente à otimização "automática", realizada pelas estruturas de busca ótima do passo, revelando vantagens das estruturas de filtragem quando projetadas de acordo com a metodologia proposta, mesmo em ambientes sujeitos a certos tipos de não-estacionaridades.

Finalmente, no último capítulo do trabalho reúne-se os principais resultados desenvolvidos, derivando conclusões importantes e tecendo-se comentários sobre a abrangência da metodologia desenvolvida, bem como motivações para trabalhos futuros.

O Algoritmo LMS

2.1. Introdução

Desenvolvido na Universidade de Stanford, na Califórnia, por Bernard Widrow e M. E. Hoff Jr. em 1959, o algoritmo LMS representa o mais importante algoritmo adaptativo utilizado atualmente. Baseado na estimação instantânea do gradiente, assemelha-se ao método do gradiente-descendente, evoluindo estocasticamente em direção ao ponto de erro mínimo da superfície de desempenho.

No contexto da solução adaptativa o algoritmo do gradiente estocástico, como também é conhecido o LMS, apresenta-se como um algoritmo robusto, eficiente e principalmente simples, revelando uma vocação natural para diversas aplicações em processamento de sinais.

De convergência relativamente lenta quando implementado com passo fixo, o algoritmo LMS pode ter seu desempenho otimizado quando se utiliza passo variável, mediante uma escolha apropriada da seqüência de passos de cálculo empregada.

Neste capítulo formaliza-se estatisticamente o problema de otimização da filtragem, mostrando a solução adaptativa como uma alternativa vantajosa frente ao filtro clássico fixo, especialmente para aplicações em tempo real. Entre a velocidade de convergência e o desajuste pequeno em regime permanente, justifica-se o uso de passo variável, resgatando uma flexibilidade originalmente presente nos algoritmos do gradiente estocástico, mas muitas vezes negligenciada no caso do algoritmo LMS [14].

2.2. Filtragem Ótima de Wiener

Certa vez, Norbert Wiener (1894-1964) previu que chegaria o dia em que os engenheiros descobririam “uma nova abordagem para a engenharia de comunicações que é primariamente estatística” [4]. O presságio se cumpriu: do trabalho pioneiro do brilhante matemático nasceram algumas das principais idéias que permeiam as áreas de processamento de sinais, telecomunicações e teoria de controle. Um problema clássico que se manifesta nestas áreas, e que encontra uma solução possível na teoria de filtragem ótima de Wiener, é a estimação de um sinal que somente pode ser observado na presença de ruído aditivo.

Através de uma estrutura de filtragem apropriada é possível estimar o sinal desejado, apresentando ao filtro estimador observações de outro sinal tomado como referência, que deve ser correlacionado com o sinal que se deseja estimar. A idéia básica é projetar o filtro de maneira a produzir a melhor estimativa possível do sinal desejado, constituindo um problema de otimização.

A formulação matemática do problema envolve a definição de uma função de desempenho, dependente das propriedades estatísticas do sinal de interesse, do sinal de referência e do ruído. A estrutura de filtragem empregada também influencia sobremaneira a abordagem do problema. Frequentemente restringe-se a estrutura escolhida, visando a tratabilidade matemática e outras considerações de ordem prática. As premissas adotadas para a formulação do problema são [1]:

- 1. A estrutura de filtragem é linear, o que determina filtros mais simples e facilita a análise matemática;*
- 2. O filtro opera em tempo discreto, permitindo sua implementação em processadores digitais;*
- 3. A resposta ao impulso do filtro possui duração finita (FIR – “Finite Impulse Response”), gerando uma estrutura intrinsecamente estável e facilitando a otimização.*

A Figura 2.1 retrata o problema da filtragem ótima.

Uma estrutura que apresenta as características citadas é a estrutura

transversal, bastante empregada em aplicações que exigem operação em tempo real.

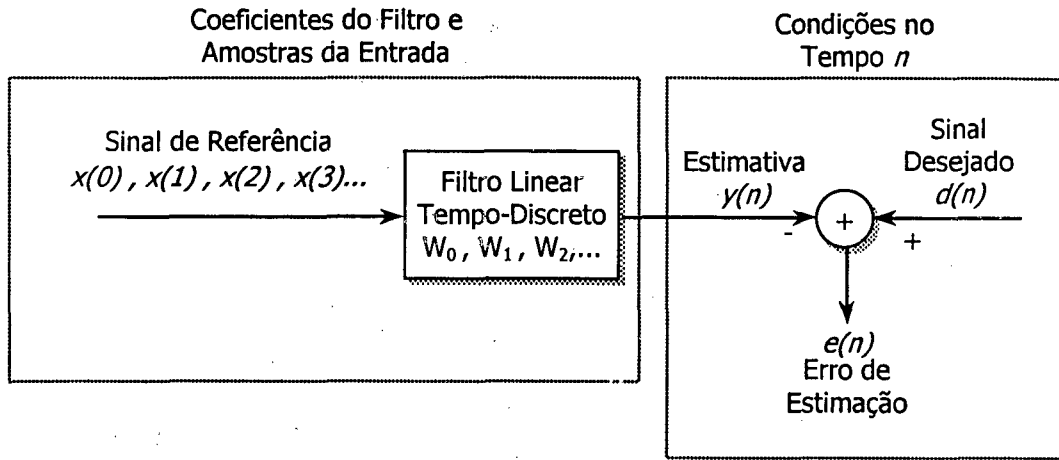


Fig. 2.1 - Formulação do Problema da Filtragem Ótima

A estrutura FIR transversal fornece uma estimativa $y(n)$ do sinal desejado $d(n)$, dada pela combinação linear das observações do sinal de referência $x(n)$ em instantes de tempo passados. A ponderação é feita pelo conjunto de coeficientes w_k que definem a resposta ao impulso do filtro:

$$y(n) = \sum_{k=0}^{N-1} w_k x(n-k) \quad (2.1a)$$

A expressão (2.1a) é muitas vezes representada na forma matricial:

$$y(n) = \mathbf{W}^T \mathbf{X}(n) = \mathbf{X}^T(n) \mathbf{W} \quad (2.1b)$$

Na expressão (2.1b) o vetor de entradas $\mathbf{X}(n)$ representa o conjunto de amostras apresentadas ao filtro no instante n :

$$\mathbf{X}(n) = [x(n), x(n-1), x(n-2), \dots, x(n-N+1)]^T \quad (2.2)$$

e o vetor \mathbf{W} contém as amostras da resposta ao impulso do filtro:

$$\mathbf{W} = [w_0, w_1, \dots, w_{N-1}]^T \quad (2.3)$$

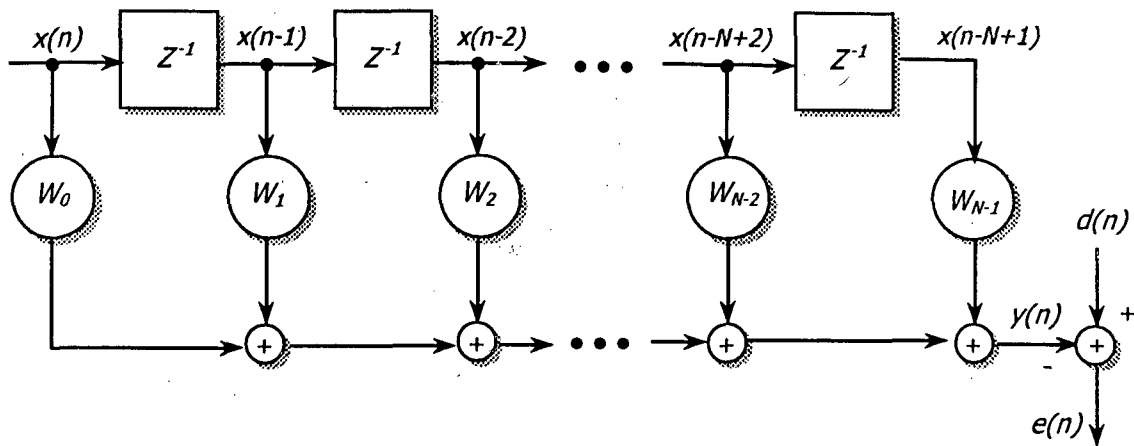


Fig. 2.2 – O filtro FIR transversal e a estimação do sinal desejado

Inerente à estimativa oferecida pelo filtro há um erro de estimação $e(n)$, definido como a diferença entre a resposta do filtro $y(n)$ e o sinal desejado $d(n)$. A Figura 2.2 detalha a estrutura do filtro FIR transversal. Desta figura e da equação (2.1b) tem-se:

$$e(n) = d(n) - y(n) \quad (2.4a)$$

$$e(n) = d(n) - \mathbf{W}^T \mathbf{X}(n) \quad (2.4b)$$

Na abordagem estatística da filtragem ótima, o sinal de referência $x(n)$ do filtro e o sinal desejado $d(n)$ são assumidos como realizações simples de processos estocásticos. Procura-se então ajustar o conjunto de coeficientes do filtro de forma a minimizar o erro de estimação $e(n)$ em algum sentido estatístico [1].

O critério estatístico utilizado na *teoria de Wiener* é a minimização do valor médio quadrático do erro de estimação, definindo a função objetivo a ser otimizada [1], [2]. A formulação estatística do problema da filtragem de Wiener pode então ser resumida como [1]:

"Projetar um filtro linear em tempo-discreto cuja saída $y(n)$ fornece

uma estimativa de uma resposta desejada $d(n)$, dado um conjunto de amostras da entrada $x(0), x(1), x(2), \dots$, tal que o valor médio quadrático do erro de estimação $e(n)$, definido como a diferença entre a estimativa fornecida pelo filtro e a resposta desejada, seja minimizado.”

Elevando (2.4b) ao quadrado obtém-se o erro de estimação quadrático:

$$e^2(n) = d^2(n) + W^T X(n) X^T(n) W - 2W^T d(n) X(n) \quad (2.5)$$

Assumindo que os sinais $d(n)$ e $x(n)$ são *estacionários* e considerando o vetor de coeficientes W fixo [3], [6] o valor esperado de (2.5) gera a função de desempenho:

$$\xi(W) = E[e^2(n)] = \sigma_d^2 + W^T E[X(n) X^T(n)] W - 2W^T E[d(n) X(n)] \quad (2.6a)$$

$$\xi(W) = \sigma_d^2 + W^T R W - 2W^T P \quad (2.6b)$$

com:

$$R = E[X(n) X^T(n)] \quad (2.7)$$

e:

$$P = E[d(n) X(n)] \quad (2.8)$$

As expressões (2.7) e (2.8) representam, respectivamente, a matriz de autocorrelação do vetor de entradas e o vetor de correlação cruzada entre o sinal desejado e o vetor de entradas.

Resultante das restrições feitas sobre a estrutura de filtragem e do critério estatístico empregado na formulação do problema, a expressão (2.6b) é claramente quadrática em W , expressando a função custo a ser minimizada. Também denominada na literatura como superfície de desempenho, trata-se de um hiperparabolóide, possuindo um *único ponto de mínimo*.

Em se tratando de uma superfície unimodal (2.6b), o gradiente nulo é

condição necessária e suficiente para determinar o conjunto ótimo de coeficientes, o qual corresponde ao ponto de mínimo da função. Denominando W_0 o vetor ótimo, tem-se então

$$\left. \nabla \xi(W) \right|_{W=W_0} = 0 \quad (2.9)$$

Da expressão (2.6b) determina-se o gradiente do erro médio quadrático:

$$\nabla \xi(W) = \frac{\partial \xi(W)}{\partial W} = 2RW - 2P \quad (2.10)$$

A aplicação da condição (2.9) na expressão do gradiente em (2.10) conduz a um importante resultado analítico conhecido como *equação de Wiener-Hopf* [1], [3], [4], [5]:

$$RW_0 = P \quad (2.11)$$

Assumindo que a matriz R é não-singular, o vetor ótimo dos coeficientes W_0 pode ser obtido da equação (2.11) :

$$W_0 = R^{-1}P \quad (2.12)$$

A expressão (2.12) é conhecida na literatura como *Solução de Wiener*, fornecendo o máximo desempenho alcançável por uma estrutura de filtragem transversal linear com resposta ao impulso finita. O filtro projetado segundo tais coeficientes leva o mesmo nome - *filtro de Wiener*.

Substituindo (2.12) na expressão (2.6b), após algum algebrismo, obtém-se o erro médio quadrático mínimo alcançável por um filtro de Wiener [3]:

$$\xi_{\min} = \sigma_d^2 - P^T R^{-1} P \quad (2.13)$$

Embora analiticamente simples e elegante, a solução de Wiener não é facilmente implementável na forma da expressão (2.12), mas define os objetivos de uma abordagem alternativa para o problema da filtragem: *a estratégia adaptativa*.

2.3.A Solução Adaptativa

Pressupondo a existência de informações estatísticas acerca dos sinais, a expressão (2.12) permite projetar o filtro FIR transversal de maneira ótima e num único passo, um sonho do ponto de vista analítico. Entretanto, a natureza de uma implementação prática inviabiliza a otimização do filtro por tal expressão - a extração em tempo real das características estatísticas dos sinais envolvidos é inviável do ponto de vista operacional. O que era formalizado sucintamente, torna-se um problema de implementação para operações em tempo real.

A dificuldade pode ser contornada utilizando-se estratégias de otimização adaptativas. Em torno do filtro é estruturado um algoritmo que retira informações disponíveis dos sinais e realiza a otimização iterativamente. O vetor de coeficientes do filtro é atualizado recursivamente, visando a solução ótima:

$$W = W(n) \quad (2.14)$$

As estruturas de filtragem implementadas desta forma denominam-se *filtros adaptativos*, possuindo algumas peculiaridades [3] :

- *São auto-otimizáveis, adaptam-se automaticamente face às variações estatísticas dos sinais, o que sugere capacidade de rastreamento em ambientes não-estacionários;*
- *Podem extrapolar um modo de comportamento e tratar novas situações após um treinamento relativamente simples, com um número finito e freqüentemente pequeno de sinais ou padrões de treinamento.*
- *Embora usualmente mais complexos e difíceis de analisar, podem aumentar substancialmente o desempenho do sistema quando as características do sinal de entrada são desconhecidas ou variantes no tempo.*

O sucesso da otimização adaptativa, contudo, depende do ambiente onde o filtro está inserido e dos algoritmos de controle da adaptação.

Com as premissas de linearidade e resposta ao impulso de duração finita, a geometria da superfície de desempenho correspondente confere uma propriedade singular à solução ótima W_0 : *é o único ponto do domínio cujo vetor gradiente é nulo*. Em qualquer outro ponto W do domínio o gradiente é não-nulo e determina a direção e sentido de máximo aumento da função naquele ponto [10]. Essa propriedade, utilizada em (2.9) para determinação da solução ótima, pode ser aproveitada numa estratégia de otimização adaptativa.

Considerando a necessidade e a suficiência da expressão (2.9), buscar a solução ótima na superfície de desempenho reflete-se em atingir o ponto de gradiente nulo. Isso leva muitos algoritmos a percorrerem o domínio da superfície de desempenho guiando-se no sentido contrário ao gradiente da função [3].

Um dos mais antigos métodos de otimização, *o método do gradiente-descendente*, utiliza o gradiente da superfície de desempenho para atualizar incrementalmente um conjunto arbitrário inicial de coeficientes $W(0)$. A equação recursiva que descreve o processo é:

$$W(n+1) = W(n) + \frac{1}{2}\mu[-\nabla\xi(n)] \quad (2.15)$$

O conjunto inicial de coeficientes $W(0)$ geralmente é escolhido como o vetor nulo, a menos que se possua algum conhecimento prévio a respeito da solução ótima.

O parâmetro μ é uma constante real positiva e controla a atualização incremental a que os coeficientes são submetidos na iteração n . Substituindo (2.10) em (2.15), chega-se ao *método do gradiente-descendente*:

$$W(n+1) = W(n) + \mu[P - RW(n)] \quad (2.16)$$

Além de herdar as características dos sistemas adaptativos, pode ser demonstrado [1] que o algoritmo do gradiente-descendente converge para a solução de Wiener com uma escolha apropriada da constante de adaptação μ . Contudo, a equação (2.16) ainda pressupõe informações estatísticas dos sinais nem sempre conhecidas, ou não facilmente estimáveis dos dados, redundando na mesma problemática da solução de Wiener em (2.12).

Uma solução possível para o impasse é estimar o gradiente dos dados, utilizando tal estimativa para guiar a adaptação. O método do gradiente-descendente, de natureza determinística, inspira a implementação de sua versão estocástica, utilizado em diversas aplicações – o *algoritmo LMS*.

2.4.O Algoritmo LMS passo Fixo

Um dos mais simples e difundidos algoritmos em processamento adaptativo de sinais, baseado na estimativa do gradiente, é o algoritmo LMS. Do inglês "Least-Mean-Square", o algoritmo LMS faz parte da família de métodos do gradiente estocástico, utilizando uma estimativa do gradiente a partir de valores instantâneos dos dados para proceder a adaptação dos coeficientes do filtro. A Figura 2.3 ilustra a situação.

O ponto de partida para a derivação do algoritmo LMS é a utilização do erro quadrático instantâneo como uma estimativa do erro médio quadrático:

$$E[e^2(n)] \approx e^2(n) \quad (2.17)$$

A aproximação feita no valor esperado do erro quadrático leva a uma simplificação substancial do ponto de vista computacional, traduzindo-se numa estimativa do gradiente extremamente simples e facilmente obtida dos dados:

$$\hat{\nabla}\xi(n) = \frac{\partial e^2(n)}{\partial W} = -2 e(n) X(n) \quad (2.18)$$

Substituindo-se a aproximação do gradiente (2.18) na expressão (2.15) deriva-se a equação de adaptação do algoritmo LMS, seguindo o mesmo princípio do algoritmo do gradiente-descendente – atualizar um conjunto inicial arbitrário de coeficientes guiando-se no sentido contrário ao gradiente estimado dos dados:

$$W(n+1) = W(n) + \frac{1}{2} \mu [-\hat{\nabla}\xi(n)] \quad (2.19)$$

$$W(n+1) = W(n) + \mu e(n) X(n) \quad (2.20)$$

Em se tratando de um algoritmo adaptativo que estima o gradiente diretamente dos dados instantâneos, é crucial que a estimativa seja não-polarizada no sentido médio estatístico. Assim, assumindo $X(n)$ e $W(n)$ independentes [1], [3], e efetuando-se o valor esperado da expressão (2.18), tem-se:

$$E[\hat{\nabla}\xi(n)] = -2 E[e(n)X(n)] = -2 E[d(n)X(n) - X(n)X^T(n)W(n)] \quad (2.21a)$$

$$E[\hat{\nabla}\xi(n)] = 2RW(n) - 2P \quad (2.21b)$$

Comparando-se (2.21b) com (2.10) resulta:

$$E[\hat{\nabla}\xi(n)] = \nabla\xi(n) \quad (2.22)$$

Conclui-se então que a estimativa do gradiente utilizada pelo LMS é não-polarizada no sentido médio estatístico, podendo ser considerada como a composição do gradiente verdadeiro e de uma parcela ruidosa aditiva [3]. Pode-se então conceber o algoritmo LMS como uma *implementação ruidosa do algoritmo do gradiente-descendente* [6], o que leva muitos autores a denominá-lo

alternativamente como *algoritmo do gradiente estocástico* [1], [9], [14].

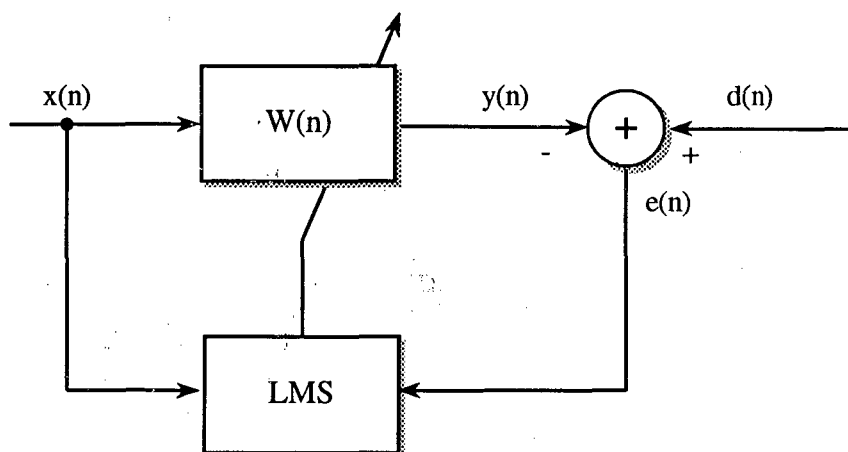


Fig. 2.3 – O algoritmo LMS

2.5. Estabilidade e Desempenho do Algoritmo LMS com Passo Fixo

O objetivo dos algoritmos de aprendizado é conduzir os filtros adaptativos transversais lineares iterativamente à solução de Wiener. Características como estabilidade, velocidade de convergência e desempenho em regime permanente constituem idiosincrasias dos algoritmos.

Semelhante ao algoritmo do gradiente-descendente, o algoritmo LMS também possui uma constante real positiva μ que controla a atualização dos coeficientes e determina seu comportamento durante a evolução do processo adaptativo.

Substituindo a equação (2.4b) em (2.20) explicita-se uma característica importante do algoritmo do gradiente estocástico:

$$W(n+1) = [I - \mu X(n)X^T(n)] W(n) + \mu d(n)X(n) \quad (2.23)$$

O termo $[I - \mu X(n)X^T(n)]$ em (2.23) representa uma malha de realimentação em torno da estrutura adaptativa, o que certamente determina restrições sobre a estabilidade do sistema. Ainda que construído em torno de uma

estrutura estável por definição (o filtro FIR), o algoritmo LMS não possui sua convergência assegurada a priori, sendo necessário atuar sobre o fator $[I - \mu X(n)X^T(n)]$.

Uma vez que, em princípio, não se tem controle sobre o vetor de entradas $X(n)$, as restrições de estabilidade ditadas pela realimentação irão incidir sobre o passo de cálculo, que deve ser escolhido adequadamente, dado o critério de convergência utilizado.

Na literatura [1], [3], [5], [12], a análise estatística da convergência do algoritmo LMS é geralmente feita sobre o *vetor de erro nos coeficientes*, definido como:

$$V(n) = W(n) - W_0 \quad (2.24)$$

De acordo com a definição (2.24), a convergência do filtro adaptativo para a solução de Wiener se traduz na convergência do vetor de erro nos coeficientes para o vetor nulo. Para analisar o comportamento do algoritmo LMS com a definição proposta, escreve-se a sua equação de adaptação em função do vetor de erro nos coeficientes, aplicando (2.24) e (2.4b) em (2.20) e considerando que $d(n) = X^T(n)W_0$:

$$V(n+1) = [I - \mu X(n)X^T(n)]V(n) \quad (2.25)$$

Assumindo a independência estatística entre $X(n)$ e $V(n)$ e efetuando o valor esperado da equação recursiva (2.25), revela-se o seu comportamento médio e a influência exercida pela estatística do sinal de referência do filtro:

$$E[V(n+1)] = [I - \mu R]E[V(n)] \quad (2.26)$$

A equação (2.26) mostra que é possível controlar a evolução média dos coeficientes atuando sobre o valor do passo de cálculo, e assim garantindo a convergência segundo o critério adotado:

$$\lim_{n \rightarrow \infty} E[V(n)] = [0] \quad (2.27)$$

A convergência da expressão (2.27) se reflete nos coeficientes do filtro adaptativo. Se o valor médio de $V(n)$ em (2.27) tende ao vetor nulo, então, de (2.24), a solução obtida com o algoritmo LMS é não-polarizada no sentido médio dos coeficientes:

$$\lim_{n \rightarrow \infty} E[W(n)] = W_0 \quad (2.28)$$

Embora revele a influência da matriz de autocorrelação R na evolução do algoritmo LMS, a expressão (2.26) não permite analisar claramente as condições necessárias à convergência. A evolução do processo repousa sobre a natureza da matriz R , que pode ser decomposta apropriadamente de forma a explicitar a sua estrutura de autovalores e autovetores [4]:

$$R = Q \Lambda Q^T \quad (2.29)$$

onde Q é a matriz de autovetores de R e $\Lambda = \text{diag}[\lambda_0 \lambda_1 \lambda_2 \dots \lambda_{N-1}]$ a matriz diagonal dos autovalores λ_i correspondente.

Pré-multiplicando (2.26) por Q^T e utilizando a relação (2.29), desacopla-se a equação matricial (2.26) em N modos independentes, permitindo concluir mais facilmente sobre a convergência dos coeficientes do filtro adaptativo:

$$E[V'(n+1)] = [I - \mu \Lambda] E[V'(n)] \quad (2.30)$$

onde $V'(n) = Q^T V(n)$.

Para garantir a estabilidade no sentido médio dos coeficientes, nitidamente todos os modos na expressão (2.30) devem satisfazer a condição

$$|1 - \mu \lambda_i| < 1 \quad ; \quad i = 0, 1, \dots, N-1 \quad (2.31)$$

O caso crítico em (2.31) é dado pelo autovalor de maior magnitude, ditando a *condição necessária e suficiente para convergência do algoritmo LMS no sentido médio dos coeficientes*:

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (2.32)$$

Escolhendo o passo de acordo com (2.32) garante-se a convergência do algoritmo LMS para a solução de Wiener no sentido médio estatístico. Todavia, esse critério de convergência não é suficiente em situações práticas [1], já que não garante a *limitação das flutuações dos coeficientes em torno da média*.

Um critério que exerce limites sobre a *dispersão* das variáveis aleatórias do sistema, já utilizado na Seção 2.2 do presente capítulo, é o critério médio quadrático. A convergência da variância dos coeficientes ou do erro médio quadrático exige limites mais restritivos em μ . Horowitz e Senne determinaram uma condição necessária e suficiente para sinais gaussianos [6]:

$$0 < \mu < \frac{2}{3\lambda_{\max}} \quad (2.33)$$

A derivação da condição (2.33) é laboriosa e não será feita aqui, mostrando-se apenas o resultado. A dedução pode ser encontrada detalhadamente em [6]. Uma aproximação conservativa da expressão (2.33) consiste em substituir o autovalor λ_{\max} pelo traço da matriz R , chegando ao resultado derivado em [11]:

$$0 < \mu < \frac{2}{3\text{Tr}[R]} \quad (2.34)$$

Além da estabilidade, o comportamento dos algoritmos durante a convergência e o desempenho em regime permanente são características fundamentais, influenciando na escolha do algoritmo a ser utilizado no aprendizado dos filtros adaptativos.

Embora seja simples, robusto, eficiente e estável do ponto de vista numérico [12], o algoritmo LMS apresenta em regime permanente um erro médio quadrático *maior do que o erro mínimo dado pela solução de Wiener*. É o preço a ser pago pela aproximação feita no gradiente [1]. O erro em excesso relativo ao erro mínimo é um indicativo de desempenho do algoritmo LMS, dado pela

diferença entre o erro médio quadrático e o nível mínimo de erro médio quadrático correspondente à solução de Wiener:

$$\xi_{ex}(n) = \xi(n) - \xi_{min} \quad (2.35)$$

Os erros médio quadrático em excesso e médio quadrático mínimo são utilizados na definição de um critério muito empregado na avaliação da qualidade da estimação de filtros adaptativos em regime permanente. Constituindo uma medida do quão próximo o LMS encontra-se do ponto ótimo no sentido médio quadrático do erro [1], [7], o *desajuste* é definido como:

$$M = \frac{\xi_{ex}(\infty)}{\xi_{min}} \quad (2.36)$$

Em se tratando do algoritmo LMS, o desajuste está intimamente ligado à escolha feita sobre o passo de cálculo. Para passos suficientemente pequenos, o desajuste é dado por [1], [3], [7], [28]:

$$M \approx \frac{\mu}{2} \cdot \text{Tr}[R] \quad (2.37)$$

onde $\text{Tr}[R]$ é o traço da matriz R .

A expressão (2.37) ilustra a influência do passo no desempenho do algoritmo LMS em regime permanente. *Passos pequenos* minimizam o efeito do ruído presente na estimação instantânea do gradiente, reduzindo o desajuste.

Durante o período transitório o comportamento do algoritmo LMS também sofre influência do passo. Isso pode ser inferido das expressões (2.20) e (2.23), nas quais verifica-se que o passo controla o ajuste incremental dos coeficientes. Grosseiramente, *passos grandes* levam a ajustes maiores a cada iteração, portanto a velocidades de convergência maiores.

Essa é a limitação do algoritmo LMS operando em passo fixo: existe uma solução de compromisso entre o desajuste em regime permanente e a velocidade de convergência no que diz respeito à escolha do passo [15]. Ou otimiza-se o regime permanente em detrimento do regime transitório, ou abre-se

mão do desajuste em prol da velocidade de convergência.

Porém, as exigências da adaptação no que concerne a escolha do passo ocorrem em estágios diferentes do aprendizado do filtro. As restrições impostas pela operação em passo fixo podem ser relaxadas quando se utiliza passos distintos em diferentes períodos da adaptação – é a *operação em passo variável*.

2.6. O Algoritmo LMS Operando com Passo Variável

Único parâmetro de projeto no algoritmo LMS convencional, o passo de cálculo fixo limita seu desempenho. Esta é a principal motivação para operação do LMS em passo variável, conferindo ao algoritmo uma flexibilidade antes inexistente. A equação de adaptação do algoritmo LMS com passo variável apresenta uma variável adicional:

$$W(n+1) = W(n) + \mu(n)e(n)X(n) \quad (2.38)$$

Heuristicamente falando, inicia-se a adaptação com passos grandes, proporcionando velocidade de convergência, como sugerem as expressões (2.20) e (2.23). Com a proximidade do regime permanente, diminui-se gradativamente o passo, reduzindo o desajuste como indica a expressão (2.37).

No caso estacionário, a operação em passo variável permite ao LMS atingir um dado desajuste de projeto em menos iterações frente à implementação clássica com passo fixo, otimizando a dinâmica do processo. A variação do passo também conduz a melhores desempenhos quando as especificações de projeto limitam o número máximo de iterações, permitindo atingir um desajuste menor no final das iterações disponíveis.

Em ambientes não-estacionários, os algoritmos de passo variável também podem ser mais eficientes no rastreamento da solução ótima quando comparados à adaptação em passo fixo [15].

A flexibilidade oferecida pela variação do passo, entretanto, exige o emprego de um *algoritmo específico para a atualização de $\mu(n)$* em (2.38), adequando o passo ao estado corrente da adaptação. Vários algoritmos para

adaptação do passo baseados no LMS foram propostos na literatura [16], [17], [18], [19], [20], [21], [22], [25], [26], [27], dentre outros. O controle do passo geralmente é feito tomando alguma variável do processo adaptativo como referência para a implementação da política descrita: diminuir gradativamente o passo com o avanço do processo adaptativo.

Contudo, a opção pelo emprego de passo variável introduz novos parâmetros que influenciam o comportamento do algoritmo. Cada estratégia de adaptação do passo altera de forma diferente o desempenho global do algoritmo em termos de estabilidade, velocidade de convergência, desajuste e, no caso de sinais não-estacionários, capacidade de rastreamento.

Os passos gerados pelos algoritmos de controle geralmente possuem natureza estocástica, em princípio podendo assumir quaisquer valores. A primeira medida a ser tomada então, é garantir a estabilidade, mantendo o passo dentro de um intervalo estável. Pode-se utilizar as expressões (2.32), (2.33), ou (2.34) para orientar a escolha do intervalo. No Capítulo 4 demonstra-se que além da estabilidade, a otimização da velocidade de convergência também impõe um limite superior para o passo, em geral distinto daqueles em (2.32), (2.33), ou (2.34).

Rastreamento no caso não-estacionário e desajuste em regime permanente são características comprometidas pelo valor inferior do passo [7], [16], [17]. Embora possam minimizar consideravelmente o desajuste em ambientes estacionários, passos muito pequenos limitam a capacidade de reação do algoritmo adaptativo às mudanças nas características estatísticas dos sinais no caso não-estacionário. A expressão (2.37) ou os resultados derivados em [1], [7] podem ser utilizados para orientar essa escolha. Caso haja necessidade, expressões mais precisas para o desajuste no caso estacionário são fornecidas no Capítulo 3.

Agrupando os limites inferior e superior do passo constitui-se o *intervalo de truncagem*, presente na implementação de todos os algoritmos de passo variável:

$$\mu(n) \in [\mu_{\min}, \mu_{\max}] \quad \forall n \quad (2.39)$$

A operação do passo dentro do intervalo dado por (2.39) garante ao LMS algumas das características esperadas de um algoritmo adaptativo, como estabilidade e convergência para a solução de Wiener. No entanto, não assegura um bom desempenho dinâmico, deixando em aberto a seqüência de passos que deve ser gerada para otimizar a convergência do algoritmo LMS. O emprego de uma seqüência adequada ao contexto de adaptação pode melhorar significativamente o desempenho global do algoritmo [14], [15], [22].

2.7. Algumas Estratégias de Adaptação com Passo Variável.

Com estruturas e motivações diversas, muitos algoritmos adaptativos de passo variável foram propostos na literatura. Alguns serão abordados neste trabalho e são descritos sucintamente a seguir.

Publicado em 1992, o algoritmo desenvolvido por Kwong e Johnston [16] utiliza uma versão filtrada do erro quadrático instantâneo para adaptar o passo, resultando numa estrutura de atualização recursiva. Trata-se de um algoritmo simples, com custo computacional adicional baixo e independente da ordem do filtro adaptativo, em relação ao caso com passo fixo. Possui dois graus de liberdade de projeto. Uma constante é escolhida empiricamente, a outra é ajustada de acordo para atingir o desajuste desejado em regime permanente.

A suposta sensibilidade ao ruído do algoritmo desenvolvido por Kwong e Johnston inspirou Aboulnasr e Mayyas [17] a modificarem a estrutura original daquele algoritmo, introduzindo uma nova figura de mérito na equação de adaptação do passo. Utilizando uma estimativa da autocorrelação do erro, obtida por uma equação diferenças auxiliar que filtra a quantidade $e(n)e(n-1)$, os autores afirmam ter diminuído a influência do ruído de medição aditivo, onipresente em situações práticas. Entretanto, o que em princípio tornaria o algoritmo vantajoso

na presença de ruído, em algumas situações não proporciona a superioridade descrita em [17] frente ao algoritmo original proposto em [16], conforme será mostrado no Capítulo 4. Simples computacionalmente, também é independente da ordem do filtro, possuindo um grau de liberdade a mais do que o algoritmo original em [16], perfazendo três constantes de projeto.

Algumas estratégias *determinísticas* para variação do passo foram propostas e analisadas na literatura [12], [14]. Seguindo a abordagem originalmente presente nos algoritmos de aproximação estocástica [1], [14], Macchi analisou em [12] uma estrutura cujo passo é feito inversamente proporcional ao número de iterações n , com constantes a ajustar. Em [14] Dirk Slock propõe uma estratégia muito similar, necessitando o ajuste de alguns parâmetros de controle. Aplicou a estratégia para controlar o algoritmo LMS normalizado (NLMS), uma variante do LMS, comparando-o posteriormente com o algoritmo LMS convencional controlado pela mesma estrutura.

Alegando uma natureza supostamente *lingüística* do passo de cálculo, Gan utilizou *lógica difusa* para controlar o passo de cálculo em [25], [26]. Com um custo computacional elevado, mas independente da ordem do filtro, o sistema de inferência difuso proposto por Gan emprega o erro quadrático instantâneo e o número de iterações como heurísticas, controlando a variável de saída, o passo de cálculo, com uma base de regras lógicas simples.

Koike sugeriu um algoritmo de controle em que o passo é aproximado ao valor ótimo teórico a cada iteração, utilizando equações que implementam o controle quase-ótimo do passo de aprendizado via acumuladores com perdas [22]. Abrangendo alguns algoritmos derivados do LMS, o algoritmo é computacionalmente dependente da ordem do filtro e realiza divisão numa das equações de adaptação, o que pode representar limitações em certos processadores. As simulações apresentadas sugerem um bom desempenho para diferentes algoritmos derivados do LMS, tais como o algoritmo sign-sign LMS, o algoritmo LMF ("Least Mean Fourth") [23], o algoritmo SRA ("Sign Regressor Algorithm"), dentre outros [22], [24].

Na literatura existem muitas contribuições relevantes nesta área. Contudo, a escassez de critérios ou de uma metodologia de avaliação de desempenho dinâmico, e algumas vezes de desempenho em regime permanente, torna incerta a escolha por um algoritmo adequado à aplicação desejada.

2.8. Conclusão

A utilização de estratégias de otimização adaptativas apresenta vantagens expressivas em relação ao projeto do filtro ótimo fixo. Além de dispensarem o conhecimento prévio de informações estatísticas mais precisas dos sinais, as estruturas adaptativas podem responder a variações ocorridas no ambiente, generalizando um modo de comportamento e adaptando-se a novas situações, utilizando em geral, poucas informações *a priori* [3].

Em se tratando de algoritmos adaptativos, o algoritmo LMS é certamente o mais utilizado em situações práticas. Nas situações típicas de processamento de sinais, sempre pode ser pensado como uma alternativa viável. Sua robustez e simplicidade inerentes o tornam um forte candidato na maioria das aplicações em tempo real. Todavia, a formulação clássica com passo fixo limita o desempenho global do algoritmo; escolher um passo que privilegie simultaneamente a dinâmica e o regime permanente é uma tarefa conflitante.

As limitações impostas pela operação em passo fixo inspiram a utilização de passo variável, sempre preferível quando realizada adequadamente. Entretanto, o emprego de passo variável necessita de algoritmos de controle que realizem a adaptação do passo com a evolução do aprendizado do filtro.

Em ambientes estacionários, os algoritmos de passo variável podem ser vistos como uma tentativa de otimizar a dinâmica do algoritmo LMS mantendo o mesmo nível de desajuste em regime permanente. Em ambientes não-estacionários, também mostram-se geralmente mais eficientes no rastreamento da solução ótima [15].

Com estruturas e filosofias distintas, os inúmeros algoritmos de passo variável existentes possuem parâmetros próprios que controlam o comportamento

do passo e a evolução do algoritmo LMS. Além da opção por uma estrutura em especial, a escolha adequada dos parâmetros subjacentes é geralmente dependente dos dados [21], o que torna a flexibilidade proporcionada pelo emprego de passo variável um problema adicional de otimização: *que conjunto de parâmetros deve ser empregado para implementar uma seqüência adequada? Qual é a seqüência de passos adequada?*

No próximo capítulo, a partir de um modelo analítico do erro médio quadrático (Apêndice) deriva-se analiticamente *a evolução ótima do algoritmo LMS*. Em conjunto com uma *representação gráfica da evolução da adaptação em passo variável*, propõe-se uma metodologia de avaliação e projeto de algoritmos de passo variável, auxiliando na resposta das questões levantadas.

Metodologia de Avaliação e Projeto de Algoritmos Adaptativos de Passo Variável

3.1. Introdução

Um problema que fatalmente se manifesta quando se decide pelo uso de algoritmos de passo variável é *qual seqüência de passos* deve ser implementada. Uma escolha displicente para a seqüência de passos gerada não oferece garantias, podendo inclusive *deteriorar* o desempenho do algoritmo LMS.

Apesar do sucesso alcançado em algumas situações, pode-se avaliar e melhorar sistematicamente o desempenho dos algoritmos de controle do passo caso se tenha um indicativo do limite de desempenho do algoritmo LMS evoluindo em passo variável. A determinação deste limite deve levar em conta que a flexibilidade proporcionada pela variação do passo configura um problema *adicional* de otimização.

A partir de algumas hipóteses empregadas com freqüência na análise do comportamento do algoritmo LMS, pode-se determinar analiticamente uma seqüência ótima de passos que o leve ao melhor desempenho possível num ambiente estacionário, formalizando a sua adaptação em passo variável.

A análise das estruturas que ajustam o passo com o progresso da adaptação é ainda mais complexa, o que leva a adoção de hipóteses adicionais àquelas presentes na teoria da independência. Empregando algumas das premissas estatísticas utilizadas normalmente nos trabalhos desenvolvidos na área, pode-se estender o modelo analítico do erro médio quadrático com passo fixo para descrever a evolução dos algoritmos de passo variável baseados no algoritmo LMS.

Este capítulo faz uma revisão analítica da evolução do algoritmo LMS com passo variável, fornecendo uma nova representação do regime transitório e permanente do algoritmo. Aplicando a seqüência ótima de passos e o modelo

analítico do erro médio quadrático apresentados, propõe-se técnicas para avaliação e projeto de algoritmos adaptativos de passo variável da família LMS.

3.2.0 Modelo Analítico do LMS Passo Fixo

A análise de algoritmos adaptativos implica na definição de uma figura de mérito, um índice de desempenho, que caracterize a evolução dos mesmos. Observando o comportamento da figura de mérito, pode-se concluir sobre a natureza dos algoritmos, prevendo seus comportamentos e projetando seus parâmetros.

Uma vez que os algoritmos adaptativos realizam iterativamente a otimização do filtro visando a solução de Wiener, a avaliação dos algoritmos geralmente herda a observação do mesmo critério da teoria de filtragem ótima de Wiener, o erro médio quadrático. Todavia, podem ser utilizados outros critérios [1], [12].

Neste trabalho, o critério utilizado será o erro médio quadrático, a figura de mérito mais difundida na literatura [3], [6], [14]. O aprendizado dos algoritmos é então avaliado pela evolução do erro médio quadrático em função do número de iterações, constituindo a *curva de aprendizado*, na qual se observa o comportamento dinâmico e a qualidade da solução apresentada pelos algoritmos após a convergência, refletida pelo erro médio quadrático em regime permanente.

A Teoria da Independência

A simplicidade do algoritmo LMS infelizmente não se preserva em sua análise estatística. Na equação (2.23) revela-se a natureza recursiva não-linear do algoritmo, o que torna difícil sua análise. Frequentemente é necessário assumir hipóteses ou aproximações acerca dos sinais envolvidos para tornar a análise matemática factível.

Para o caso especial em que o conjunto de vetores $\{X(n), d(n)\}$ é não-correlacionado no tempo, é possível caracterizar completamente a evolução do erro médio quadrático em termos dos *momentos de quarta ordem* do processo $\{X(n), d(n)\}$, que podem ser relacionados a *momentos de segunda ordem* se os processos $\{X(n), d(n)\}$ são assumidos conjuntamente gaussianos e com média nula [6].

Essas premissas são vastamente adotadas na literatura, sendo reunidas em um conjunto de hipóteses simplificadoras conhecido como *hipótese ou teoria da independência*. Considerando a nomenclatura da Figura 2.3, estas hipóteses podem ser formalizadas da seguinte maneira [6], [9], [21]:

- *O conjunto $\{X(n), d(n)\}$ é estatisticamente independente do conjunto $\{X(k), d(k)\}$ para $n \neq k$;*
- (H.1)

- *O vetor de entradas $X(n)$ e a resposta desejada $d(n)$ são simples realizações de processos estocásticos estacionários conjuntamente gaussianos e com média nula;*
- (H.2)

Estas premissas são claramente inválidas para algumas aplicações, tais como filtros transversais [6]. No entanto, um princípio corrente na filosofia da ciência diz que *um bom modelo é aquele que preserva a mesma estrutura lógica do sistema sob estudo* [29]. Sob este princípio, as hipóteses agrupadas na teoria da independência ainda retém suficiente informação sobre a estrutura lógica do processo adaptativo, gerando resultados confiáveis em diversas aplicações práticas [1].

O Modelo Analítico

À luz da teoria da independência pode-se desenvolver um modelo analítico para a evolução do erro médio quadrático do sistema mostrado na Figura 2.3.

Substituindo a definição do vetor de erro nos coeficientes (2.24) em (2.4b) tem-se:

$$e(n) = d(n) - V^T(n)X(n) - W_o^T X(n) \quad (3.1)$$

Definindo o *erro produzido pelo filtro de Wiener* como:

$$e_o(n) = d(n) - W_o^T X(n) \quad (3.2)$$

Com a definição (3.2), a expressão (3.1) resulta em:

$$e(n) = e_o(n) - V^T(n)X(n) \quad (3.3)$$

Elevando a expressão (3.3) ao quadrado tem-se:

$$e^2(n) = e_o^2(n) - 2e_o(n) V^T(n)X(n) + V^T(n)X(n)X^T(n)V(n) \quad (3.4)$$

Tomando o valor esperado de (3.4) chega-se ao erro médio quadrático:

$$\xi(n) \triangleq E[e^2(n)] = E[e_o^2(n)] - 2E[e_o(n) V^T(n)X(n)] + E[V^T(n)X(n)X^T(n)V(n)] \quad (3.5)$$

Definindo $\xi_{\min} = E[e_o^2(n)]$ como o erro médio quadrático mínimo correspondente à solução de Wiener e utilizando o *princípio da ortogonalidade* (Apêndice), a expressão (3.5) resulta em:

$$\xi(n) = \xi_{\min} + E[V^T(n)X(n)X^T(n)V(n)] \quad (3.6)$$

A evolução do erro médio quadrático está condicionada então à determinação do último termo da equação (3.6). Em se tratando de uma quantidade escalar, a equação (3.6) fica inalterada se escrita em função dos traços de suas parcelas:

$$\xi(n) = \xi_{\min} + \text{Tr}\{E[V^T(n)X(n)X^T(n)V(n)]\} \quad (3.7)$$

Sem prejuízo de generalização, os operadores “traço” e “valor esperado” podem ser comutados:

$$\xi(n) = \xi_{\min} + E[\text{Tr}\{V^T(n)X(n)X^T(n)V(n)\}] \quad (3.8)$$

Usando propriedades do traço do produto de matrizes, pode-se reescrever (3.8) como:

$$\xi(n) = \xi_{\min} + E[\text{Tr}\{X(n)X^T(n)V(n)V^T(n)\}] \quad (3.9)$$

Restituindo a ordem original dos operadores e invocando a teoria da independência:

$$\xi(n) = \xi_{\min} + \text{Tr}\{E[X(n)X^T(n)]E[V(n)V^T(n)]\} \quad (3.10)$$

Utilizando (2.7) e definindo $K(n) = E[V(n)V^T(n)]$ como a *matriz de correlação do vetor de erro nos coeficientes* chega-se à expressão (3.11):

$$\xi(n) = \xi_{\min} + \text{Tr}\{RK(n)\} \quad (3.11)$$

A expressão (3.11) descreve a evolução do erro médio quadrático a partir da evolução da matriz $K(n)$. No algoritmo LMS, a evolução de $K(n)$ pode ser determinada recursivamente através de uma equação de diferenças. Esta equação é determinada no Apêndice e é dada por:

$$K(n+1) = K(n) - \mu \cdot [RK(n) + K(n)R] + \mu^2 \cdot [R \cdot \text{Tr}\{RK(n)\} + 2RK(n)R + R \cdot \xi_{\min}] \quad (3.12)$$

As equações (3.11) e (3.12) constituem o *modelo analítico para a evolução do erro médio quadrático do algoritmo LMS com passo fixo*. Descrevem o comportamento do algoritmo refletido em sua figura de mérito, o que permite analisar a evolução da adaptação *a priori*, levando em consideração as informações estatísticas disponíveis da especificação do problema.

A utilização do modelo analítico é simples. Dada uma inicialização e selecionando-se um valor apropriado para o passo, gera-se a curva de

aprendizado correspondente, com características próprias, taxa de convergência e erro em regime permanente.

Desenvolvido para passo fixo, o modelo também pode ser convenientemente explorado para abordar a evolução em passo variável, gerando resultados importantes, como mostram as seções seguintes.

3.3. Evolução Ótima do Algoritmo LMS

Um resultado importante que pode ser extraído do modelo analítico com passo fixo é a evolução ótima do algoritmo LMS. Representando um critério absoluto da convergência do algoritmo com passo variável, a evolução ótima é fundamental para avaliação das estruturas de controle do passo de cálculo existentes.

Traçando-se curvas de aprendizado para a mesma inicialização e valores de passo gradativamente diferentes, pode-se gerar uma superfície de aprendizado $\xi(\mu, n)$, parametrizada em μ , que representa todas as possibilidades de evolução do algoritmo LMS com passo fixo. Esta superfície revela graficamente o compromisso subjacente à escolha do passo para otimização da convergência.

Como mostra a Figura 3.1, para uma situação típica, é possível descrever uma trajetória ótima pela superfície, buscando a cada iteração o ponto de erro mínimo e o passo correspondente. Esse ponto pertence a uma curva de aprendizado que foi gerada com passo fixo, desde a inicialização. Assim, na iteração n existe uma curva de aprendizado de passo fixo que atinge o mesmo patamar de erro que a curva de aprendizado de passo variável, resultante da busca na superfície. Embora a seqüência de passos obtida com essa estratégia de otimização proporcione velocidade de convergência com desajustes pequenos em regime permanente, ela não leva o algoritmo LMS ao limite de desempenho com passo variável, não representando, portanto, a evolução ótima do algoritmo.

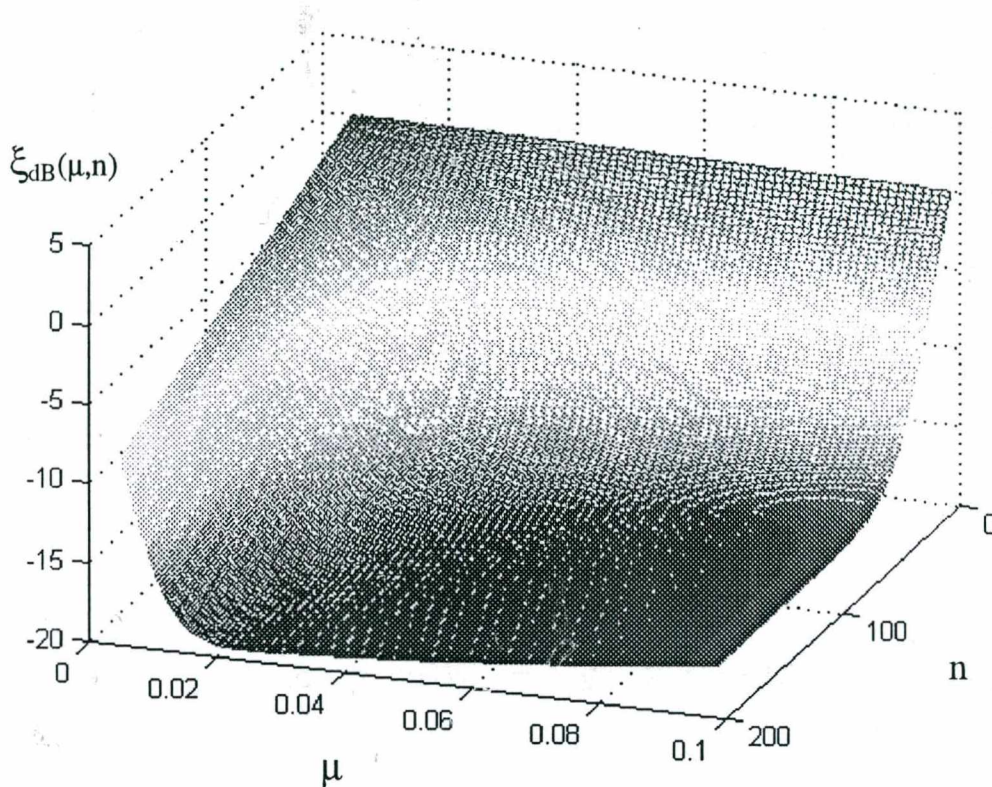


Fig. 3.1 - Superfície de aprendizado $\xi(\mu, n)$

A evolução ótima pode ser obtida mudando-se a estratégia de otimização, determinando o passo ótimo simultaneamente à evolução do erro, desde a inicialização até o regime permanente. Retornando às equações do modelo analítico do algoritmo LMS com passo fixo, verifica-se na equação de atualização (3.12) uma dependência quadrática em função do passo. O problema de otimização pode ser formulado da seguinte maneira: partindo de uma certa condição inicial $\xi(n)$, deseja-se escolher o passo que determine o maior decréscimo possível do erro médio quadrático na evolução para o patamar seguinte $\xi(n+1)$. Escolhendo o passo que otimize a atualização na equação (3.12) a cada iteração, gera-se a seqüência ótima de passos a ser implementada pelos algoritmos de passo variável. Escrevendo o erro médio quadrático em (3.11) para a iteração $(n+1)$ tem-se:

$$\xi(n+1) = \xi_{\min} + \text{Tr}\{\text{RK}(n+1)\} \quad (3.13)$$

Substituindo (3.12) em (3.13) , com alguma manipulação, chega-se à expressão que descreve a evolução do algoritmo do estágio atual n para o patamar de erro seguinte em $(n+1)$:

$$\begin{aligned} \xi(n+1) = & \xi_{\min} + \text{Tr}\{\text{RK}(n)\} - 2\mu \cdot \text{Tr}\{\text{R}^2\text{K}(n)\} \\ & + \mu^2 \cdot [\text{Tr}\{\text{R}^2\} \cdot \text{Tr}\{\text{RK}(n)\} + 2 \cdot \text{Tr}\{\text{R}^3\text{K}(n)\} + \xi_{\min} \cdot \text{Tr}\{\text{R}^2\}] \end{aligned} \quad (3.14)$$

A equação (3.14) é um polinômio do segundo grau em μ , cujos coeficientes dependem das propriedades estatísticas dos sinais e do estado da adaptação na iteração n . Pode ser demonstrado que o termo quadrático em μ é sempre positivo [4], o que determina então a existência de um único passo que leva o algoritmo ao maior decréscimo possível do erro em direção à iteração seguinte. Portanto, a condição necessária e suficiente para determinação do passo ótimo na iteração n é dada pelo gradiente da função (3.14) relativo a μ :

$$\frac{\partial}{\partial \mu} [\xi(n+1)]_{\mu=\mu_{\text{opt}}} = 0 \quad (3.15)$$

Aplicando a condição (3.15) em (3.14) pode-se determinar analiticamente a seqüência de passos:

$$0 - 2 \cdot \text{Tr}\{\text{R}^2\text{K}(n)\} + 2\mu_{\text{opt}} \cdot [\text{Tr}\{\text{R}^2\} \cdot \text{Tr}\{\text{RK}(n)\} + 2 \text{Tr}\{\text{R}^3\text{K}(n)\} + \xi_{\min} \cdot \text{Tr}\{\text{R}^2\}] = 0 \quad (3.16)$$

O passo ótimo para a iteração n , que determina a evolução do algoritmo LMS para o menor patamar de erro médio quadrático possível na iteração $(n+1)$, é:

$$\mu_{\text{opt}}(n) = \frac{\text{Tr}\{\text{R}^2\text{K}(n)\}}{\text{Tr}\{\text{R}^2\} \cdot \text{Tr}\{\text{RK}(n)\} + 2 \text{Tr}\{\text{R}^3\text{K}(n)\} + \xi_{\min} \cdot \text{Tr}\{\text{R}^2\}} \quad (3.17)$$

Utilizando o passo $\mu = \mu_{\text{opt}}(n)$ na expressão (3.12) para todo n , em conjunto com a expressão (3.11), gera-se a *evolução ótima do erro médio quadrático no algoritmo LMS para operação em passo variável*, satisfeitas as hipóteses simplificadoras utilizadas na determinação do modelo analítico do erro.

Gardner utilizou uma idéia similar em [9], mas a manteve no âmbito teórico. Koike derivou uma expressão para a seqüência ótima e utilizou-a para inspirar uma estrutura de controle quase-ótima do passo [22].

A idéia aqui é utilizar a evolução ótima derivada em conjunto com uma representação apropriada da evolução dos algoritmos em passo variável, constituindo assim *uma metodologia de avaliação e projeto dos algoritmos de controle do passo*.

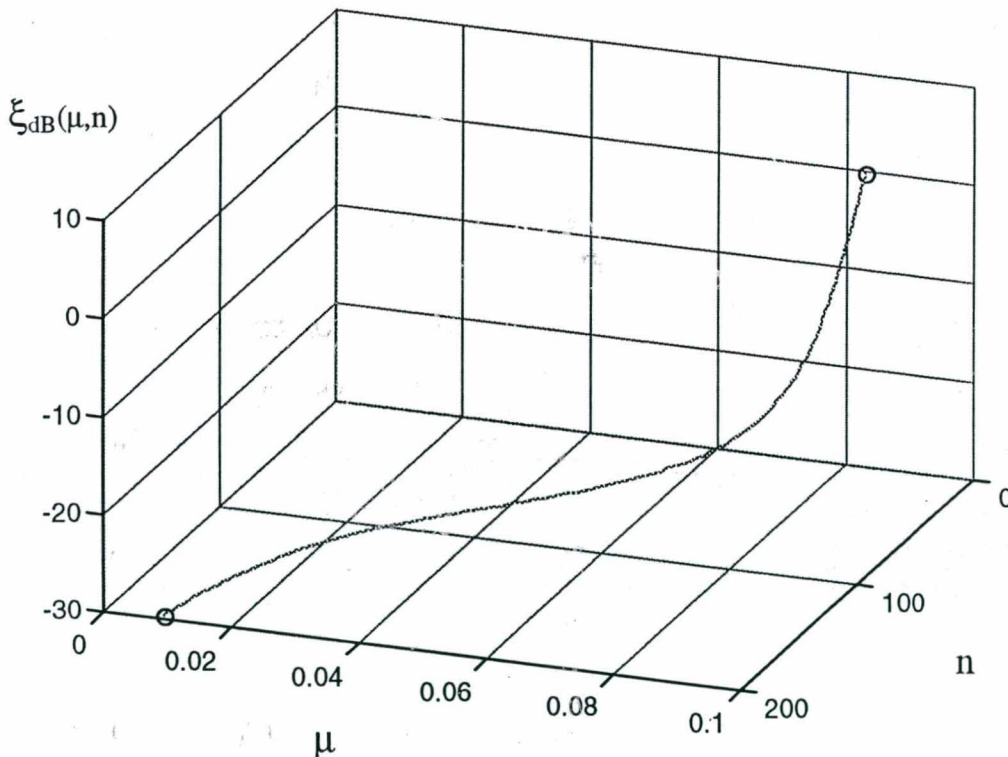


Fig. 3.2 - Trajetória ótima percorrida no espaço $[\xi, \mu, n]$

Assim como a evolução ótima (Figura 3.2), os algoritmos de passo variável baseados no algoritmo LMS também podem ser representados por trajetórias no espaço tridimensional $[\xi, \mu, n]$, de acordo com a seqüência de passos gerada e o erro médio quadrático correspondente. A evolução destes algoritmos poderia então ser avaliada comparando as trajetórias descritas com a trajetória ótima gerada pelas equações (3.11), (3.12) e (3.17). É importante ressaltar que, ao contrário da trajetória numérica ótima, obtida selecionando-se curvas de aprendizado de passo fixo, a trajetória ótima analítica e as trajetórias dos algoritmos de passo variável *não estão contidas na superfície paramétrica*, mas esta sugere que a evolução dos algoritmos de passo variável pode ser caracterizada por trajetórias no espaço. Todavia, a dificuldade de visualização no espaço tridimensional torna a comparação das trajetórias pouca conclusiva, complicando a avaliação dos algoritmos.

As trajetórias espaciais, entretanto, possuem representação nos planos $[\xi, n]$, $[\mu, n]$ e $[\xi, \mu]$, que compõem o triedro $[\xi, \mu, n]$. Desta forma, pode-se contornar o problema da visualização, comparando as curvas de desempenho dos algoritmos com as curvas ótimas teóricas em cada um destes planos.

3.4. Caracterização Paramétrica de Algoritmos de Passo Variável

Projetada nos planos $[\mu, n]$ e $[\xi, n]$, a trajetória ótima gera a seqüência ótima de passos $\mu_{opt}(n)$ e a curva de aprendizado ótima $\xi_{opt}(n)$ correspondente. Caracterizando a evolução ótima, estas curvas fornecem uma ferramenta importante para a avaliação do desempenho de algoritmos adaptativos de passo variável. Em avaliações típicas de algoritmos apresentadas na literatura [16], [17], [21], [27], as curvas características dos algoritmos são comparadas entre si nos dois planos clássicos citados, sem qualquer resultado teórico que sirva como medida absoluta de desempenho. Este procedimento pode levar a conclusões erradas, como será mostrado em alguns exemplos do Capítulo 4.

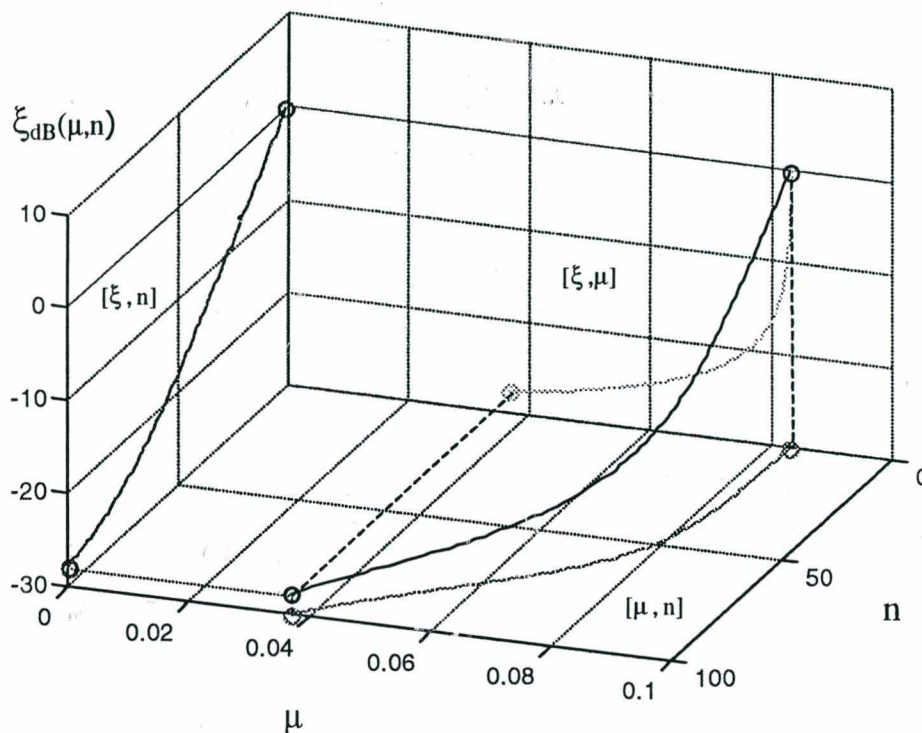


Fig. 3.3 - A trajetória ótima no espaço e as projeções nos planos

Outra projeção até então inexplorada, em se tratando de algoritmos de passo variável, situa-se no plano $[\xi, \mu]$. A curva resultante é implícita em n , capturando informações simultâneas do erro médio quadrático e do passo. A Figura 3.3 ilustra as projeções possíveis da trajetória ótima tridimensional.

Uma vez definido o problema, através da matriz R , do erro mínimo ξ_{\min} e da inicialização $K(0)$, seqüências de passos iguais, naturalmente, geram trajetórias espaciais iguais, e portanto projeções idênticas no plano $[\xi, \mu]$. Seqüências de passos diferentes geram trajetórias espaciais diferentes, mas poderiam, eventualmente, produzir curvas de aprendizado iguais (plano $[\xi, n]$), dada a dependência quadrática da equação de evolução do erro em função do passo (equação 3.14). Considerando que na equação 3.14 o termo quadrático em μ é sempre positivo [4], para cada escolha não-ótima do passo, existe outra equivalente que determina a evolução do erro para o mesmo patamar na iteração seguinte. No entanto, as projeções correspondentes no plano $[\xi, \mu]$ seriam

diferentes, dado que neste plano são combinadas informações simultâneas do erro e do passo - *não existem trajetórias espaciais distintas que resultem em projeções iguais no plano* $[\xi, \mu]$.

Em se tratando de uma caracterização compacta e legítima, a representação da evolução dos algoritmos no plano $[\xi, \mu]$ será utilizada no restante do trabalho, sobre a qual será desenvolvida uma metodologia de avaliação e projeto da dinâmica e do regime permanente de algoritmos de passo variável, como mostra a próxima seção.

3.5. O Plano de Aprendizado

Conforme explanado, a evolução ótima do erro médio quadrático e a seqüência ótima de passos seriam suficientes para avaliar os algoritmos de passo variável. No entanto, observar o impacto da seqüência de passos gerada pelos algoritmos de controle sobre o comportamento do erro médio quadrático nem sempre é uma tarefa trivial, quando realizada separadamente nas curvas do erro médio quadrático e do passo como funções de n . A representação dos algoritmos no plano $[\xi, \mu]$ reúne em uma única curva a informação antes disponível em curvas separadas, caracterizando de maneira mais compacta e completa o comportamento dos algoritmos de passo variável. Assim, envolvendo a constante de aprendizado e a figura de mérito adjacente, o plano $[\xi, \mu]$ será referenciado no restante do trabalho como *plano de aprendizado*.

Tal como o regime transitório, o regime permanente também pode ser caracterizado graficamente no plano de aprendizado. Na superfície paramétrica da Figura 3.1 verifica-se que, na medida em que as iterações avançam, as curvas de aprendizado, traçadas para passos gradativamente diferentes, convergem para o valor de regime, visível apenas parcialmente devido ao número de iterações mostrado. Reunindo os pontos correspondentes ao regime permanente de cada curva de aprendizado, gera-se uma outra curva, que na condição limite ($n \rightarrow \infty$), relaciona o erro médio quadrático em regime com o passo correspondente, sendo passível de projeção no plano de aprendizado, portanto.

Na Figura 3.3 observa-se no plano ao fundo um primeiro esboço do plano de aprendizado, ao qual pode-se acrescentar a curva do regime permanente comentada no parágrafo anterior, permitindo descrever tanto a dinâmica quanto o regime permanente dos algoritmos de passo variável.

Uma vez que os algoritmos de passo variável geralmente determinam o passo de cálculo em função de alguma variável do processo adaptativo [16], [17], [18], [21], [25], [26], achou-se conveniente refletir os eixos do plano de aprendizado (plano $[\mu, \xi]$), explicitando o passo de cálculo como variável dependente do erro médio quadrático. A Figura 3.4 ilustra a situação.

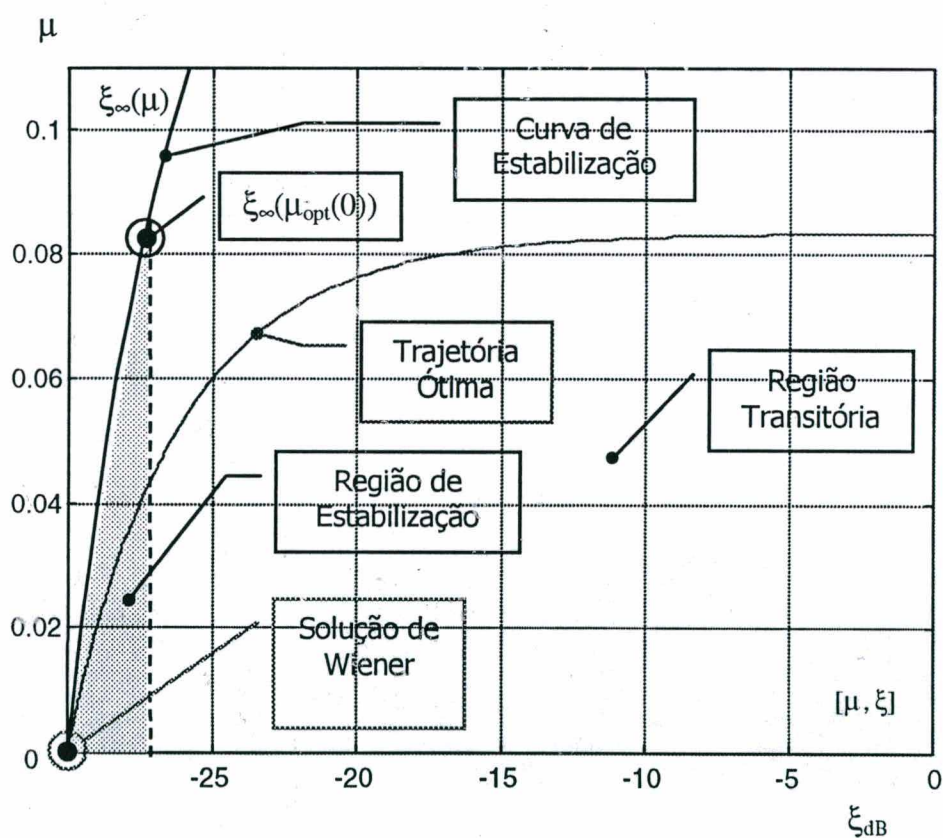


Fig. 3.4 – O plano de aprendizado

O plano de aprendizado reúne informações importantes para avaliação do desempenho de algoritmos adaptativos de passo variável. Estas características,

mostradas na Figura 3.4, são descritas a seguir.

O Regime Transitório e a Trajetória Ótima

A partir da inicialização, os algoritmos evoluem pelo plano de aprendizado descrevendo trajetórias que representam a etapa transitória do processo adaptativo. Assim, *a dinâmica dos algoritmos de passo variável pode ser representada pela trajetória descrita no plano de aprendizado.*

A trajetória ótima é o lugar geométrico dos pontos no plano $[\mu, \xi]$ que caracterizam a evolução ótima do algoritmo LMS segundo as expressões (3.11), (3.12) e (3.17). Os algoritmos adaptativos de passo variável apresentarão uma dinâmica eficiente na medida em que suas trajetórias características se aproximem da trajetória ótima no plano de aprendizado.

O Regime Permanente e a Curva de Estabilização

Conforme prossegue a adaptação, a proximidade do regime permanente tende a estabilizar as trajetórias, tornando-as estacionárias na condição limite. O que nas curvas clássicas seria representado por níveis constantes ao longo do número de iterações, no plano de aprendizado torna-se um ponto de estabilização. *O regime permanente de um algoritmo é representado por um ponto no plano de aprendizado.* O lugar geométrico dos pontos onde ocorre a estabilização para as diferentes trajetórias possíveis é definido como *curva de estabilização*. As trajetórias que se aproximam "transversalmente" à curva de estabilização, tendendo a interceptá-la, na condição limite ($n \rightarrow \infty$) tornar-se-ão estacionárias, com o ponto de estabilização representando o regime permanente, com desajuste correspondente determinado a partir da projeção do respectivo ponto no eixo ξ .

A Região de Estabilização

As proximidades da curva de estabilização compreendem uma região de convergência, em geral, lenta. A concordância com a trajetória ótima é determinante nesta região.

A Solução de Wiener

Representa o ponto de estabilização ótimo, onde o algoritmo atinge o erro médio quadrático mínimo em regime permanente dado pela solução de Wiener, com desajuste nulo portanto. Este é o ponto para o qual os algoritmos de passo variável convergiriam se o passo fosse reduzido de forma suficientemente gradual até $\mu(\infty) = 0$. No entanto, a maioria dos algoritmos de adaptação da constante de aprendizado determinam um valor não nulo para o passo em regime permanente, causando desajuste diferente de zero. Assim, o plano de aprendizado pode auxiliar no estudo da polarização dos algoritmos de controle do passo de cálculo, evidenciando os pontos de estabilização.

3.6. Geração e Utilização do Plano de Aprendizado

O plano de aprendizado fornece uma nova visualização da dinâmica e do regime permanente dos algoritmos de passo variável, a qual pode ser utilizada para avaliação e projeto dos mesmos.

Nesta seção será detalhada a metodologia de construção do plano, envolvendo a trajetória ótima, a curva de estabilização e um modelo analítico para geração de trajetórias de algoritmos de passo variável.

Também são apresentadas algumas heurísticas para auxiliar na avaliação de trajetórias no plano.

3.6.1. A Trajetória Ótima

Utilizando o passo ótimo (3.17) no modelo analítico do LMS (3.11) e (3.12), gera-se a curva de aprendizado ótima. Traçando esta em função do passo ótimo (3.17) descreve-se a trajetória ótima no plano de aprendizado.

O algoritmo utilizado para determinar a trajetória ótima, reunindo as equações (3.11), (3.12) e (3.17), é mostrado a seguir:

$W(0)$ (Dado)

$V(0) = W(0) - W_0$

$K(0) = E\{V(0)V(0)^T\}$

$n = 0, n_{\max}$

$$\xi_{\text{opt}}(n) = \xi_{\min} + \text{Tr}\{RK(n)\}$$

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

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

próximo n

(3.18)

3.6.2. A Curva de Estabilização

A curva de estabilização possui um papel importante na avaliação dos algoritmos adaptativos de passo variável, especificando o mínimo desajuste possível para cada valor do passo de adaptação e caracterizando assim o regime permanente em passo fixo.

Embora gerada para passo fixo e represente o regime permanente neste caso, as premissas estatísticas (H.3, H.4), que serão adotadas na derivação do modelo analítico do algoritmo LMS com passo variável na Seção 3.6.3, tornam a curva de estabilização legítima para descrever também o regime permanente do algoritmo LMS evoluindo com passo variável.

Conforme comentado na Seção 3.5, a curva de estabilização pode ser inferida na superfície de aprendizado, fazendo o número de iterações tender ao infinito. O lugar geométrico correspondente é definido matematicamente como:

$$\xi_{\infty}(\mu) = \xi(n, \mu)_{n \rightarrow \infty} \quad (3.19)$$

A expressão (3.19) também pode ser escrita em função do desajuste:

$$\xi_{\infty}(\mu) = \xi_{\min} + M(\mu) \cdot \xi_{\min} \quad (3.20)$$

A geração desta curva é simples. Numericamente pode-se obtê-la evoluindo o modelo analítico do algoritmo LMS em passo fixo para alguns passos distintos, durante um número suficiente de iterações que permita alcançar aproximadamente o regime permanente. Posteriormente interpola-se os pontos gerando a curva total.

A curva de estabilização também pode ser determinada analiticamente. A seguir apresenta-se as expressões para sinais de entrada brancos e correlacionados.

Ruído Branco

Quando o sinal de referência $x(n)$ é ruído branco com variância σ_x^2 , sua matriz de auto-correlação R é diagonal:

$$R = \sigma_x^2 \cdot I \quad (3.21)$$

Aplicando a definição (3.19) em (3.11), com R dada por (3.21), tem-se:

$$\xi_\infty(\mu) = \xi_{\min} + \sigma_x^2 \cdot \text{Tr}\{K(\infty)\} \quad (3.22)$$

Com o sinal de referência ruído branco, os elementos da diagonal principal da matriz $K(n)$ são idênticos e satisfazem a seguinte equação de diferenças [13]:

$$\phi(n+1) = [1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4(N+2)]\phi(n) + \mu^2\sigma_x^2\xi_{\min} \quad (3.23)$$

com:

$$\text{Tr}\{K(n)\} = N \cdot \phi(n) \quad (3.24)$$

A solução de (3.23) é direta:

$$\phi(n) = [1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4(N+2)]^n \phi(0) + \mu^2\sigma_x^2\xi_{\min} \sum_{j=0}^{n-1} [1 - 2\mu\sigma_x^2 + \mu^2\sigma_x^4(N+2)]^j \quad (3.25)$$

Utilizando a relação (3.24) em (3.22):

$$\xi_{\infty}(\mu) = \xi_{\min} + \sigma_x^2 \cdot N \cdot \phi(\infty) \quad (3.26)$$

Determinar a curva de estabilização reflete-se na solução de (3.26), que pode ser obtida de (3.25) fazendo o número de iterações $n \rightarrow \infty$. Para uma escolha apropriada do passo que garanta estabilidade, o termo em (3.25) que depende da inicialização relaxa a zero. O termo restante representa um somatório infinito dos termos de uma progressão geométrica (PG) cuja soma S_{∞} , facilmente demonstrável, é dada por:

$$S_{\infty} = \frac{a_1}{1-q} \quad (3.27)$$

onde $q < 1$ é a razão e a_1 o termo inicial. Aplicando (3.27) ao somatório infinito, resultante de (3.25) e (3.26), determina-se $\phi(\infty)$:

$$\phi(\infty) = \frac{\mu \xi_{\min}}{2 - \mu \sigma_x^2 (N+2)} \quad (3.28)$$

Substituindo (3.28) em (3.26) determina-se a curva de estabilização para o sinal de referência ruído branco:

$$\xi_{\infty}(\mu) = \xi_{\min} + \frac{\mu \sigma_x^2 N}{2 - \mu \sigma_x^2 (N+2)} \xi_{\min} \quad (3.29a)$$

Ou, equivalentemente:

$$\xi_{\infty}(\mu) = \frac{1}{1 - \frac{N}{2} \left(\frac{\mu \sigma_x^2}{1 - \mu \sigma_x^2} \right)} \cdot \xi_{\min} \quad (3.29b)$$

Sinais Correlacionados

Para o caso de sinais de referência correlacionados, a aplicação da definição da curva de estabilização no modelo analítico recursivo do algoritmo LMS é mais complexa, podendo-se utilizar expressões aproximadas para gerar a curva. Existem alguns resultados na literatura [1], [30].

Horowitz e Senne derivaram em [6] uma expressão para o desajuste em regime permanente que corrobora o resultado em (3.28) para $x(n)$ ruído branco. Por inferência, pode-se utilizá-la para o caso de sinais correlacionados:

$$M(\mu) = \frac{\eta(\mu)}{1 - \eta(\mu)} \quad (3.30)$$

Com:

$$\eta(\mu) = \sum_{i=1}^N \frac{\mu\lambda_i}{2 - 2\mu\lambda_i} \quad (3.31)$$

Substituindo (3.31) em (3.30) e posteriormente em (3.20), após alguma manipulação chega-se a uma expressão para a curva de estabilização no caso de sinais correlacionados:

$$\xi_{\infty}(\mu) = \frac{1}{1 - \frac{1}{2} \sum_{i=1}^N \frac{\mu\lambda_i}{1 - \mu\lambda_i}} \cdot \xi_{\min} \quad (3.32)$$

Pode-se também adotar a interpolação a partir de alguns pontos de estabilização gerados numericamente com o modelo passo fixo do algoritmo LMS, como sugerido no caso de $x(n)$ ruído branco.

3.6.3. Geração de Trajetórias de Algoritmos de Passo Variável no Plano de Aprendizado

Gerar trajetórias características de algoritmos de passo variável no plano de aprendizado consiste basicamente em traçar suas curvas de aprendizado em função das seqüências de passos correspondentes. Através de simulações Monte Carlo, pode-se representar qualquer algoritmo adaptativo de passo variável no plano de aprendizado.

Naturalmente, é bastante desejável gerar as trajetórias analiticamente, sendo necessários modelos analíticos para a evolução do passo de cálculo e do erro médio quadrático dos algoritmos. Ocorre que a análise de algoritmos de passo variável é ainda mais complexa do que no caso com passo fixo, sendo necessário adotar aproximações simplificadoras adicionais àquelas reunidas na teoria da independência.

Na literatura, a premissa de independência do passo $\mu(n)$ em relação a outras variáveis do processo adaptativo, $W(n)$, $e(n)$ e $X(n)$, é amplamente utilizada [8], [15], [16], [17], [21].

Em [16], o passo $\mu(n)$ é considerado independente do erro $e(n)$, do vetor de entradas $X(n)$ e do vetor de erro nos coeficientes $V(n)$.

Mathews e Xie observaram em [21] que o passo $\mu(n)$, o passo quadrático $\mu^2(n)$ e as variáveis $X(n)$, $W(n)$ e $e(n)$ podem ser assumidos não-correlacionados, especialmente quando as flutuações estatísticas do passo $\mu(n)$ são pequenas quando comparadas às flutuações em $X(n)$ e $e(n)$.

Assim, adicionalmente à teoria da independência serão utilizadas duas hipóteses simplificadoras presentes na literatura, que permitem estender o modelo analítico do algoritmo LMS com passo fixo para descrever algoritmos de passo variável, mantendo a trajetória ótima autêntica:

- O passo de cálculo $\mu(n)$ e seu valor quadrático $\mu^2(n)$ são estatisticamente independentes do vetor de entradas $X(n)$ e do vetor de erro nos coeficientes $V(n)$;

(H.3)

- O valor esperado do passo de cálculo quadrático é aproximadamente igual ao quadrado do valor médio do passo, $E[\mu^2(n)] \approx E^2[\mu(n)]$;

(H.4)

Essas hipóteses têm levado a bons resultados, como no caso de alguns algoritmos propostos na literatura e descritos sucintamente no Capítulo 2 [12], [14], [16], [17], [18], [22]. A segunda premissa pode ser relaxada caso se possua um modelo analítico para a evolução do valor médio do passo quadrático $E[\mu^2(n)]$ (ver Apêndice).

Utilizando a teoria da independência e as premissas adicionais citadas anteriormente, chega-se a um *modelo analítico para a evolução do erro médio quadrático do algoritmo LMS com passo variável* (Apêndice):

$$\xi(n) = \xi_{\min} + \text{Tr}\{RK(n)\} \quad (3.33)$$

$$K(n+1) = K(n) - E[\mu(n)] \cdot [RK(n) + K(n)R] + E^2[\mu(n)] \cdot [R \cdot \text{Tr}\{RK(n)\} + 2RK(n)R + R \cdot \xi_{\min}] \quad (3.34)$$

As expressões (3.33) e (3.34) descrevem o comportamento do erro médio quadrático dos algoritmos LMS de passo variável. Com uma inicialização pertinente e um modelo para a evolução média da seqüência de passos, pode-se caracterizá-los no plano de aprendizado.

3.6.4. A Região de Estabilização e Heurísticas para Avaliação de Trajetórias no Plano de Aprendizado

A avaliação dos algoritmos no plano de aprendizado é feita graficamente, comparando as trajetórias descritas com a trajetória ótima. Os algoritmos são avaliados de acordo com a região percorrida no plano, existindo regiões preferenciais, e outras que devem ser evitadas. É interessante derivar algumas heurísticas que guiem a avaliação das trajetórias dos algoritmos de passo variável.

Definição da Região de Estabilização

Nesta região a evolução dos algoritmos deve receber atenção especial. Representando a proximidade do regime permanente, a convergência nas proximidades da curva de estabilização tende a ser lenta, sendo útil definir graficamente limites para tais "proximidades", dado que a escolha do passo é mais crítica nessa região. A definição da região implica na escolha de um limite superior para o passo, com o erro médio quadrático associado. Uma boa definição para a região de estabilização é utilizar passo ótimo inicial $\mu_{opt}(0)$ dado por (3.17) e o erro em regime correspondente:

$$\xi_{\infty}(\mu)_{\mu=\mu_{opt}(0)} \quad (3.35)$$

Traçando-se uma reta suporte, paralela ao eixo dos passos (Reta tracejada na Figura 3.4), que passe pelo ponto correspondente à estabilização com passo fixo ótimo inicial $\mu_{opt}(0)$ e ao erro associado, delimita-se o plano em região de estabilização e região transitória, esta última percorrida durante as etapas iniciais da adaptação. Justifica-se tal escolha pressupondo-se que os algoritmos sejam inicializados de maneira ótima, com a seqüência de passos situada abaixo deste valor de passo, dado que a seqüência ótima de passos e a curva de aprendizado associada são sempre decrescentes a partir da inicialização,

uma vez garantida a estabilidade do algoritmo. A *região de estabilização* compreende a área entre a curva de estabilização e a reta suporte (região hachurada na Figura 3.4).

Heurísticas para Avaliação das Trajetórias

Definidas as regiões do plano e as curvas representativas do regime permanente e transitório, pode-se definir algumas heurísticas para auxiliar na avaliação e, posteriormente, no projeto de algoritmos de passo variável (Capítulo 4):

- A região de estabilização engloba uma área crítica, na qual a estabilização já se faz possível, dependendo da localização da trajetória, determinando uma área com baixas taxas de convergência. Trajetórias relativamente distantes da trajetória ótima na região transitória, mas que se aproximem da curva ótima na região de estabilização podem representar uma boa convergência;
- A curva de estabilização representa o lugar geométrico do conjunto de pontos para o qual a taxa de convergência é nula (regime permanente). Trajetórias próximas a ela devem ser evitadas, além de representar uma diminuição da taxa de convergência, aumenta-se a possibilidade de polarização do passo de cálculo e uma correspondente estabilização com desajustes elevados;
- A região inferior do plano de aprendizado, próxima ao eixo ξ , deve ser evitada, pois representa uma região de dinâmica lenta (μ pequeno);
- Embora implícito em n , o plano de aprendizado ainda preserva alguma informação temporal da adaptação. É possível avaliar os algoritmos para um dado número de iterações pré-estabelecido.

3.7. Conclusão

A grande problemática dos algoritmos adaptativos de passo variável é o ajuste de uma seqüência de passos adequada. Visando uma formalização da abordagem "empírica" presente na literatura, desenvolveu-se um modelo analítico para a evolução ótima do algoritmo LMS, em (3.11), (3.12) e (3.17), definindo os objetivos dos algoritmos de passo variável existentes.

Foi mostrado que os algoritmos adaptativos evoluindo em passo variável podem ser caracterizados por trajetórias no espaço $[\xi, \mu, n]$ que, projetadas nos planos componentes, $[\xi, n]$, $[\mu, n]$ e $[\xi, \mu]$, geram respectivamente a curva de aprendizado, a seqüência de passos e uma terceira curva, também característica, que reúne em si a informação disponível nas duas outras curvas. A curva resultante no plano $[\xi, \mu]$ representa uma trajetória implícita em n , descrevendo o comportamento do algoritmo adaptativo da inicialização ao regime permanente.

Representando a evolução ótima e o regime permanente do algoritmo LMS através de lugares geométricos característicos no plano de aprendizado $[\xi, \mu]$, desenvolveu-se uma metodologia de avaliação e projeto de algoritmos de passo variável derivados do LMS. *Os algoritmos são avaliados de acordo com a região percorrida por suas trajetórias características no plano de aprendizado.*

Embora as trajetórias características possam ser geradas por simulações Monte Carlo, o modelo analítico do LMS passo fixo foi estendido para o caso com passo variável, permitindo determiná-las analiticamente.

O capítulo a seguir aplica a metodologia desenvolvida a alguns algoritmos existentes. É mostrado que a nova metodologia pode ser empregada tanto para avaliação quanto para projetos de algoritmos adaptativos.

Aplicações da Metodologia Desenvolvida

4.1. Introdução

Uma escolha dentre os inúmeros algoritmos existentes requer a consideração de características como complexidade computacional, robustez, e comportamento dinâmico para um dado desajuste em regime permanente. A complexidade computacional é ditada pela estrutura de adaptação do passo, algumas mais complexas [1], [8], [22], outras mais simples [16], [17], [18]. Maior complexidade, no entanto, nem sempre se traduz em melhor desempenho, como será demonstrado.

Neste capítulo aplica-se a metodologia derivada no Capítulo 3, revelando a potencialidade dos algoritmos de filtragem, que adaptam o passo recursivamente baseando-se em alguma variável do processo adaptativo. Quando corretamente projetados, podem apresentar desempenho superior a seus concorrentes mais complexos.

Dedicou-se particular atenção ao algoritmo proposto em 1992 por Kwong e Johnston em [16]. Sua simplicidade e facilidade de projeto pela metodologia desenvolvida no Capítulo 3 o tornam uma excelente escolha, principalmente em ambientes onde o sinal de referência é ruído branco, como é o caso de muitas aplicações de identificação de sistemas. Nestas condições, as características de projeto tendem a ser conservadas, mesmo em ambientes sujeitos a certos tipos de não-estacionaridades.

Demonstra-se também que, conhecidas algumas propriedades estatísticas dos sinais, o projeto da dinâmica dos algoritmos de passo variável torna-se uma alternativa profícua e viável, quando comparada aos algoritmos que realizam a busca automática do passo com o proceder da adaptação [1], [20], [21], [22].

4.2. Projeto de Algoritmos de Passo Variável para Identificação de Sistemas

Os algoritmos de controle do passo possuem parâmetros próprios que regulam a evolução da constante de adaptação durante o processo de aprendizado do filtro. Inserindo nos modelos analíticos as informações estatísticas do ambiente no qual está inserido o filtro adaptativo, pode-se prever o comportamento do algoritmo adaptativo utilizado, ajustando os parâmetros de controle para otimizar a convergência. Projetar algoritmos na metodologia proposta corresponde a *corrigir suas trajetórias no plano de aprendizado*, otimizando a dinâmica para o desajuste necessário em regime permanente.

Para utilizar os modelos analíticos derivados no Capítulo 3, além do modelo para evolução do valor médio do passo, é preciso inicializá-los coerentemente com a natureza do problema prático. Isso reflete-se num valor inicial para o passo de cálculo e para a matriz $K(n)$ em (3.12) :

$$\mu(0) = \mu_0 \quad (4.1)$$

$$K(0) = E[V(0) V^T(0)] \quad (4.2)$$

Considerando o vetor nulo como estimativa inicial do vetor dos coeficientes do filtro, a inicialização mais usada em sistemas práticos, tem-se:

$$W(0) = [0]_{N \times 1} \quad (4.3)$$

Aplicando-se a definição (2.24) e a inicialização (4.3) em (4.2) resulta:

$$K(0) = W_0 W_0^T \quad (4.4)$$

A expressão (4.4) revela um problema concreto em situações de projeto: a inicialização do modelo analítico implica na utilização de informações não conhecidas a priori. A obtenção da solução de Wiener é, paradoxalmente, o próprio objetivo da adaptação. Entretanto, para o caso particular e importante em

que $x(n)$ é ruído branco, pode-se contornar o problema da inicialização. Este caso corresponde, por exemplo, a uma das mais importantes aplicações de filtragem adaptativa: a *identificação de sistemas*. A Figura 4.1 apresenta o diagrama em blocos do problema de identificação de sistemas [3].

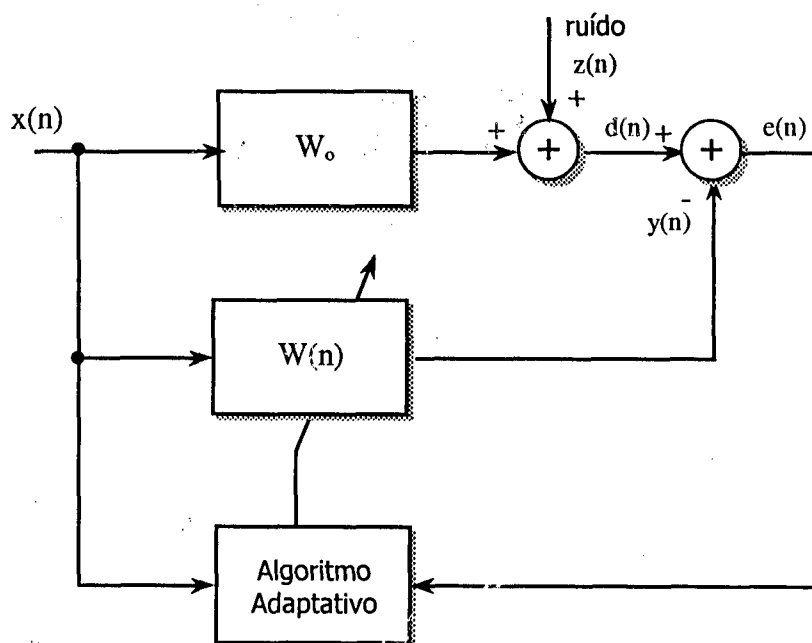


Fig. 4.1 – O problema de Identificação de Sistemas

Considerando $R = \sigma_x^2 \cdot I$ (sinal de referência $x(n)$ ruído branco) as expressões (3.11), (3.12) e (3.17) resultam em:

$$\xi(n) = \xi_{\min} + \sigma_x^2 \cdot \text{Tr}\{K(n)\} \quad (4.5)$$

$$\mu_{\text{opt}}(n) = \frac{\text{Tr}\{K(n)\}}{(N+2) \cdot \sigma_x^2 \cdot \text{Tr}\{K(n)\} + N \cdot \xi_{\min}} \quad (4.6)$$

$$K(n+1) = K(n) - 2\sigma_x^2 \mu \cdot K(n) + \mu^2 \cdot [\sigma_x^4 I \cdot \text{Tr}\{K(n)\} + 2\sigma_x^4 K(n) + \sigma_x^2 \xi_{\min} I] \quad (4.7)$$

As equações (4.5) e (4.6) necessitam apenas da informação do traço da matriz $K(n)$, que pode ser obtido da expressão (4.7):

$$\text{Tr}\{K(n+1)\} = \text{Tr}\{K(n)\} - 2\sigma_x^2 \mu \cdot \text{Tr}\{K(n)\} + \mu^2 \cdot [\sigma_x^4 N \cdot \text{Tr}\{K(n)\} + 2\sigma_x^4 \text{Tr}\{K(n)\} + \xi_{\min} \sigma_x^2 N]$$

$$(4.8)$$

De (4.5), (4.6) e (4.8) percebe-se que quando o sinal de referência é ruído branco, a evolução do modelo analítico depende apenas do traço da matriz $K(n)$, que é uma quantidade escalar. Inicializar o modelo implica então em determinar $\text{Tr}\{K(0)\}$. Inicializando o filtro adaptativo por (4.3) e considerando (4.4), tem-se:

$$\text{Tr}\{K(0)\} = \text{Tr}\{W_0 W_0^T\} = \text{Tr}\{W_0^T W_0\} = |W_0|^2 \quad (4.9)$$

Pode-se então inicializar o modelo a partir de uma estimativa da norma da solução de Wiener. Essa informação não é problemática e pode ser estimada dos dados no caso de identificação de sistemas. Nesse caso, a solução de Wiener será igual à resposta ao impulso do sistema a identificar. Da Figura 4.1:

$$d(n) = X^T(n) W_0 + z(n) \quad (4.10)$$

onde o ruído de medição $z(n)$ tem média nula e é independente de qualquer outro sinal.

Elevando a expressão (4.10) ao quadrado:

$$d^2(n) = d^T(n)d(n) = W_0^T X(n)X^T(n) W_0 + 2z(n)X^T(n)W_0 + z^2(n) \quad (4.11)$$

Tomando o valor esperado de (4.11) e considerando que o ruído de medição $z(n)$ é independente dos outros sinais:

$$\sigma_d^2 = E[d^2(n)] = E[d^T(n)d(n)] = W_0^T E[X(n)X^T(n)]W_0 + E[z^2(n)] = W_0^T R W_0 + \sigma_z^2 \quad (4.12)$$

Para $x(n)$ ruído branco,

$$\sigma_d^2 = W_0^T \cdot \sigma_x^2 \cdot I \cdot W_0 + \sigma_z^2 = \sigma_x^2 \cdot |W_0|^2 + \sigma_z^2 \quad (4.13)$$

$$\therefore |W_0|^2 = \frac{\sigma_d^2 - \sigma_z^2}{\sigma_x^2} \quad (4.14)$$

A inicialização do modelo para $x(n)$ ruído branco fica então:

$$\text{Tr}\{K(0)\} = \frac{\sigma_d^2 - \sigma_z^2}{\sigma_x^2} \quad (4.15)$$

onde a variância σ_x^2 é conhecida e as quantidades σ_z^2 e σ_d^2 podem ser estimadas da seqüência de dados. Além disso, de (2.12), (2.13) e (4.12) verifica-se que $\xi_{\min} = \sigma_z^2$, para o caso de identificação de sistemas, com o filtro de Wiener e o filtro adaptativo apresentando a mesma ordem (Figura 4.1).

As equações (4.5), (4.6) e (4.8) representam o modelo analítico da evolução ótima do algoritmo LMS descrito em (3.18) e particularizado para o sinal de referência $x(n)$ ruído branco. Empregando a relação (3.24) em (4.5), em (4.6) e em (4.8), e utilizando a equação recursiva (3.23), pode-se simplificar ainda mais o modelo, transformando a evolução matricial das equações numa recursão escalar, menos complexa computacionalmente. Reunindo as expressões resultantes na forma de algoritmo e considerando que o filtro adaptativo seja inicializado por (4.3), tem-se:

$$\begin{aligned} \phi_{\text{opt}}(0) &= \frac{\sigma_d^2 - \sigma_z^2}{N\sigma_x^2} ; \\ n &= 0, n_{\text{max}} \\ \xi_{\text{opt}}(n) &= \sigma_z^2 + N \cdot \sigma_x^2 \cdot \phi_{\text{opt}}(n) \\ \mu_{\text{opt}}(n) &= \frac{\phi_{\text{opt}}(n)}{(N+2) \cdot \sigma_x^2 \cdot \phi_{\text{opt}}(n) + \sigma_z^2} \\ \phi_{\text{opt}}(n+1) &= [1 - 2\sigma_x^2 \mu_{\text{opt}}(n) + (N+2)\sigma_x^4 \mu_{\text{opt}}^2(n)] \phi_{\text{opt}}(n) + \mu_{\text{opt}}^2(n) \cdot \sigma_x^2 \cdot \sigma_z^2 \\ &\text{next } n \end{aligned} \quad (4.16)$$

Para o caso geral, o modelo analítico do algoritmo LMS envolve uma equação recursiva matricial, dada por (3.12), cuja inicialização necessita da informação completa acerca da solução de Wiener, e não apenas de sua norma, como em (4.15). Isso é descrito em [1] como *direcionalidade* do algoritmo LMS. Neste caso, a metodologia desenvolvida pode ser empregada na avaliação de

desempenho de algoritmos de passo variável, onde todos os dados podem ser estipulados a priori, num contexto de simulação.

Uma questão que emerge é a inicialização do passo nos algoritmos utilizados no controle do mesmo. Como inicializar o passo de adaptação? Embora seja arbitrária, esta escolha pode levar a diferentes desempenhos, uma vez que a adaptação é um processo dinâmico e, portanto, sofre influência das condições iniciais. A próxima seção aborda este problema.

4.2.1. Passo Fixo Ótimo LMS – Um Novo Resultado

Em situações em que velocidade de convergência é importante e deseja-se operar com passo fixo, torna-se necessária uma otimização do passo de adaptação. Como mostra a superfície paramétrica na Figura 3.1, passos “grandes” fornecem as maiores taxas iniciais de convergência [1], [17].

Existem vários resultados na literatura no que concerne a otimização da velocidade em passo fixo. De acordo com a abordagem adotada, chega-se a resultados diferentes para o passo ótimo a ser utilizado.

Em [12] e [31] é fornecido um valor ótimo para o passo em se tratando de velocidade de convergência do erro médio quadrático:

$$\mu_* = \frac{1}{N\sigma_x^2} \quad (4.17)$$

Horowitz e Senne em [6] e Feuer e Weinstein em [11] chegaram a um mesmo resultado partindo de abordagens distintas, com $x(n)$ ruído branco:

$$\mu_{**} = \frac{1}{(N+2)\sigma_x^2} \quad (4.18)$$

As expressões clássicas levam em consideração apenas a estatística do sinal de referência. Não obstante, é razoável esperar que informações estatísticas de outras variáveis envolvidas influenciem na velocidade de convergência do algoritmo. Esta expectativa realmente é verificada quando se analisa a expressão derivada para a seqüência de passos ótima (4.6). Escolhendo-se o valor do passo

fixo igual ao passo ótimo inicial da operação em passo variável tem-se:

$$\mu_{\text{fopt}} = \mu_{\text{opt}}(0) \quad (4.19)$$

$$\mu_{\text{fopt}} = \frac{\text{Tr}\{\mathbf{K}(0)\}}{(N+2) \cdot \sigma_x^2 \cdot \text{Tr}\{\mathbf{K}(0)\} + N \cdot \sigma_z^2} \quad (4.20)$$

Aplicando (4.15) em (4.20):

$$\mu_{\text{fopt}} = \frac{\sigma_d^2 - \sigma_z^2}{(N+2) \sigma_x^2 (\sigma_d^2 - \sigma_z^2) + N \sigma_x^2 \sigma_z^2} \quad (4.21a)$$

$$\mu_{\text{fopt}} = \frac{\text{SNR}}{(N+2) \sigma_x^2 \cdot \text{SNR} + N \sigma_x^2} \quad (4.21b)$$

onde $\text{SNR} = \frac{\sigma_d^2 - \sigma_z^2}{\sigma_z^2}$ é a razão sinal-ruído na saída da planta ("Signal-to-Noise Ratio").

A expressão (4.21b) é uma alternativa a (4.17) e (4.18), maximizando a *taxa inicial de convergência do erro médio quadrático*. Adicionalmente ao fato de considerar mais informações sobre o contexto de adaptação, o novo resultado possui algumas peculiaridades quando comparado aos resultados clássicos. A escolha do passo pela nova expressão conduz a valores conservativos, *além de otimizar a velocidade de convergência do erro médio quadrático, propicia maior robustez e menores desajustes em regime permanente*.

Para relação sinal-ruído (SNR) grande ($\text{SNR} \rightarrow \infty$), a expressão (4.21b) aproxima-se de (4.18). Em situações práticas, para as quais os filtros possuem dimensão considerável ($N \gg 2$), a expressão (4.18) recai em (4.17), a expressão (4.21b), no entanto, resulta em:

$$\mu_{\text{fopt}} \approx \frac{1}{N \sigma_x^2} \cdot \frac{\text{SNR}}{\text{SNR} + 1} \quad (4.22a)$$

$$\mu_{\text{fopt}} \approx \frac{1}{N\sigma_x^2} \cdot \frac{1}{1 + \frac{1}{\text{SNR}}} \quad (4.22b)$$

Claramente nota-se um fator de redução, em relação às expressões (4.17) e (4.18), que é função da razão sinal-ruído. Dada a natureza passiva da maioria dos sistemas físicos, os quais freqüentemente atenuam o sinal de entrada, o valor ótimo do passo dado por (4.22) tende a ser reduzido, comparado com (4.17) ou (4.18).

Para filtros com ordem elevada ($N \gg 2$) e grandes razões sinal-ruído, as expressões (4.17), (4.18), (4.22) tendem ao mesmo valor, dado em (4.17).

O novo resultado pode ainda ser aproveitado na implementação de algoritmos de passo variável, em que se faz necessário estipular um intervalo de truncamento que assegure simultaneamente a convergência, responsabilidade do limite superior (μ_{max}), e boas características de regime permanente - nível desejado de desajuste em ambientes estacionários e capacidade de rastreamento em casos não-estacionários - características determinadas pelo limite inferior (μ_{min}) [1], [7], [16], [17]. Também é aconselhável inicializar-se os algoritmos de adaptação do passo com este valor, uma vez que a inicialização pode influenciar a dinâmica de algoritmos adaptativos.

Algumas simulações foram realizadas, ilustrando a aplicação do novo resultado.

Exemplos

Nos exemplos a seguir são analisados os projetos do passo fixo ótimo pela nova expressão (4.21b) e pelo resultado mais utilizado na literatura, a expressão (4.17). O caso utilizado é o de identificação de sistemas, com o sinal de referência $x(n)$ ruído branco gaussiano de variância unitária, excitando o filtro adaptativo e a planta sob identificação. A planta desconhecida representa um sistema passivo, atenuando o sinal de referência $x(n)$, o que resulta num sinal

desejado com potência mais próxima ao nível do ruído de medição. A expressão (4.17) é indiferente à redução da variância do sinal desejado, fornecendo um valor do passo dependente apenas da variância do sinal de entrada e da ordem do filtro adaptativo. Para otimizar a velocidade de convergência, a nova expressão (4.21b) propõe a redução do valor do passo em função da relação sinal-ruído SNR na saída da planta. A planta utilizada em cada um dos exemplos simulados é

$$W_o^T = \frac{(\sigma_d^2 - \sigma_z^2)^{1/2}}{\sqrt{N}} \cdot [1; 1; \dots; 1], \text{ com o desvio padrão } (\sigma_d^2 - \sigma_z^2)^{1/2} = (\text{SNR} \cdot \sigma_z^2)^{1/2}$$

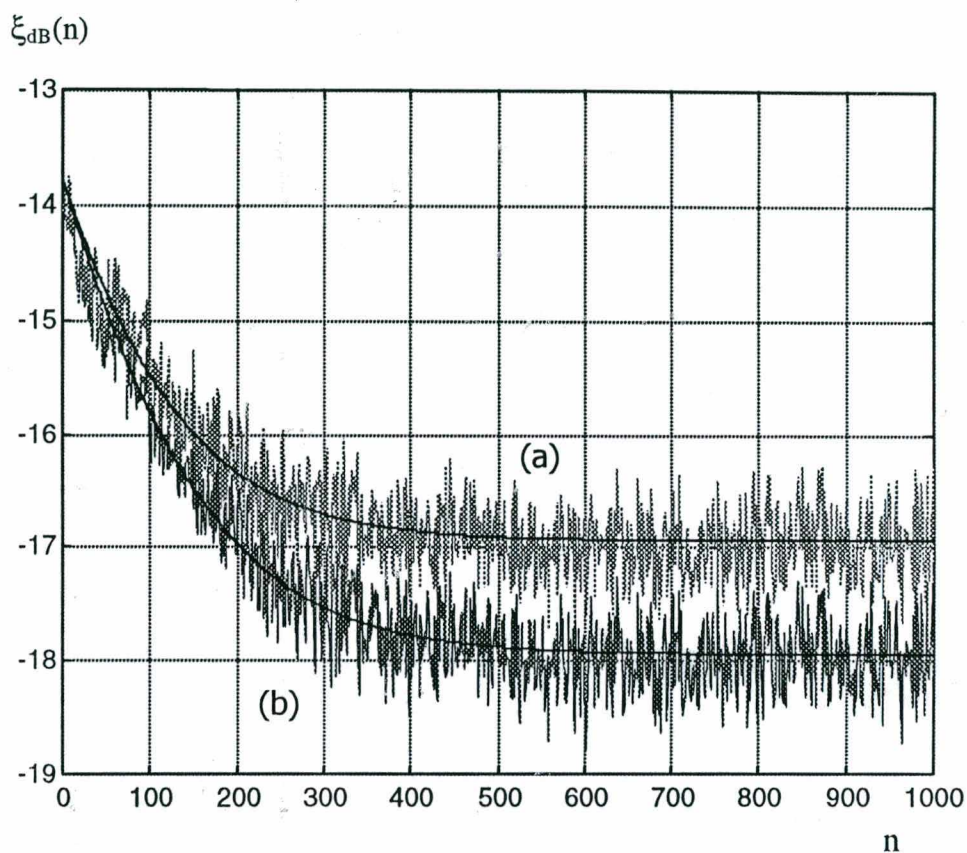
determinado de acordo com a razão sinal-ruído desejada. O ruído de medição utilizado nos exemplos é gaussiano com média nula e possui variância $\sigma_z^2 = 10^{-2}$. A tabela I reúne os parâmetros utilizados nos exemplos apresentados.

Tabela I – Dados de Simulação $\left(\text{SNR} = \frac{\sigma_d^2 - \sigma_z^2}{\sigma_z^2} \right)$

Ex	N	SNR	SNR _{dB}	μ_* (4.17)	μ_{fopt} (4.21b)	$ \xi_*(\infty) - \xi_{\text{fopt}}(\infty) _{\text{dB}}$
1	100	3,162	5	0,01	0,0075	1
2	100	10	10	0,01	0,0089	0,5
3	10	100	20	0,1	0,0826	1

Nos dois primeiros exemplos, utilizou-se um filtro com ordem relativamente elevada (N=100), a situação mais comum na prática.

No primeiro exemplo, além de acelerar a convergência, o passo ótimo dado por (4.21b) apresenta um erro médio quadrático em regime de aproximadamente *1 dB abaixo do resultado clássico* (equação 4.17). A Figura 4.2 apresenta as curvas do exemplo 1, cujos parâmetros são descritos na Tabela I.



*Fig. 4.2 – Exemplo 1. Curvas contínuas: previsões teóricas
(a) Resultado clássico. (b) Novo resultado
Simulação Monte Carlo com 500 realizações*

A Figura 4.3 corresponde ao exemplo 2 e apresenta o comportamento do erro médio quadrático para uma razão sinal-ruído mais elevada. Neste caso ainda pronuncia-se uma diferença apreciável a favor do novo resultado. O passo dado por (4.21b) acelera a convergência e apresenta um erro em regime permanente cerca de $0,5 \text{ dB}$ abaixo do erro correspondente ao passo escolhido pela expressão (4.17), como mostra a Tabela I.

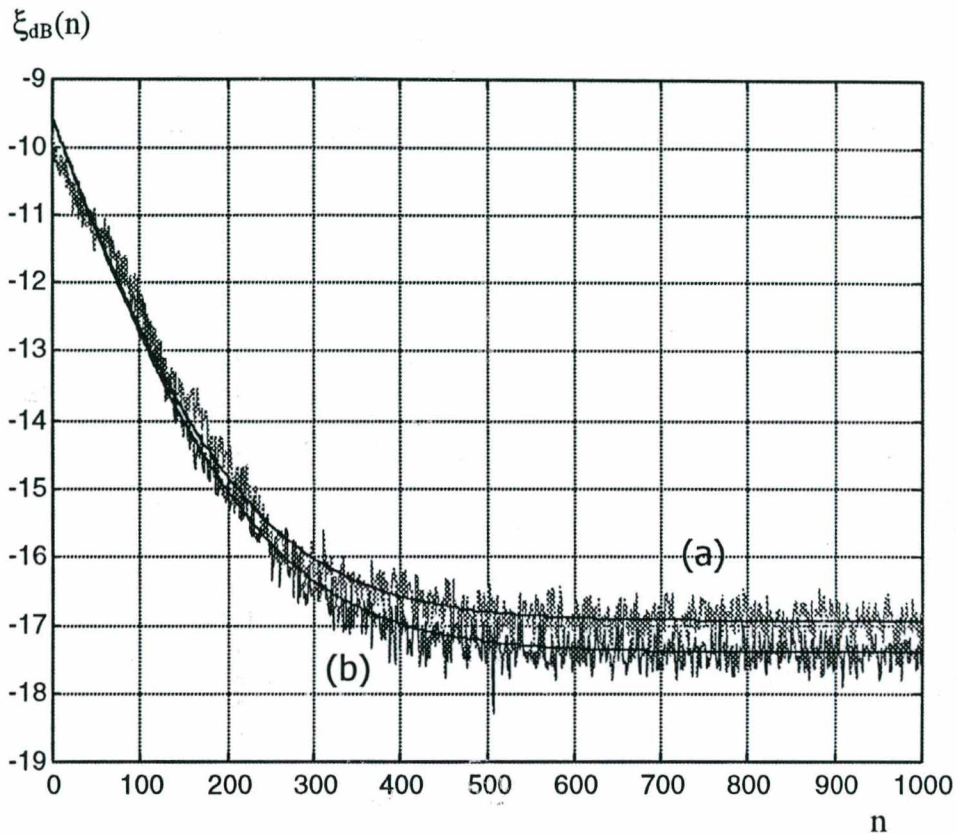


Fig. 4.3 – Exemplo 2. Curvas contínuas: previsões teóricas
 (a) Resultado clássico. (b) Novo resultado.
 Simulação Monte Carlo com 1000 realizações

O exemplo 3 foi simulado com a razão sinal-ruído relativamente elevada (20dB), em que a planta não atenua o sinal de referência, correspondendo a um filtro de Wiener com norma unitária. As expressões (4.17) e (4.21b) novamente fornecem valores diferentes para o passo ótimo, traduzindo-se em desempenhos diferentes, como mostra a Figura 4.4. Com um erro médio quadrático em regime permanente em torno de 1 dB *abaixo* do correspondente ao passo ótimo clássico, o novo passo ótimo melhorou o desempenho em regime permanente, além de acelerar a dinâmica do filtro adaptativo.

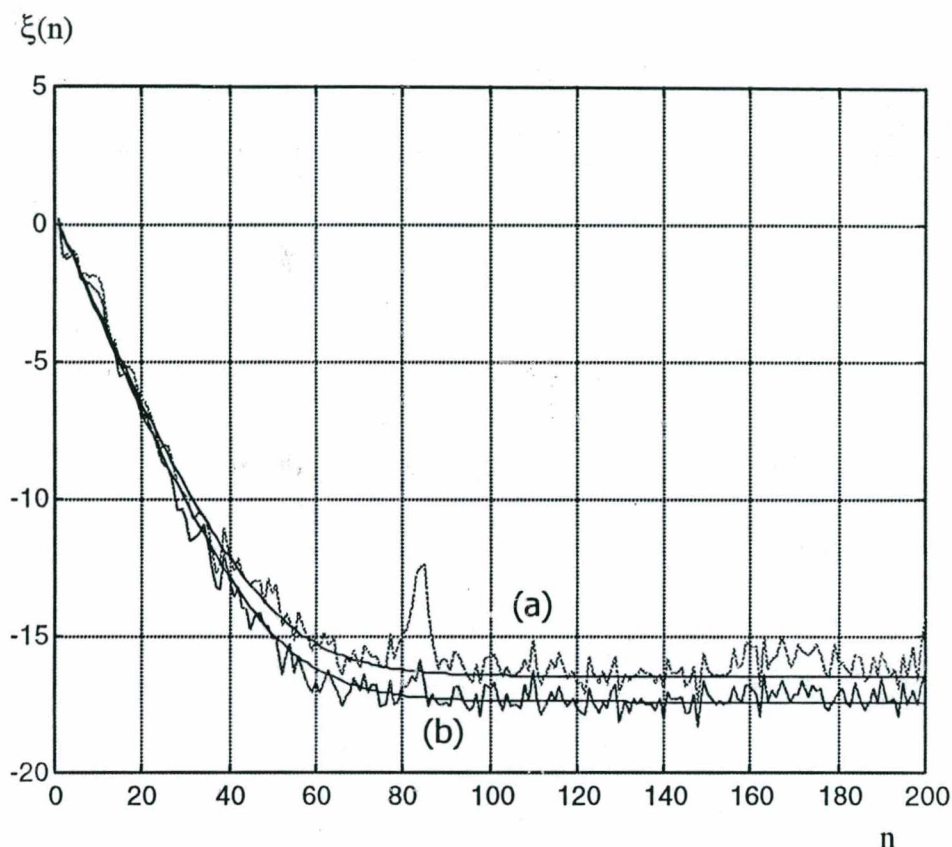


Fig. 4.4 – Exemplo 3. Curvas contínuas: previsões teóricas. (a) Resultado clássico. (b) Novo resultado. Simulação Monte Carlo com 200 realizações

4.2.2. Um Algoritmo Simples e Projetável: VSS

Em 1992, Kwong e Johnston propuseram um algoritmo simples de passo variável da família LMS. Denominado originalmente "Variable Step-Size LMS Algorithm", ou simplesmente "VSS", o algoritmo proposto ajusta o passo pelo erro de estimação quadrático.

Nada mais natural que utilizar a própria figura de mérito do algoritmo LMS para controlar a adaptação do passo. Vinculando-se a constante de adaptação ao erro quadrático de predição, grandes erros determinarão o aumento do passo de cálculo, conferindo velocidade ao algoritmo no início da adaptação. Conforme a adaptação prossegue, erros de predição pequenos reduzirão o passo, resultando em desajustes pequenos em regime permanente [16].

O algoritmo de adaptação do passo é simples e apresenta a grande

vantagem de ser *independente da ordem do filtro*, resultando em um aumento irrisório no custo computacional, relativo ao caso típico com passo fixo [16]. Basicamente, o algoritmo filtra o erro quadrático instantâneo implementando uma equação recursiva de primeira ordem:

$$\mu(n+1) = \alpha \cdot \mu(n) + \gamma \cdot e^2(n) \quad (4.23)$$

Valores limites $[\mu_{\min}, \mu_{\max}]$ determinam o intervalo de truncamento, necessário para manter o algoritmo estável (μ_{\max}) e permitir o rastreamento da solução ótima quando num ambiente não-estacionário (μ_{\min}) [1], [7], [16].

As constantes α e γ devem ser ajustadas de acordo com o comportamento desejado do algoritmo. Os autores derivaram uma expressão para o desajuste em regime permanente que orienta a escolha destas constantes, válida para o caso estacionário e para desajustes pequenos [16]:

$$M = \frac{1 - \left[1 - \frac{2(3-\alpha)\gamma\sigma_z^2}{1-\alpha^2} \text{Tr}[\mathbf{R}] \right]^{1/2}}{1 + \left[1 - \frac{2(3-\alpha)\gamma\sigma_z^2}{1-\alpha^2} \text{Tr}[\mathbf{R}] \right]^{1/2}} \quad (4.24)$$

Escolhendo-se α , determina-se a constante γ necessária para atingir o desajuste desejado M , dado por (4.24). Seguindo essa política, sugere-se em [16] a utilização de $\alpha = 0,97$, uma vez que este valor “parece funcionar bem nas simulações realizadas”, ajustando γ pela expressão fornecida. Ocorre que não existe evidências de que $\alpha = 0,97$ seja um valor apropriado em qualquer situação.

Duas constantes, dois graus de liberdade, duas especificações de projeto: dinâmica e regime permanente. Modificando convenientemente o procedimento de projeto proposto pelos autores, é possível ajustar a dinâmica e o regime permanente do algoritmo VSS, com o auxílio do plano de aprendizado e da expressão (4.24), respectivamente.

A expressão (4.24) pode ser manipulada explicitando γ em função do desajuste desejado M e da constante α , além dos parâmetros estatísticos:

$$\gamma = \left[1 - \left(\frac{1-M}{1+M} \right)^2 \right] \cdot \frac{(1-\alpha^2)}{2(3-\alpha)\text{Tr}\{\mathbf{R}\}\sigma_z^2} \quad (4.25)$$

Com o desajuste desejado M e uma *estimativa inicial* para α , determina-se a constante γ correspondente através de (4.25). O conjunto de constantes selecionado gera uma trajetória característica no plano de aprendizado, que pode ser corrigida refinando a escolha inicial de α sob orientação da trajetória ótima e das heurísticas descritas no Capítulo 3. Para isto, é preciso gerar as trajetórias características.

Conforme descrito no Seção 3.6.3, para caracterizar o algoritmo VSS no plano paramétrico de aprendizado é necessário um modelo do passo de cálculo médio. Assumindo que o passo situe-se dentro do intervalo de truncagem [16] e tomando o valor esperado de (4.23), obtém-se facilmente um modelo recursivo para o valor médio do passo:

$$E[\mu(n+1)] = \alpha E[\mu(n)] + \gamma E[e^2(n)] \quad (4.26)$$

$$\mu_{\text{VSS}}(n+1) = \alpha \mu_{\text{VSS}}(n) + \gamma \xi(n) \quad (4.27)$$

Empregando o modelo para o passo de cálculo médio (4.27) no modelo analítico do algoritmo LMS com passo variável, apresentado em (3.33) e (3.34), descreve-se o comportamento do algoritmo VSS, permitindo representá-lo analiticamente no plano de aprendizado. Dado o desajuste de projeto M , o conjunto de constantes α_{cor} e γ_{cor} que determina a maior proximidade entre a trajetória característica e a trajetória ótima, segundo as heurísticas da Seção 3.6.4, otimiza o comportamento do algoritmo VSS, *corrigindo* sua dinâmica enquanto mantém o desajuste em regime permanente de projeto.

Utilizando o modelo descrito em (4.16) gera-se a trajetória ótima, necessária para avaliação da dinâmica do algoritmo VSS. A expressão (3.29b) descreve a curva de estabilização, caracterizando o regime permanente das trajetórias possíveis do algoritmo VSS.

Reunindo as expressões necessárias para gerar o plano de aprendizado e aquelas para descrever o comportamento analítico do algoritmo VSS chega-se ao algoritmo que permite *projetar* o regime transitório e permanente do algoritmo VSS, sendo utilizado no restante do trabalho:

$$M = \text{Desajuste Desejado} ;$$

$$\alpha = \alpha_0 ;$$

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

$$\xi_{\infty}(\mu) = \frac{1}{1 - \frac{N}{2} \left(\frac{\mu \sigma_x^2}{1 - \mu \sigma_x^2} \right)} \cdot \xi_{\min} ; \quad (\text{Curva de Estabilização - (3.29b)})$$

$$\phi_{\text{opt}}(0) = \frac{\sigma_d^2 - \sigma_z^2}{N\sigma_x^2} ; \quad \phi_{\text{vss}}(0) = \frac{\sigma_d^2 - \sigma_z^2}{N\sigma_x^2} ; \quad (\text{Inicialização})$$

$$n = 0, n_{\max}$$

$$\xi_{\text{opt}}(n) = \xi_{\min} + N \cdot \sigma_x^2 \cdot \phi_{\text{opt}}(n)$$

$$\xi_{\text{vss}}(n) = \xi_{\min} + N \cdot \sigma_x^2 \cdot \phi_{\text{vss}}(n)$$

$$\mu_{\text{opt}}(n) = \frac{\phi_{\text{opt}}(n)}{(N+2)\sigma_x^2 \cdot \phi_{\text{opt}}(n) + \sigma_z^2}$$

$$\phi_{\text{opt}}(n+1) = [1 - 2\sigma_x^2\mu_{\text{opt}}(n) + (N+2)\sigma_x^4\mu_{\text{opt}}^2(n)]\phi_{\text{opt}}(n) + \mu_{\text{opt}}^2(n)\sigma_x^2\sigma_z^2$$

$$\phi_{\text{vss}}(n+1) = [1 - 2\sigma_x^2\mu_{\text{vss}}(n) + (N+2)\sigma_x^4\mu_{\text{vss}}^2(n)]\phi_{\text{vss}}(n) + \mu_{\text{vss}}^2(n)\sigma_x^2\sigma_z^2$$

$$\mu_{\text{vss}}(n+1) = \alpha \mu_{\text{vss}}(n) + \gamma \xi_{\text{vss}}(n)$$

próximo n

(4.28)

Apresenta-se a seguir um exemplo com especificações de projeto rigorosas, demonstrando a aplicação do método.

Um Exemplo com Especificações Rigorosas

A natureza de uma implementação prática geralmente determina o emprego de filtros adaptativos FIR com ordem elevada, muitas vezes com centenas de coeficientes. Em algumas situações, além da ordem elevada, podem ser necessárias especificações severas de regime permanente, implicando em desajustes pequenos. Widrow *et al.* utilizaram um desajuste de $M=2,5\%$ em [7]. Visando limites um pouco mais restritivos, inclusive para testar a eficácia da metodologia proposta quando submetida a especificações mais rigorosas, estipulou-se um desajuste desejado em regime de $M=1\%$.

O exemplo seguinte corresponde à identificação de sistemas, utilizando como sinal de referência $x(n)$ um ruído branco gaussiano com variância unitária. O filtro de Wiener utilizado para representar o sistema sob identificação é $W_0^T = [0,1; 0,1; \dots; 0,1]$, possuindo ordem $N=100$, a mesma do filtro adaptativo, resultando num filtro de Wiener com norma unitária. Prevenindo a modelagem exata, foi adicionado à saída da planta um ruído branco gaussiano $z(n)$ com variância $\sigma_z^2 = 10^{-3}$, modelando o ruído de medição sempre presente em situações práticas.

Aplicando a sugestão apresentada em [16], $\alpha_{vss}=0,97$, determina-se de (4.25) a constante $\gamma_{vss}=0,0057$ correspondente para se atingir o desajuste desejado ($M=1\%$). A trajetória do algoritmo, associada a este conjunto de parâmetros, é representada pela curva (a) no plano de aprendizado na Figura 4.5, o qual foi gerado por (4.28). Nota-se que esta trajetória percorre um caminho distante do caminho ótimo (curva (b)), devendo ser corrigida.

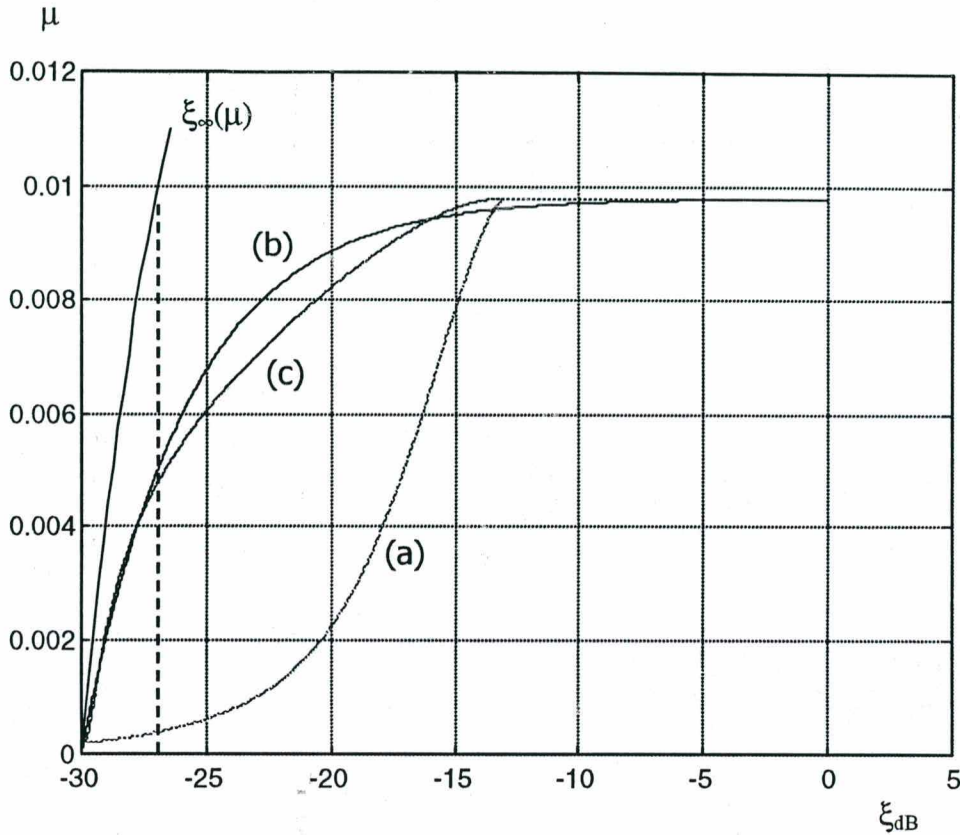


Fig. 4.5 – (a) Trajetória original. (b) Trajetória ótima. (c) Trajetória corrigida. Retra tracejada: início da região de estabilização.

A correção da trajetória resulta no conjunto de parâmetros $\alpha_{cor}=0.998$ e $\gamma_{cor}=3.9133 \cdot 10^{-4}$, com $\mu_{min}=0$, $\mu(0)=\mu_{max}=\mu_{fopt}=0,0098$ (equação 4.21). Claramente mais próxima da trajetória ótima (b), a trajetória corrigida, representada pela curva (c) na Figura 4.5, deve levar a um melhor desempenho do algoritmo.

A Figura 4.6 apresenta as evoluções do erro médio quadrático para o algoritmo projetado de acordo com [16] e para o projeto feito pela técnica proposta. As curvas contínuas correspondem às previsões teóricas baseadas em (3.33) e (3.34). A curva teórica correspondente à evolução ótima do erro (algoritmo (4.16)) confunde-se com a curva corrigida, não sendo mostrada portanto. Nota-se claramente que o novo projeto é superior, levando a uma convergência muito mais rápida para o mesmo nível de desajuste.

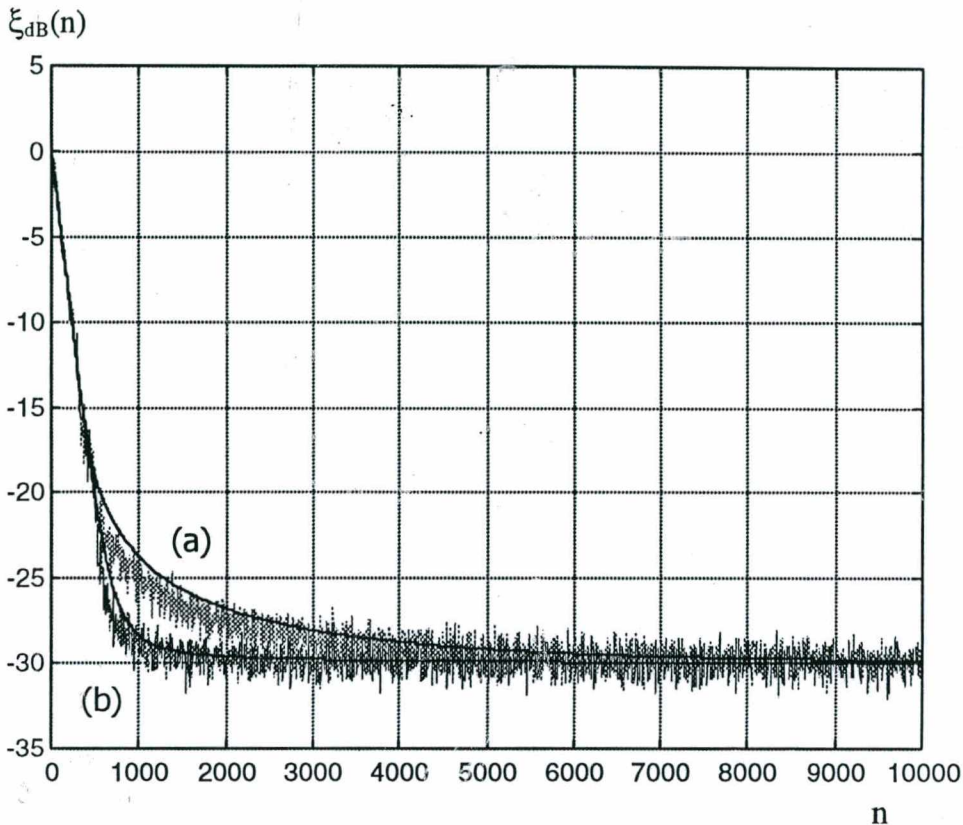


Fig. 4.6 – Curvas de aprendizado. (a) Curva original. (b) Curva otimizada. Linhas contínuas: curvas teóricas. Simulação Monte Carlo com 100 realizações

Embora não apresentadas aqui, as simulações realizadas com desajustes ainda menores (0,2% ou menos) utilizados em [17], por exemplo, revelaram discrepâncias consideravelmente maiores entre os desempenhos da nova metodologia derivada e daquela proposta em [16], sempre favoráveis à nova metodologia.

4.2.3 Comparando Algoritmos: MVSS versus VSS

À exceção de Koike [22], recentemente, o resultado ótimo demonstrado no Capítulo 3 geralmente não é empregado na literatura, seja para avaliação, ou para projeto dos algoritmos. Esta seção demonstra como pode-se chegar a conclusões errôneas por falta de critérios de avaliação da dinâmica dos algoritmos.

O algoritmo VSS inspirou o surgimento de outro algoritmo de adaptação do passo que, seguindo uma estrutura similar, foi denominado MVSS ("Modified

Variable Step-Size”) [17]. A equação de atualização do passo é dada por:

$$\mu(n+1) = \alpha\mu(n) + \gamma p^2(n) \quad , \quad \mu_{\min} \leq \mu(n) \leq \mu_{\max} \quad (4.29)$$

Comparando (4.29) com (4.23) nota-se a semelhança entre as duas estruturas, a menos da figura de mérito $p(n)$ usada para ajuste do passo. É exatamente essa a motivação do novo algoritmo. Segundo seus autores, a utilização do erro quadrático em (4.23) para controlar a adaptação do passo não reflete precisamente o progresso da adaptação, devido à presença de ruído de medição, inerente a qualquer sistema físico [17].

Alternativamente ao erro quadrático, Aboulnasr e Mayyas propuseram em [17] o emprego da correlação entre $e(n)$ e $e(n-1)$ como medida do estado da adaptação, e estimada através da média temporal:

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1) \quad (4.30)$$

Se o ruído de medição for branco, o valor esperado de (4.30) tende a zero com o progresso da adaptação, o que é útil para o ajuste do passo. Contudo, isso nem sempre é garantido [19]. Em aplicações de cancelamento eco, tal como em teleconferência, há a possibilidade da presença de ruídos de medição *coloridos*, nas quais o algoritmo MVSS também estaria sujeito a mesma crítica feita por seus autores ao algoritmo VSS: sensibilidade à presença de ruído.

As constantes α , γ e β devem ser escolhidas e determinam o comportamento do algoritmo. A introdução da constante adicional β , inexistente no algoritmo VSS, gera um grau de liberdade a mais. Por outro lado, o projeto do comportamento em regime permanente e no regime transitório, que no algoritmo VSS fica bem determinado com as técnicas propostas neste trabalho, continua seguindo uma abordagem “empírica”, baseada em resultados de simulação. Para ambientes estacionários, sugere-se em [17] que a constante β seja escolhida próxima a 1 ; enquanto $\alpha=0,97$, como sugerido originalmente em [16], “parece ser uma boa escolha para ambientes estacionários e não-estacionários”. Para a escolha prévia de α e β , a terceira constante γ pode ser otimizada para atingir um

nível mínimo de desajuste em regime, de acordo com a expressão fornecida em [17]. Entretanto, o procedimento sugerido não é suficientemente preciso para todas as aplicações. A escolha empírica de *duas* constantes, α e β , com a otimização posterior da terceira constante γ , dificulta o ajuste comparativo de trajetórias pelo plano de aprendizado, embora ainda este possa auxiliar no processo de escolha.

Em [17] foi apresentada uma análise do comportamento em regime permanente do algoritmo, não havendo portanto um modelo para a evolução do valor médio do passo de cálculo, o que dificulta a representação analítica no plano de aprendizado. Entretanto, para fins de comparação de desempenho com o algoritmo VSS, pode-se representar a evolução das trajetórias do algoritmo MVSS usando os resultados das simulações Monte Carlo, estimando a média estatística do erro quadrático e do passo de cálculo.

Nas simulações realizadas em [17] não foi mencionado o filtro de Wiener utilizado para modelar a planta sob identificação. Portanto, os exemplos que serão abordados nessa seção dizem respeito àqueles com sinal de referência $x(n)$ ruído branco, quando a evolução do erro médio quadrático não toma uma forma direcional [1] e pode-se reproduzi-los mesmo desconhecendo o filtro de Wiener empregado, a partir das informações fornecidas nas simulações. Nestes casos, a avaliação feita no plano de aprendizado demonstra que as simulações apresentadas em [17] não são conclusivas. O algoritmo VSS não está projetado adequadamente, levando à prematura conclusão acerca da superioridade do algoritmo MVSS.

No caso de sinais correlacionados, o conhecimento da solução de Wiener é imprescindível para caracterizar o problema da otimização do passo [14], bem como para representar o algoritmo no plano de aprendizado. Como essa informação não foi fornecida pelos autores em [17], não é possível fazer qualquer comparação de desempenho nestes casos.

Verifica-se de (4.5) que a evolução do erro médio quadrático depende somente do traço da matriz $K(n)$, ou do valor esperado da norma do vetor de erro

nos coeficientes. Qualquer filtro que possua norma igual a (4.14) garantirá ao algoritmo a mesma inicialização correspondente a (4.15), podendo portanto ser utilizado em detrimento do filtro de Wiener desconhecido - *o erro médio quadrático será equivalente ao gerado em [17]*.

De (4.5) e (4.9) chega-se à seguinte relação para o erro médio quadrático inicial, válida para $x(n)$ ruído branco e para a inicialização (4.3):

$$\xi(0) = \sigma_z^2 + \sigma_x^2 \cdot |W_0|^2 \quad (4.31)$$

As curvas apresentadas em [17] representam a evolução do erro médio quadrático em excesso. De (4.31) e de (2.35):

$$\xi_{\text{ex}}(0) = \xi(0) - \sigma_z^2 = \sigma_x^2 \cdot |W_0|^2 \quad (4.32a)$$

$$|W_0|^2 = \frac{\xi_{\text{ex}}(0)}{\sigma_x^2} \quad (4.32b)$$

Estimando o erro médio quadrático em excesso inicial das curvas apresentadas pode-se determinar a norma do filtro de Wiener utilizado nos exemplos. Seleciona-se então *um filtro de ordem N (fornecida em [17]) com coeficientes arbitrários*, corrigindo sua norma pela expressão (4.32b). Esse filtro comporta-se equivalentemente ao filtro de Wiener desconhecido, no que diz respeito ao erro médio quadrático. Isso possibilita a reprodução de alguns exemplos e suas respectivas avaliações pela metodologia aqui proposta.

Outra questão é a inicialização do passo de cálculo. Em [16] é dito que o valor inicial para o passo normalmente utilizado no algoritmo VSS é $\mu(0) = \mu_{\text{max}}$, embora afirme-se também que o algoritmo é insensível a esta escolha. Isto é correto somente para o comportamento do algoritmo em regime permanente, naturalmente. O comportamento dinâmico é afetado pelas condições iniciais.

Em [17] não é mencionada explicitamente a inicialização empregada para o passo, mas pelas considerações lá feitas e observando que o algoritmo MVSS é uma modificação do algoritmo VSS original, é razoável supor que o valor inicial utilizado seja também $\mu(0) = \mu_{\max}$, como sugerido em [16].

Algoritmo VSS Eficiente para Baixa SNR

Um argumento bastante explorado para justificar o algoritmo MVSS é a suposta sensibilidade de seu precursor, o algoritmo VSS, ao ruído de medição σ_z^2 . O MVSS é apresentado como uma evolução do VSS, mas isto não necessariamente é sustentado pelas simulações mostradas em [17].

Será reproduzido o *exemplo 1* de [17] - entrada ruído branco com baixa SNR. O sinal de referência é ruído branco gaussiano com variância unitária, excitando um sistema desconhecido FIR de ordem $N=4$ e um filtro adaptativo FIR de mesma ordem. O ruído de medição na saída da planta tem variância $\sigma_z^2 = 1$. O algoritmo MVSS foi utilizado com os parâmetros $\alpha_{\text{mvss}}=0,97$, $\gamma_{\text{mvss}}=10^{-3}$, $\beta_{\text{mvss}}=0,99$, escolhidos de maneira a produzir um erro médio quadrático em excesso de aproximadamente -34 dB em regime permanente. O algoritmo VSS foi implementado com $\alpha_{\text{vss}}=0,97$ e $\gamma_{\text{vss}}=10^{-5}$, o que corresponde, aplicando (4.24), a um desajuste de $M \approx 6,9 \cdot 10^4$. Esse desajuste implica num erro em excesso de aproximadamente -32 dB, contrastando com o valor previsto em [17], mas corroborado pelas simulações. Os algoritmos foram originalmente inicializados e truncados com $\mu(0) = \mu_{\max} = 0.1$ e $\mu_{\min} = 10^{-5}$ [16], [17].

De [17, Figura 1] estima-se $\xi_{\text{exdB}}(0) \approx 15$ dB, ou em termos absolutos $\xi_{\text{ex}}(0) \approx 31,6$. De (4.32b) determina-se $|W_o|^2 \approx 31,6$, dado que a variância de $x(n)$ é unitária. Um filtro que pode então ser utilizado para representar o sistema a identificar, resultando na mesma evolução do erro médio quadrático, é $W_o^T = [2,8 ; 2,8 ; 2,8 ; 2,8]$. A Figura 4.7 avalia a qualidade dos projetos em [17].

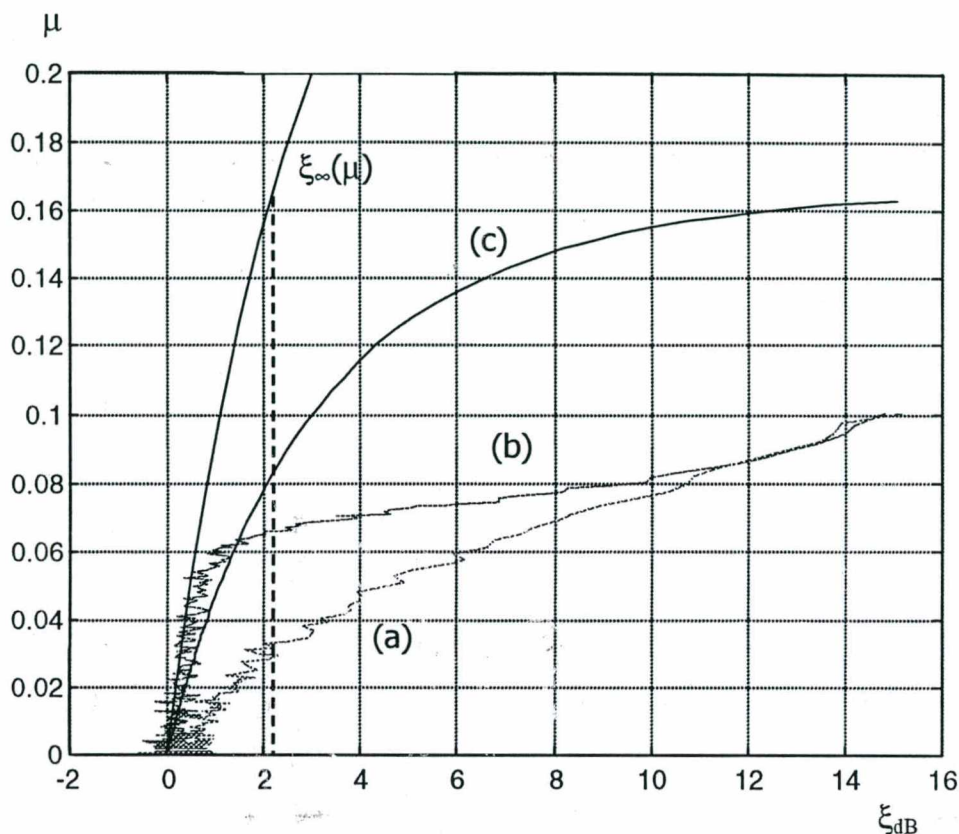


Fig. 4.7 – Exemplo 1 original [17]. (a) Algoritmo VSS. (b) Algoritmo MVSS. (c) Trajetória ótima. Retra Tracejada: início da região de estabilização. Simulação Monte Carlo com 200 realizações

Análise do Plano de Aprendizado

Analisando a Figura 4.7 verifica-se que:

- Para a mesma inicialização, a trajetória do algoritmo MVSS se aproxima da trajetória ótima na região de estabilização, ao contrário do algoritmo VSS, o que explica o seu desempenho superior no exemplo apresentado em [17];
- A dinâmica do algoritmo VSS pode ser corrigida mantendo-se o desajuste proposto. Isto inclui a sintonia dos parâmetros, inicialização e truncamento com valor do passo dado por (4.21b).

Utilizando o modelo (4.28), corrigiu-se a trajetória do algoritmo VSS para o desajuste de projeto $M \approx 6.9 \cdot 10^{-4}$, chegando-se aos valores $\alpha_{\text{cor}}=0,973$ e $\gamma_{\text{cor}}=9,05 \cdot 10^{-6}$. Adicionalmente, o algoritmo de controle do passo (4.23) foi inicializado e truncado com $\mu(0)=\mu_{\text{max}}=\mu_{\text{fopt}} \approx 0.163$, valor esse calculado de (4.21b). A Figura 4.8 compara a trajetória corrigida à original do artigo. As trajetórias teóricas mostram boa concordância com as trajetórias simuladas.

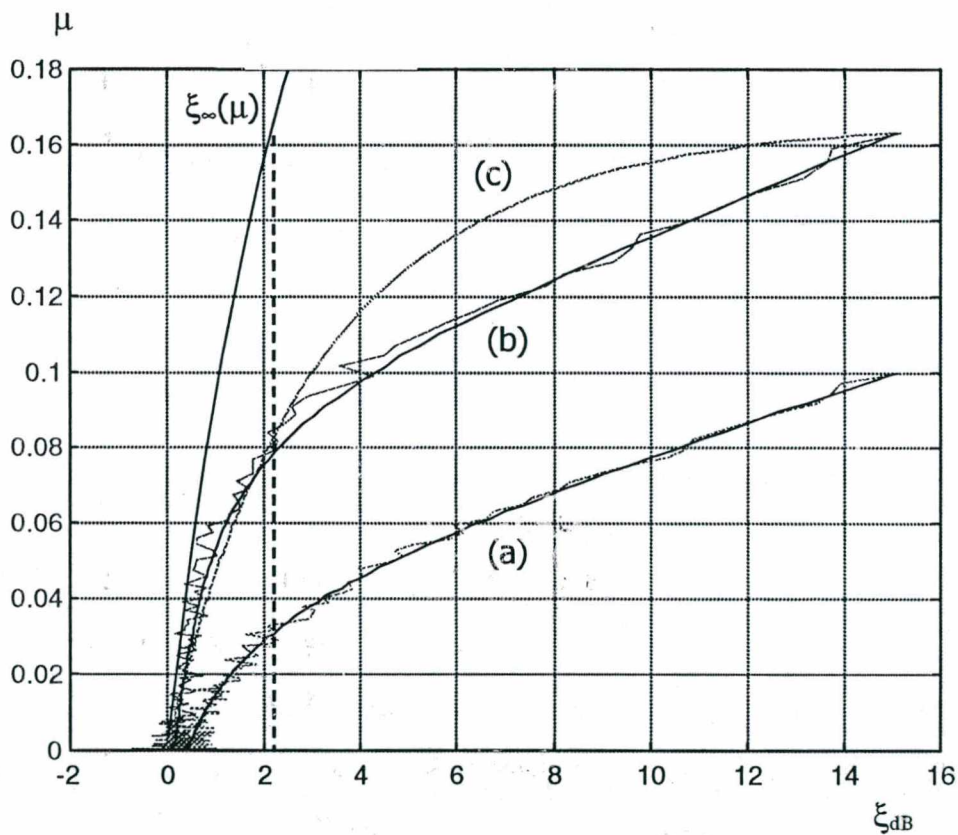


Fig. 4.8 - Correção da Trajetória do algoritmo VSS. (a) Algoritmo VSS original, curva teórica e simulada. (b) Algoritmo VSS corrigido e a previsão teórica correspondente. (c) trajetória ótima. Retra tracejada: início da região de estabilização. Simulação Monte Carlo com 200 realizações

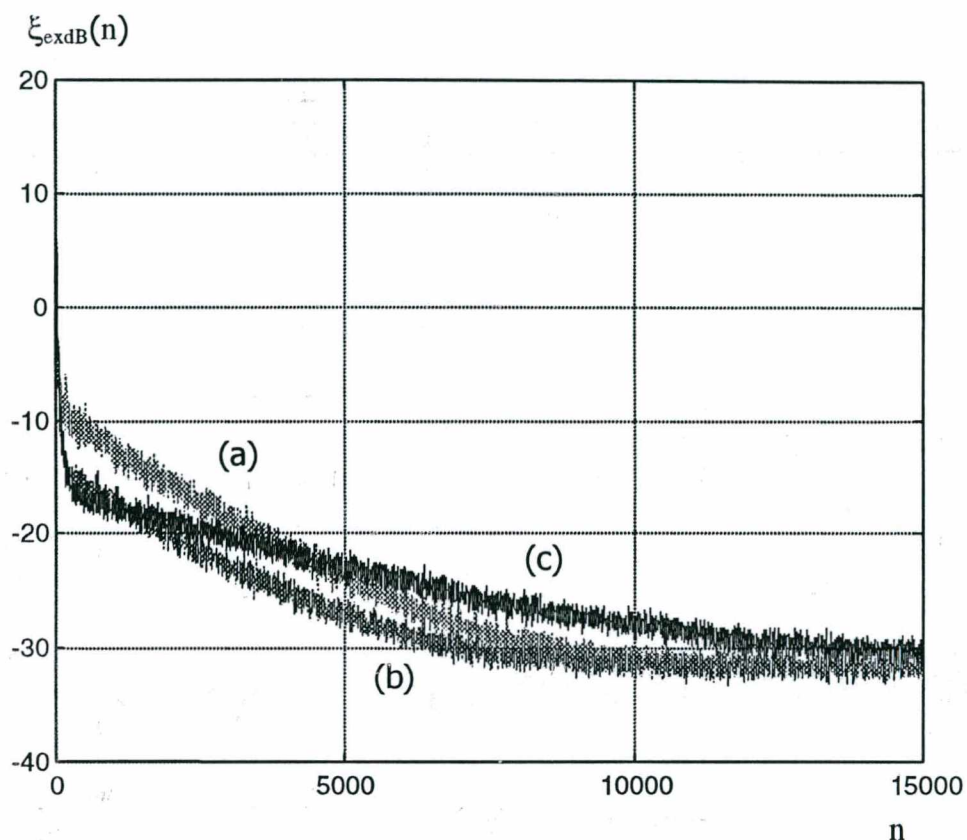


Fig. 4.9 – Algoritmo MVSS versus Algoritmo VSS. (a) Algoritmo VSS original. (b) Algoritmo VSS corrigido. (c) Algoritmo MVSS. Simulação Monte Carlo com 200 realizações.

As Figuras 4.9 e 4.10 demonstram a eficácia da correção da trajetória no algoritmo VSS, superando o algoritmo MVSS numa situação em que este deveria figurar como a melhor opção, segundo seus autores. Em particular, a Figura 4.10 mostra o algoritmo VSS com maior taxa de convergência inicial para o mesmo desajuste, divergindo da informação dada em [17].

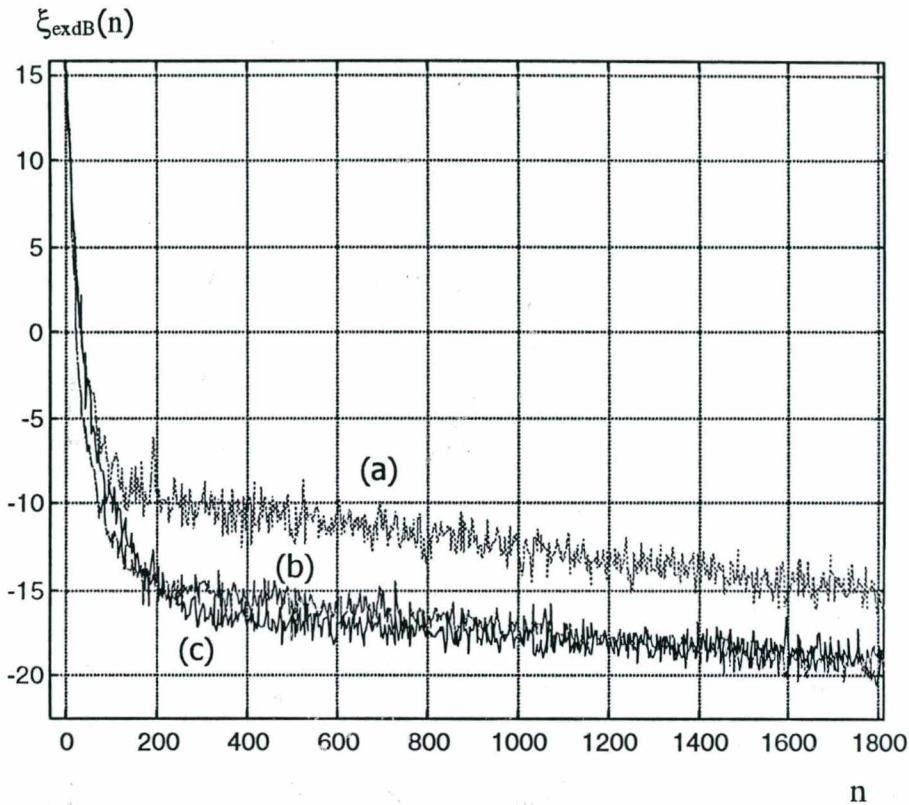


Fig. 4.10 - Detalhe da Figura 4.9 . (a) Algoritmo VSS original. (b) Algoritmo VSS corrigido. (c) Algoritmo MVSS. Simulação Monte Carlo com 200 realizações

Algoritmo MVSS para Grande SNR: Um Exemplo de Dinâmica Precária

O exemplo 3 em [17] também será reproduzido para avaliação no plano de aprendizado. Este exemplo é semelhante ao anterior, com parâmetros $\alpha_{mvss}=0,97$, $\beta_{mvss}=0,99$, $\gamma_{mvss}=1$, $\alpha_{vss}=0,97$, $\gamma_{vss}=0,02$, selecionados para produzir um erro médio quadrático em excesso de aproximadamente -60dB em regime permanente. O ruído de medição utilizado tem variância $\sigma_z^2 = 10^{-3}$. O intervalo de truncamento utilizado foi $\mu_{\max}=0,1$ e $\mu_{\min}=5 \cdot 10^{-4}$. Analisando a Figura 7 em [17], obtém-se $\xi_{\text{exdB}}(0) \approx 10 \text{ dB}$, o que equivale a $\xi_{\text{ex}}(0) \approx 10$ em valores absolutos. De (4.32b) chega-se a $|W_0|^2 \approx 10$, para $x(n)$ com variância unitária. Um filtro que comporta-se equivalentemente ao filtro de Wiener não fornecido é $W_0^T = [1,58 ; 1,58 ; 1,58 ; 1,58]$, e é utilizado para reproduzir o exemplo.

Como mostram as curvas na Figura 7 em [17], o algoritmo MVSS supostamente seria mais eficiente do que os demais algoritmos ali apresentados, o algoritmo VSS inclusive. Contudo, a Figura 4.11 mostra que os parâmetros do algoritmo VSS não foram sintonizados de maneira adequada, novamente levando a conclusões errôneas sobre a natureza do algoritmo.

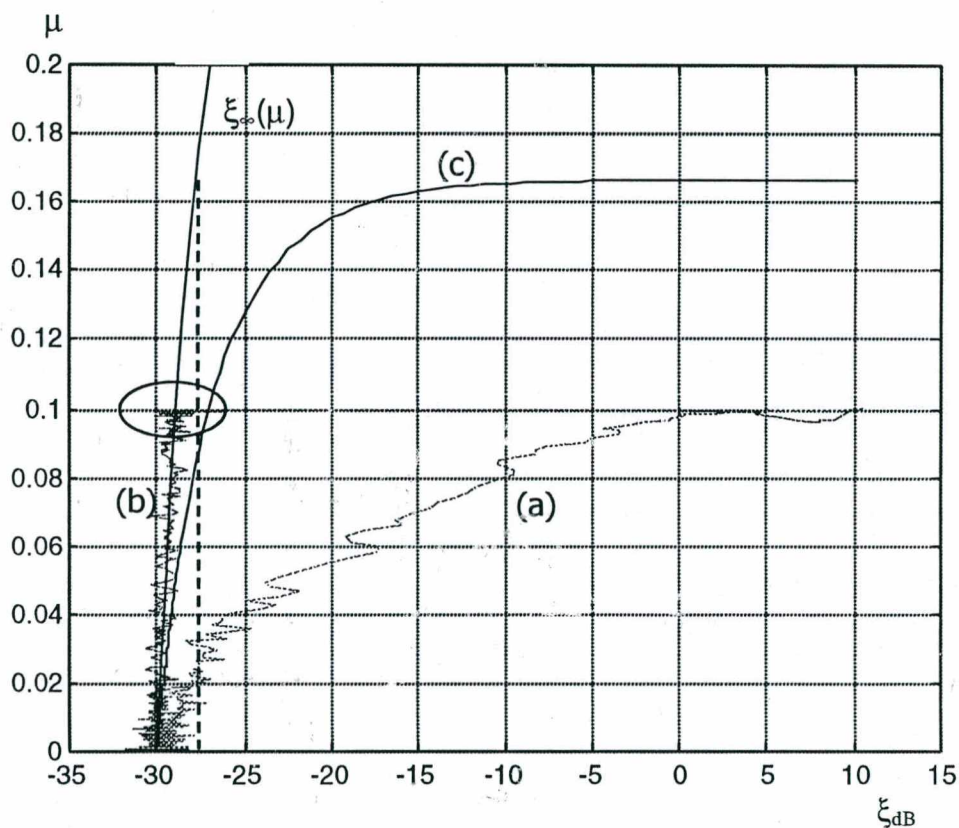


Fig. 4.11 – Avaliação das trajetórias originais. (a) Algoritmo VSS. (b) Algoritmo MVSS. (c) Trajetória ótima. Retra tracejada: início da região de estabilização. Simulação Monte Carlo com 200 realizações

Análise do Plano:

Analisando-se as curvas da Figura 4.11 tem-se:

- O algoritmo MVSS possui uma dinâmica precária para um algoritmo de passo variável, muito semelhante à dinâmica de passo fixo, chegando mesmo a sofrer uma breve polarização (estagnação) do passo, sinalizada pela elipse na Figura 4.11. Saindo da polarização, a sua trajetória acompanha a curva do regime permanente, o que certamente não é uma boa política, tornando o algoritmo lento;

- O algoritmo VSS percorre uma trajetória consideravelmente distante da trajetória ótima, inclusive na região de estabilização. Pode-se otimizar seu desempenho, corrigindo sua trajetória pela sintonia dos parâmetros.

O desajuste correspondente às constantes utilizadas para o algoritmo VSS é $M \approx 1,38 \cdot 10^{-3}$. Projetando o algoritmo pelo plano paramétrico chega-se ao conjunto de constantes que corrige a trajetória para o desajuste dado, $\alpha_{\text{cor}} = 0,977$ e $\gamma_{\text{cor}} = 0,0154$. A inicialização e o truncamento foram feitos com $\mu(0) = \mu_{\text{fopt}}$, $\mu_{\text{max}} = \mu_{\text{fopt}}$ e $\mu_{\text{fopt}} \approx 0,167$, respectivamente. O limite inferior do intervalo de truncamento utilizado foi $\mu_{\text{min}} = 5 \cdot 10^{-4}$, como citado no artigo.

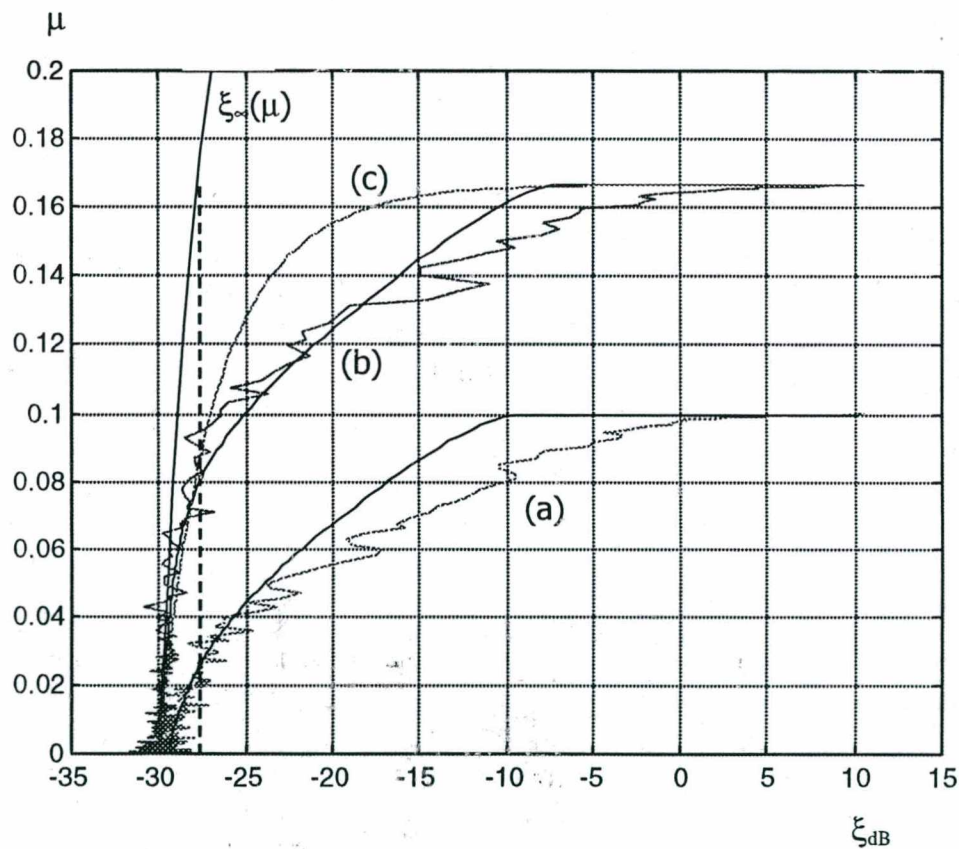
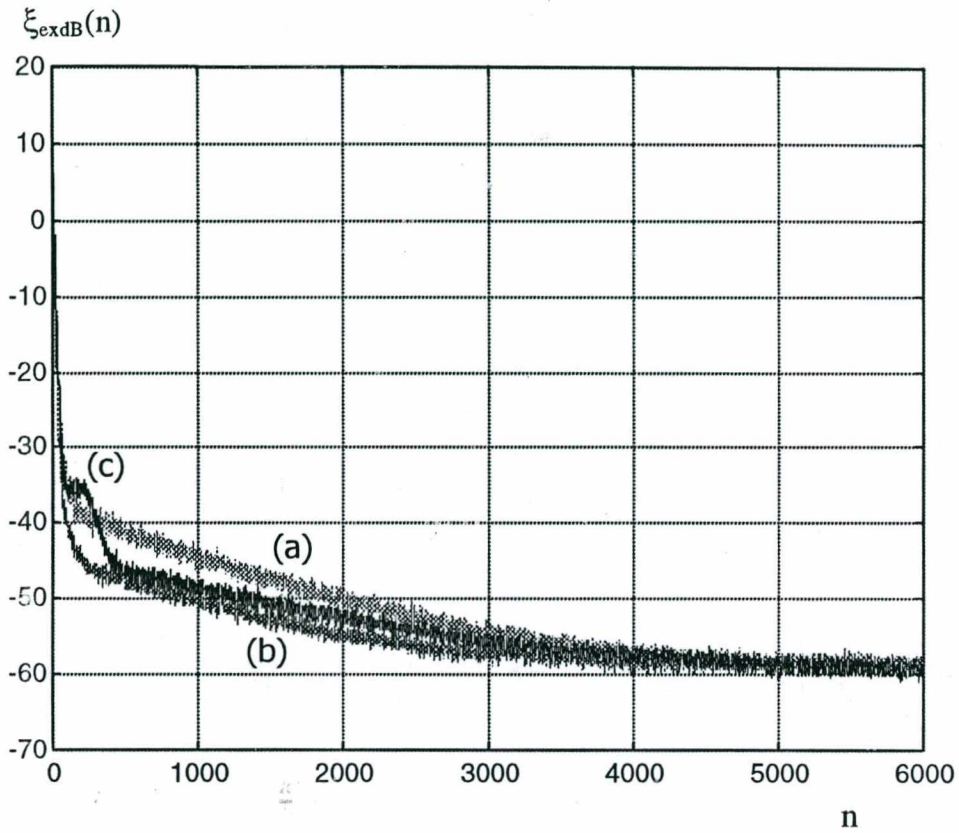


Fig.4.12 – Correção da trajetória do algoritmo VSS. (a) Trajetória original e a curva teórica (b) Trajetória corrigida e a curva teórica. (c) Trajetória ótima. Retra tracejada: início da região de estabilização. Simulação Monte Carlo com 200 realizações



*Fig. 4.13 - Erro Médio Quadrático em Excesso
(a) Algoritmo VSS original. (b) Algoritmo VSS otimizado. (c) Algoritmo MVSS
Simulação Monte Carlo com 200 realizações*

Na Figura 4.12 mostra-se a trajetória original e a trajetória corrigida pela técnica proposta. Além da boa concordância entre as curvas teóricas e as simuladas, a correção da trajetória do algoritmo VSS resulta em uma melhoria considerável em seu desempenho, superando o algoritmo MVSS em todos os instantes da adaptação, como mostra a Figura 4.13.

A carência de um modelo analítico para a evolução média do passo no algoritmo MVSS torna a correção de suas trajetórias uma tarefa extremamente laboriosa. Na tentativa de fazê-lo descrever uma trajetória mais adequada no exemplo da Figura 4.11, o algoritmo MVSS foi simulado com os parâmetros originais, mas com inicialização e truncagem ótimas (Figura 4.14).

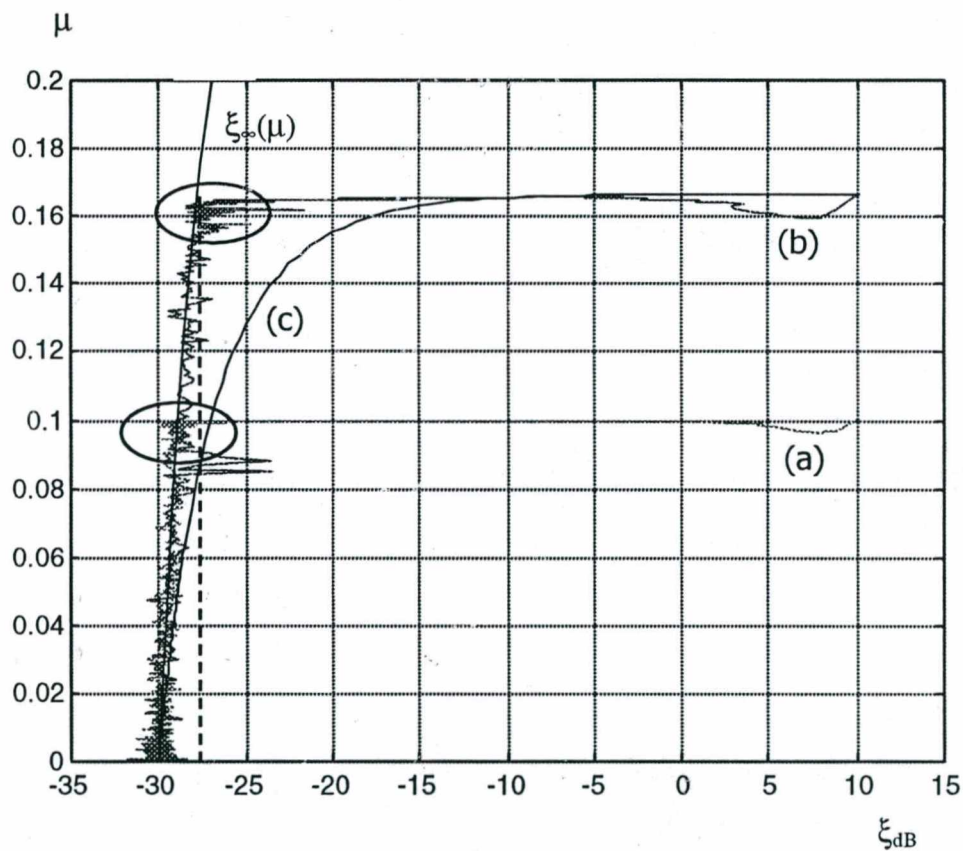
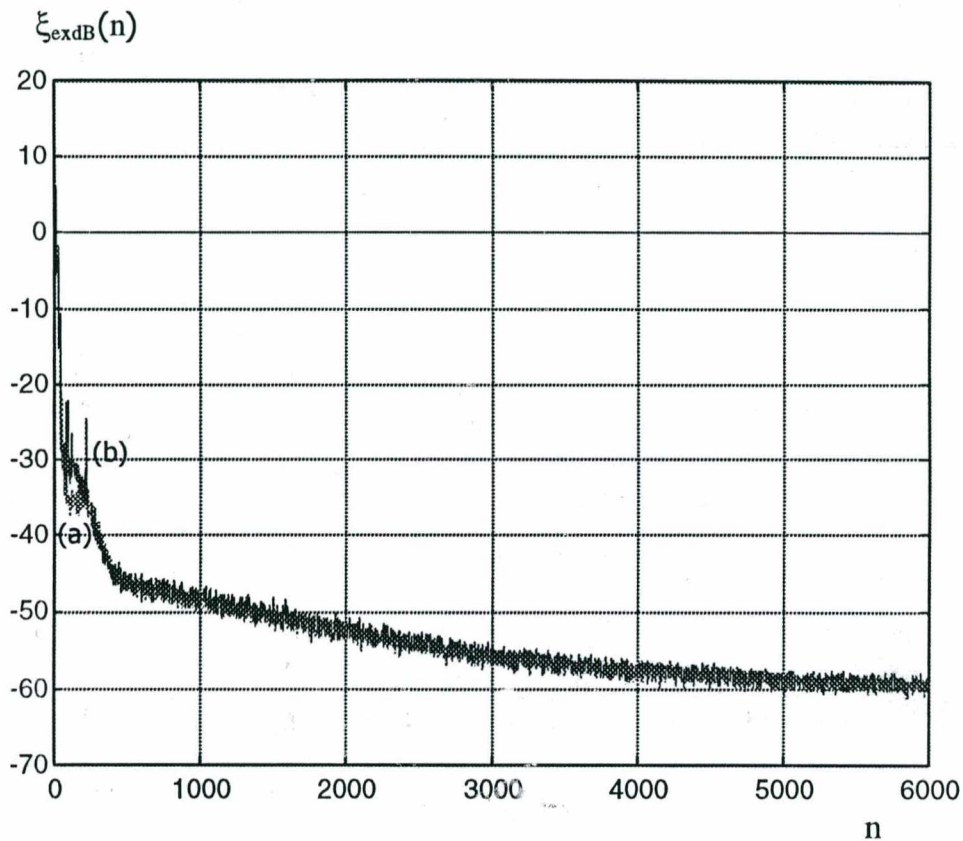


Fig. 4.14 - Impacto da truncagem e inicialização ótimas no algoritmo MVSS
 (a) Algoritmo MVSS original. (b) Algoritmo MVSS com inicialização e truncagem ótimas.
 (c) Trajetória ótima. Retra tracejada: início da região de estabilização
 Simulação Monte Carlo com 200 realizações



*Fig. 4.15 - Impacto da Truncagem Ótima no Algoritmo MVSS
(a) Algoritmo MVSS original. (b) Algoritmo MVSS com inicialização e truncagem ótimas.
Simulação Monte Carlo com 200 realizações*

Como mostram as curvas (b), nas Figuras 4.15 e 4.16, a dinâmica correspondente não sofre mudanças significativas, apresentando novamente um comportamento semelhante à evolução em passo fixo, como o projeto original em [17].

Para o conjunto de constantes sugerido em [17], o algoritmo MVSS apresenta claramente uma tendência à polarização do passo, ressaltada pelas elipses nas Figuras 4.11, 4.14 e 4.16.

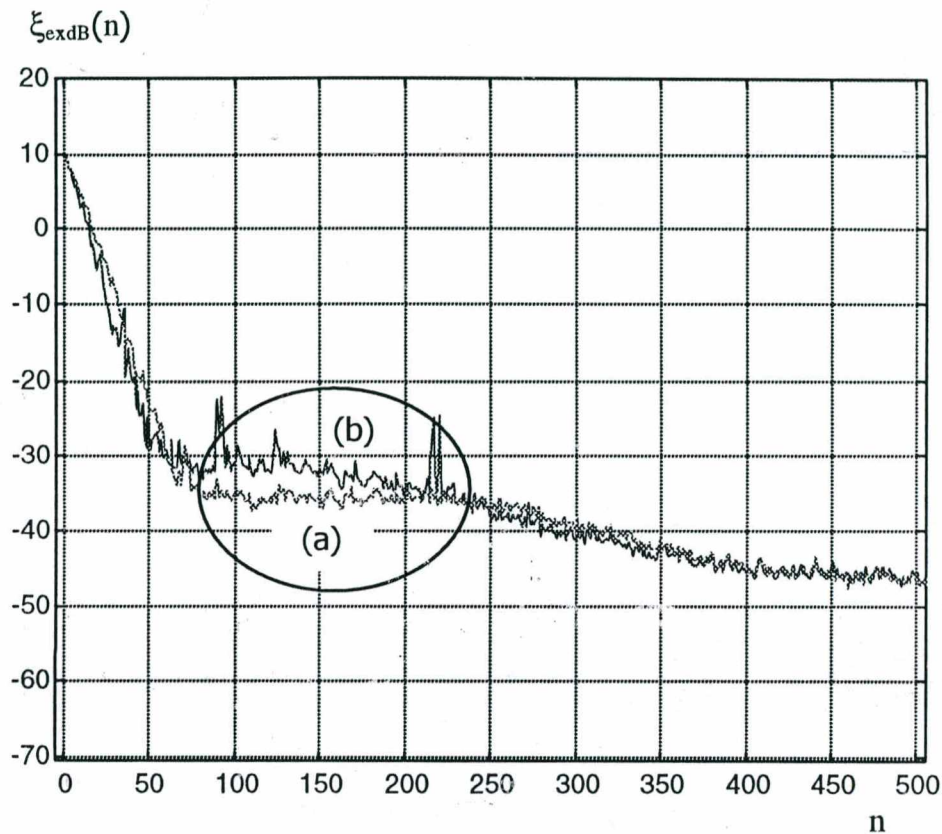


Fig. 4.16 – Detalhe da Figura 4.15
 (a) Algoritmo MVSS original. (b) Algoritmo MVSS com inicialização e truncagem ótimas.
 Simulação Monte Carlo com 500 realizações

4.3. Projeto da Dinâmica dos Algoritmos ou Busca Ótima?

Alguns algoritmos existentes [1], [20], [21] buscam de maneira aproximada o passo ótimo a cada iteração, usando estimativas do gradiente do erro em relação ao passo de cálculo para ajustá-lo automaticamente, numa abordagem semelhante àquela utilizada pelo algoritmo LMS para adaptação dos coeficientes do filtro. O propósito desta seção é avaliar a busca automática pretendida por alguns algoritmos de controle frente à correção da dinâmica para o desajuste de projeto, proposta neste trabalho.

Mathews e Xie [21] exploraram o princípio do algoritmo gradiente-descendente para ajustar o passo, reduzindo o erro médio quadrático

de estimação com o avanço da adaptação. O algoritmo possui um grau de liberdade, uma constante a escolher que controla o comportamento adaptativo do passo. Isso pode ser limitante quando as especificações de projeto são rigorosas. Em [17], mostra-se que o algoritmo descrito pode ser inaceitavelmente lento quando as especificações de projeto tornam-se mais rigorosas (desajustes pequenos). Portanto, este algoritmo não será abordado aqui.

Haykin resgata em [1] um algoritmo de sintonia automática do passo analisado em [20], que também segue a abordagem gradiente-descendente para ajustar a constante de aprendizado. Contudo, o algoritmo descrito é computacionalmente complexo e parece apresentar problemas de polarização do passo pelas simulações realizadas (não mostradas aqui), não sendo uma boa alternativa para especificações rigorosas de regime permanente.

Em publicação recente [22], Koike apresenta um algoritmo que segue a política de ajuste automático do passo e que parece comportar-se bem para desajustes pequenos. A cada iteração o passo é aproximado ao valor ótimo teórico via acumuladores com perdas, realizando um controle quase-ótimo [22]. O algoritmo foi derivado levando em consideração não-linearidades, tornando-o aplicável em outros algoritmos procedentes do algoritmo LMS [22]. Aqui, entretanto, será abordado o algoritmo LMS convencional. As expressões de adaptação do passo fornecidas em [22], particularizadas para o LMS convencional, são apresentadas a seguir.

$$\mu(n) = \frac{q_0^T(n) q(n)}{\tau(n)} \quad (4.33)$$

com:

$$q_0(n+1) = (1-\rho)q_0(n) + \rho e(n)X(n) \quad (4.34)$$

$$q(n+1) = (1-\rho)q(n) + \rho e(n)X(n) \quad (4.35)$$

$$\tau(n+1) = (1-\rho_\tau)\tau(n) + \rho_\tau [e(n)X^T(n)X(n-L_d)]^2 \quad (4.36)$$

Os fatores de perdas ρ e ρ_τ influenciam o comportamento do algoritmo, tanto em regime permanente quanto durante o regime transitório. As expressões (4.34) e (4.35) realizam a mesma operação, por isso apenas uma expressão precisa ser implementada no caso convencional do algoritmo LMS. A constante $L_d \geq 1$ depende do nível de correlação das amostras do vetor de entrada [22]. Como no caso abordado aqui, $x(n)$ é ruído branco, e o valor utilizado é $L_d=1$, embora a sua variação não tenha resultado em alterações perceptíveis nas simulações realizadas. O algoritmo será referenciado como NASS, do inglês "Novel Adaptive Step-Size Control Algorithm".

Desempenho em Pequenos Desajustes: Um Exemplo

Serão comparados o algoritmo VSS [16], projetado pela metodologia desenvolvida, e o algoritmo proposto em [22]. Ambos os algoritmos foram adequados para atingir um desajuste de $M \approx 1\%$, verificando-se a dinâmica correspondente de ambos no plano de aprendizado. O filtro de Wiener utilizado foi $W_o^T = \frac{1}{\sqrt{30}} [1; 1; 1; \dots; 1]$, o que determina $|W_o|=1$ para $N=30$. A variância do ruído de medição $z(n)$, branco e gaussiano, adicionado à saída da planta é $\sigma_z^2 = 10^{-3}$. O sinal de referência utilizado é ruído branco gaussiano com variância unitária.

O conjunto de parâmetros para o algoritmo VSS projetado pela metodologia proposta é $\alpha_{cor}=0,994$ e $\gamma_{cor}=0,0039$. Seguindo a sugestão de [22], o algoritmo NASS foi implementado com fatores de perdas $\rho=\rho_\tau=0,002$, visando o desajuste de projeto. Foi utilizada uma constante L_d unitária, em se tratando de ruído branco na entrada do filtro. As simulações Monte Carlo foram implementadas com 100 realizações. Ambos os algoritmos foram inicializados e truncados com os valores de passo ótimo, com $\mu(0)=\mu_{max}=\mu_{fopt} \approx 0,0312$ (calculado de (4.21b)). O limite inferior utilizado para o intervalo de truncamento foi $\mu_{min}=0$.

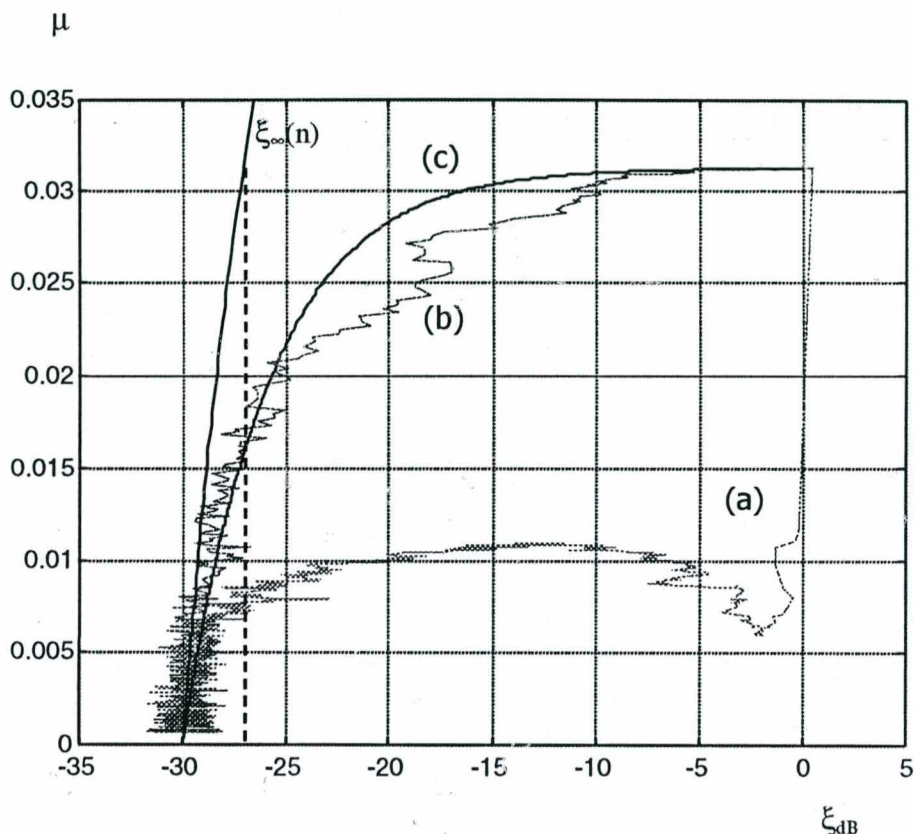


Fig.4.17 – Comparação: VSS versus NASS. (a) Algoritmo NASS. (b) Algoritmo VSS otimizado. (c) Trajetória ótima. Reto tracejada: início da região de estabilização. Simulação Monte Carlo com 100 realizações

A Figura 4.17 mostra as trajetórias percorridas pelos algoritmos. Nitidamente a trajetória do algoritmo VSS otimizado (b) é superior a trajetória do algoritmo NASS (a), o qual percorre regiões de dinâmica lenta, entrando na região de estabilização distante da trajetória ótima. Isto determina uma dinâmica ineficiente, exigindo um número maior de iterações para atingir o desajuste de projeto.

Na Figura 4.18 confirma-se a avaliação do plano. Enquanto o algoritmo NASS é lento no início da adaptação, supera posteriormente o algoritmo VSS original, simulado com $\alpha=0,97$ e $\gamma=0,0019$ para atingir o desajuste desejado, convergindo mais rápido. O algoritmo VSS otimizado, entretanto, supera a ambos, apresentando uma dinâmica superior para o mesmo desajuste em regime permanente.

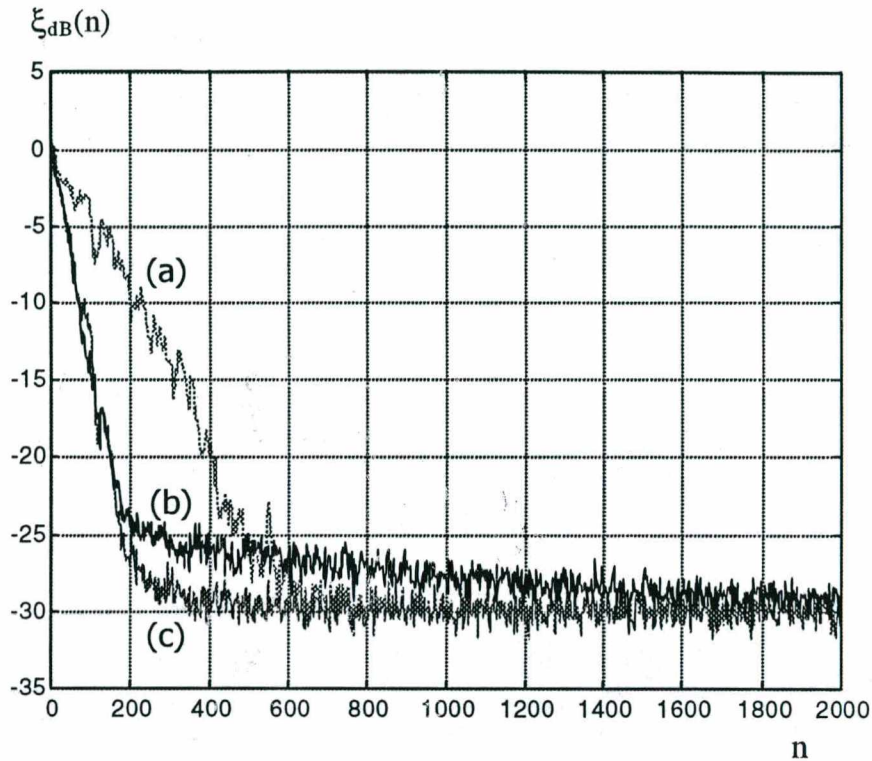


Fig.4.18 – Erro médio quadrático. Algoritmo NASS versus Algoritmo VSS
(a) Algoritmo NASS. (b) Algoritmo VSS original. (c) Algoritmo VSS otimizado.
Simulação Monte Carlo com 100 realizações

Os algoritmos de passo variável que implementam a filtragem de alguma variável do contexto adaptativo para ajustar o passo [16], [17], [18], [28], podem ser vantajosos frente àqueles que implementam a busca automática [1], [20], [21], [22]. Além de computacionalmente mais simples e independentes da ordem do filtro, possuem melhor desempenho quando bem projetados. Para um certo desajuste, apresentam uma dinâmica mais eficiente. Particularmente, a aplicação da metodologia de projeto desenvolvida neste capítulo no algoritmo VSS o revela como um bom candidato para implementação em diversas situações em que os algoritmos adaptativos são utilizados; notadamente, identificação de sistemas.

4.4. Um Algoritmo Determinístico para Controle do Passo

Originalmente nos algoritmos do gradiente estocástico era comum o emprego do passo de cálculo variável [14], proporcionando as vantagens já descritas no Capítulo 2. No mesmo trabalho, o autor descreve o algoritmo LMS como *um algoritmo do gradiente estocástico no qual o passo não decresce* ("a nonvanishing step-size version of a stochastic gradient algorithm") e propõe uma seqüência de passos inversamente proporcional ao número de iterações, também explorada por Macchi em [12], resgatando a abordagem original dos algoritmos do gradiente estocástico [1]:

$$\mu(n) = \frac{A}{n+B}, \quad \mu_{\min} \leq \mu(n) \leq \mu_{\max} \quad (4.37)$$

A política de controle do passo empregada atende as hipóteses adicionais utilizadas na derivação do modelo analítico do erro médio quadrático na Seção 3.6.3 do Capítulo 3, seguindo uma evolução determinística e independente do ambiente do filtro adaptativo.

Nesta seção demonstra-se um exemplo de projeto do algoritmo descrito em (4.37) pelo plano de aprendizado. As curvas simuladas apresentam uma boa concordância com a previsão teórica. A planta sob identificação possui ordem $N=10$ com coeficientes $W_o^T = \frac{1}{\sqrt{10}} \cdot [1; 1; \dots; 1]$, o que determina um filtro de Wiener com norma unitária. O filtro adaptativo utilizado para modelar a planta desconhecida foi implementado com mesma ordem. O sinal de referência utilizado é ruído branco gaussiano com variância unitária. Previnindo a modelagem exata, adicionou-se à saída da planta um ruído de medição $z(n)$ branco e gaussiano com variância $\sigma_z^2 = 10^{-2}$. O intervalo de truncamento utilizado foi $\mu_{\max} = \mu_{\text{fopt}} = 0,0826$ e $\mu_{\min} = 0$. A inicialização nesse caso não se faz necessária, dada a natureza da estrutura de adaptação do passo.

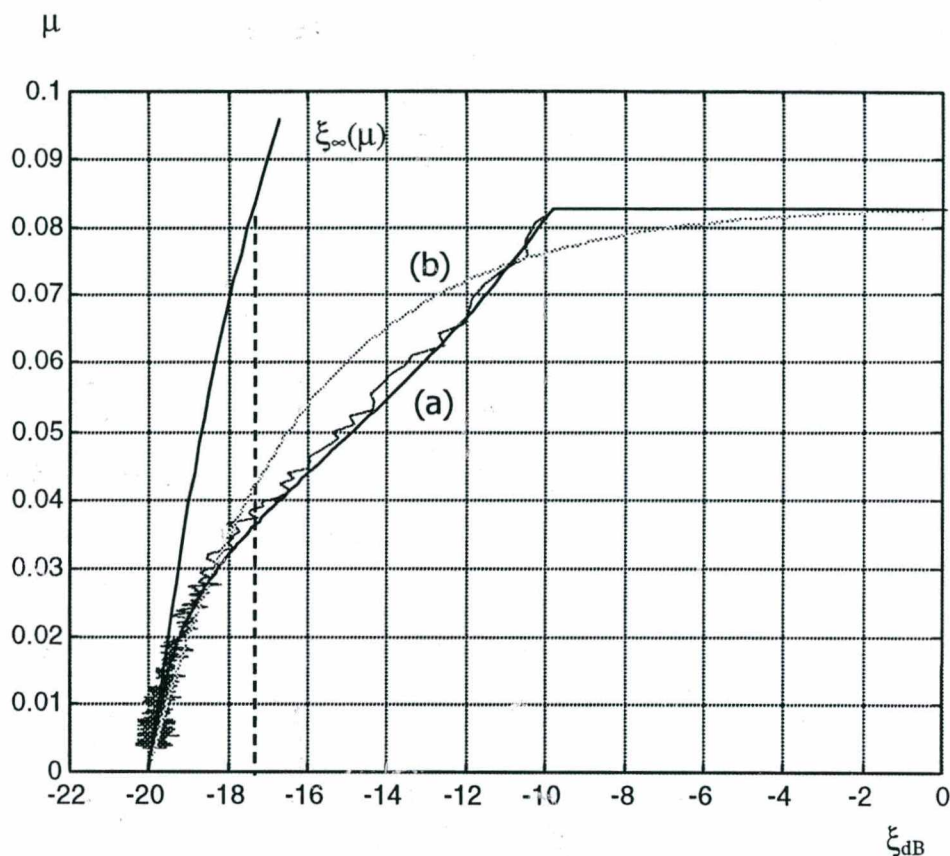


Fig. 4.19 – Estrutura Determinística. (a) Trajetória do algoritmo e evolução teórica. (b) Trajetória ótima. Retra tracejada: início da região de estabilização
Simulação Monte Carlo com 1000 realizações

Ajustando a trajetória do algoritmo pelo plano de aprendizado, chega-se a um conjunto de constantes que leva a uma boa evolução no plano, $A=2,5$ e $B=1$. A Figura 4.19 apresenta o plano de aprendizado para a situação.

A Figura 4.20 mostra a evolução do erro médio quadrático: curvas teórica e simulada. A curva teórica do erro médio quadrático ótimo é praticamente coincidente com a projetada.

Como verifica-se de (4.37), a evolução do passo é sempre decrescente, o que evita a polarização do mesmo, mas pode dificultar o rastreamento da solução ótima quando em ambientes não-estacionários. A operação de divisão no mecanismo de adaptação do passo pode ser problemática, dependendo das limitações práticas existentes nos processadores. Entretanto, este é mais um

exemplo ilustrativo que corrobora a estrutura de projeto apresentada, mostrando boa concordância entre a previsão teórica e as curvas simuladas.

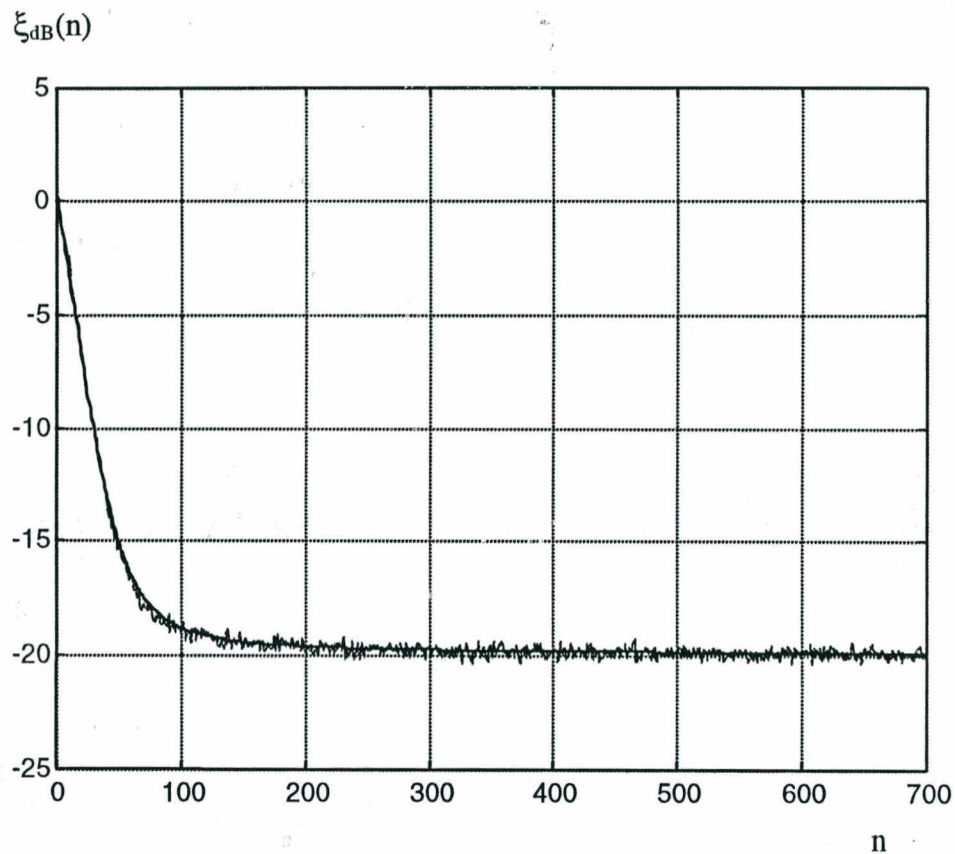


Fig. 4.20 – Curva de aprendizado simulada e a previsão teórica.
Simulação Monte Carlo com 1000 realizações

4.5.Caso Não-Estacionário

Sistemas físicos em geral são não-estacionários. Para algumas aplicações, entretanto, essa característica se manifesta de forma quase imperceptível dentro do intervalo de observação utilizado, justificando a premissa de estacionaridade adotada em inúmeros trabalhos da literatura. Em outras circunstâncias, a planta sob identificação pode sofrer mudanças abruptas em sua estrutura, o que se reflete na alteração da solução de Wiener. Esse tipo de comportamento é classificado como *não-estacionaridade abrupta*.

A metodologia desenvolvida no decorrer deste trabalho também pressupõe a invariância dos parâmetros do sistema físico sob identificação. No entanto, é interessante avaliar o impacto de não-estacionaridades no comportamento do algoritmo que será empregado na identificação do sistema físico desconhecido, avaliando sua robustez e a qualidade da solução apresentada quando numa situação adversa.

Esta seção avalia a sensibilidade das técnicas de projeto desenvolvidas à presença de não-estacionaridades *abruptas*, quando, em algum período da adaptação, a planta do sistema sofre uma variação brusca em seus parâmetros, provocando um "salto" no erro médio quadrático [16], [17], [19], [21].

O algoritmo VSS será comparado ao algoritmo de controle quase-ótimo mais recente na literatura, o algoritmo NASS [22], mostrando novamente superioridade e afirmando-se como uma boa opção mesmo em ambientes não-estacionários.

4.5.1. Algoritmo NASS Versus Algoritmo VSS

O exemplo da Seção 4.3 será repetido com os mesmos parâmetros. Os algoritmos são liberados à convergência, quando, na iteração $n=1000$, manifesta-se uma mudança abrupta na solução de Wiener, $W'_0 = -W_0$. [17], [21], provocando o "salto" descrito no erro médio quadrático.

A Figura 4.21 apresenta os resultados da simulação Monte Carlo. Verifica-se que mesmo projetado para um desajuste pequeno, $M=1\%$, o algoritmo VSS otimizado ainda mantém sua superioridade no caso não-estacionário abrupto, afirmando a vantagem da metodologia desenvolvida sobre o procedimento proposto em [16].

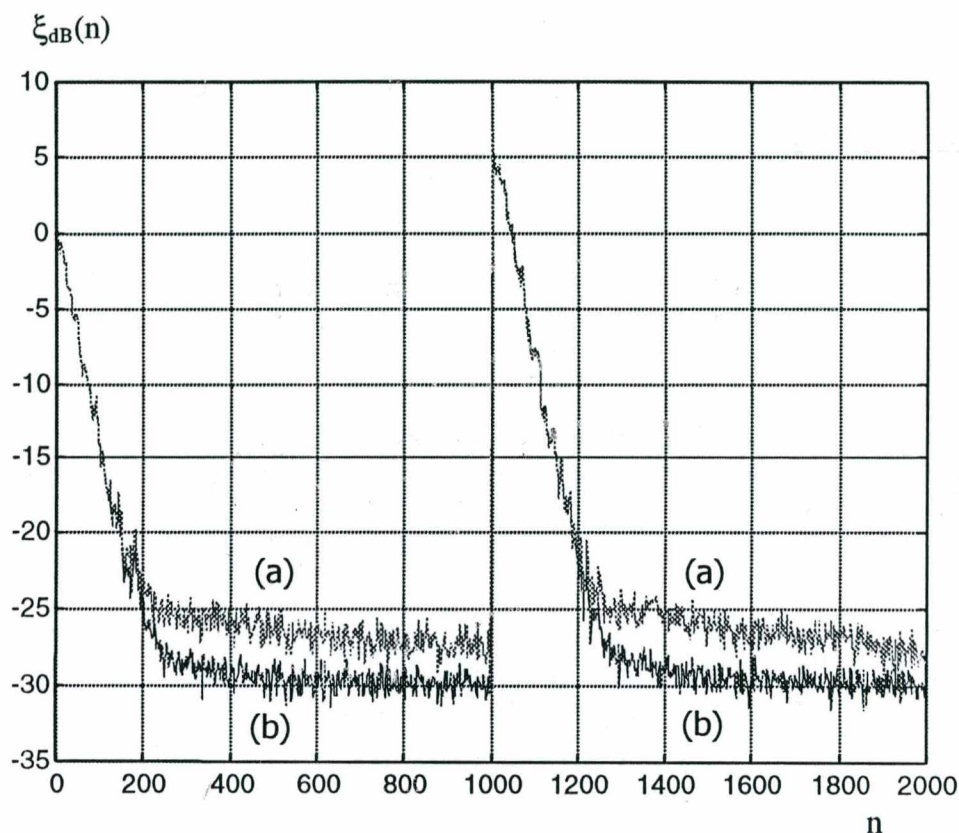


Fig. 4.21 – Erro médio quadrático. Mudança abrupta da solução de Wiener em $n=1000$
(a) Algoritmo VSS original. (b) Algoritmo VSS otimizado.
Simulação Monte Carlo com 100 realizações

A Figura 4.22 compara o desempenho dos algoritmos VSS e NASS. Quando submetido à não-estacionaridade, o algoritmo VSS praticamente mantém seu desempenho, convergindo rapidamente para o desajuste de projeto. O algoritmo NASS demonstra maior sensibilidade, aumentando o tempo de convergência relativo à condição inicial, não justificando a complexidade computacional empregada no ajuste do passo. O algoritmo VSS novamente afirma-se como uma boa alternativa.

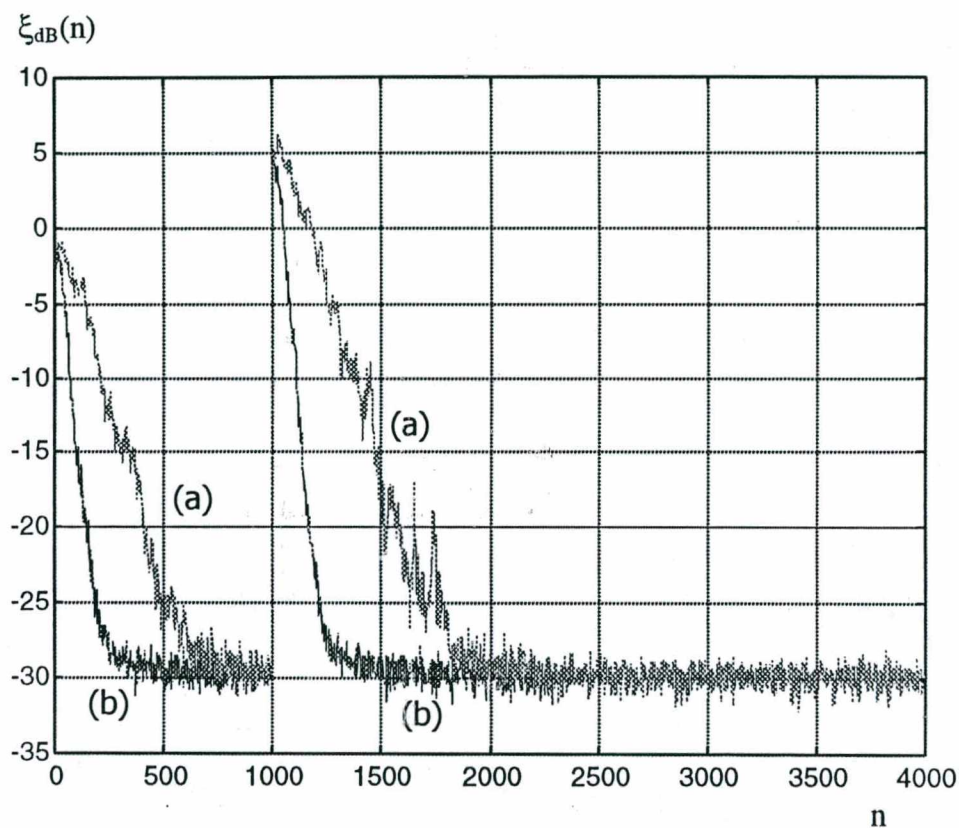


Fig. 4.22 – Erro médio quadrático. Mudança abrupta da solução de Wiener em $n=1000$
(a) Algoritmo NASS. (b) Algoritmo VSS otimizado.
Simulação Monte Carlo com 100 realizações

Por tratar-se de uma abordagem distinta, ferindo as premissas utilizadas no desenvolvimento da metodologia, as não-estacionaridades contínuas, onde a solução de Wiener altera-se continuamente no tempo, não serão abordadas aqui, exigindo atenção futura.

4.6. Conclusão

Inviável no caso geral correlacionado, o projeto da dinâmica de algoritmos de passo variável mostra-se possível nos casos em que o sinal de referência comporta-se como ruído branco, levando a diversas aplicações, conforme explanou-se no decorrer deste capítulo.

Da evolução ótima desenvolvida no Capítulo 3, derivou-se um novo resultado para otimização da velocidade de convergência utilizando passo de adaptação fixo, válido quando o sinal de referência é ruído branco. Levando em conta mais informações sobre a natureza do ambiente adaptativo, a nova técnica leva a um melhor desempenho do que os resultados existentes [6], [11], [12], [31], convergindo mais rápido e com desajustes menores em regime permanente. Adicionalmente, o passo fixo ótimo pode ser utilizado para inicializar os algoritmos de controle do passo, também estabelecendo um limite superior ótimo para a evolução da seqüência de passos, mantendo o algoritmo estável.

Pela estrutura simples e, principalmente, por satisfazer razoavelmente bem as premissas utilizadas na derivação do modelo analítico no Capítulo 3, o algoritmo descrito em [16] foi escolhido para demonstrar o ambiente de projeto desenvolvido. Possuindo dois graus de liberdade, permite o projeto da dinâmica e do regime permanente. Utiliza uma figura de mérito ostensivamente criticada em [17], o erro quadrático, mas ao contrário do ali mencionado e também em [22], pode apresentar bons desempenhos quando comparado a outros algoritmos, se adequadamente projetado.

Abordou-se também o algoritmo MVSS, apresentado em [17] como uma evolução do algoritmo VSS [16], mostrando como pode-se chegar a conclusões erradas a respeito da natureza dos algoritmos quando não se emprega critérios eficientes de avaliação. Algumas simulações ali apresentadas foram reproduzidas e comparadas com o algoritmo VSS otimizado pelo plano de aprendizado. Nos casos explorados o algoritmo VSS revelou-se superior, mesmo em ambientes ruidosos, supostamente os nichos de aplicação do algoritmo MVSS,

como defendido em [17].

Na Seção 4.3 mostrou-se que, especialmente para pequenos desajustes em regime permanente, o algoritmo VSS é superior em desempenho e possui menor complexidade computacional, quando comparado aos geralmente complexos algoritmos de busca quase-ótima do passo [1], [20], [21], [22]. Normalmente dependentes da ordem do filtro utilizado, esses algoritmos podem se tornar proibitivos em certas aplicações, com desempenhos que muitas vezes não justificam a complexidade computacional empregada.

A premissa básica para desenvolvimento do trabalho foi a estacionaridade do ambiente adaptativo. Embora aceitável em muitas situações práticas, é sempre interessante avaliar o impacto de possíveis variações nas características do sistema sob identificação. Isso foi feito na Seção 4.5, na qual avalia-se a sensibilidade do algoritmo VSS a variações bruscas na solução ótima, confrontando-o com o algoritmo proposto em [22]. O algoritmo VSS também revelou desempenho superior, quando projetado pelas técnicas apresentadas anteriormente.

Outro tipo de não-estacionaridade que manifesta-se regularmente em situações práticas é a contínua, na qual as propriedades estatísticas de alguns (ou todos) sinais modificam-se continuamente no decorrer do tempo. Esse caso não foi explorado, carecendo de estudos futuros.

Capítulo 5

Conclusões, Críticas, Conjecturas, e Sugestões para Trabalhos Futuros

"Quem alcança um ideal, vai além dele"

Friedrich Nietzsche

5.1. Conclusões

Seguramente a mais atrativa característica do algoritmo LMS é a sua simplicidade. A mais detrativa é a sua convergência relativamente lenta, quando comparado a algoritmos como o RLS ou o filtro de Kalman. O algoritmo LMS apresenta uma vocação natural para operação de filtros adaptativos, especialmente em aplicações em que se exige simplicidade e a velocidade de convergência não é um fator determinante, apresentando uma robustez satisfatória mesmo frente a erros de implementação [9].

É corrente na literatura que a operação do algoritmo LMS empregando passo variável é sempre uma abordagem preferível. Sob qualquer ponto de vista de otimização, a adaptação é melhor assistida pela utilização de passo variável [14]. Em ambientes estacionários, a escolha de uma seqüência de passos apropriada acelera significativamente a convergência do algoritmo para um dado desajuste em regime permanente. Em ambientes não-estacionários, os algoritmos de passo variável são geralmente mais eficientes no rastreamento da solução ótima [15].

Das vantagens proporcionadas pela adaptação do passo surgiram inúmeros algoritmos de controle da constante de aprendizado, com estruturas, filosofias e parâmetros diferentes, o que pode tornar a opção por uma estrutura em especial um trabalho "hercúleo".

A idéia central deste trabalho foi desenvolver uma metodologia de avaliação e projeto dos algoritmos de controle, aproveitando alguns resultados

existentes e derivando outros, em detrimento da proposição de novos algoritmos de passo variável. Com uma metodologia apropriada para avaliação do desempenho em regime permanente e das características de convergência, observando também a complexidade computacional das estruturas, pode-se decidir sobre o algoritmo de controle do passo a ser utilizado.

O ponto de partida foi a análise da dinâmica do algoritmo LMS. A necessidade de indicativos absolutos de desempenho era clara. Foi então derivado o modelo analítico da evolução ótima do algoritmo LMS com passo variável, envolvendo a evolução ótima do erro médio quadrático e a seqüência de passos ótima correspondente [9] , [22]. A evolução ótima estabelece os objetivos dos algoritmos de passo variável existentes. Entretanto, era também necessária uma forma de avaliação dos algoritmos que aproveitasse adequadamente o resultado ótimo derivado.

A avaliação dos algoritmos de passo variável normalmente feita na literatura envolve a observação da seqüência de passos produzida pelo algoritmo de controle e a evolução correspondente do erro médio quadrático [16], [17], [19], [21]. Contudo, avaliar a influência da seqüência de passos na curva de aprendizado não é tarefa trivial quando realizada em curvas separadas.

Em busca de uma maneira mais eficiente de aproveitar a evolução ótima derivada na avaliação dos algoritmos, mostrou-se que a adaptação em passo variável pode ser caracterizada de forma mais genérica por trajetórias no espaço tridimensional, reunindo numa única curva informações simultâneas do passo, do erro médio quadrático e do número de iterações. Assim, comparando as trajetórias descritas pelos algoritmos com a trajetória ótima, obtida da evolução ótima derivada, poder-se-ia avaliar o desempenho dos algoritmos, em um único lugar geométrico. Resolveu-se um problema, criou-se outro: comparar curvas contidas no espaço não é algo corriqueiro, quando muito, factível.

As trajetórias espaciais, em princípio de pouco valor teórico, possuem projeções nos diedros erro médio quadrático em função do número de iterações $[\xi, n]$, e passo de cálculo em função do número de iterações $[\mu, n]$, gerando as

curvas clássicas utilizadas na literatura para avaliação dos algoritmos de passo variável - a seqüência de passos e a curva de aprendizado. Estas projeções poderiam ser utilizadas para avaliação dos algoritmos, mas em diedros separados, exigindo a observação das duas curvas clássicas para uma avaliação mais confiável da evolução da adaptação.

Uma terceira projeção, até então inexplorada, toma lugar no terceiro diedro que compõe o espaço $[\xi, \mu, \eta]$ no qual estão contidas as trajetórias genéricas: o erro médio quadrático em função do passo de cálculo, denominado *plano de aprendizado*.

Conforme procedem as iterações, os algoritmos sob análise descrevem trajetórias características no plano de aprendizado, combinando informações simultâneas da figura de mérito do algoritmo LMS e da constante de aprendizado. Isto gerou uma representação mais completa e compacta, na qual o número de iterações é um parâmetro implícito.

Superou-se assim o problema da visualização das trajetórias genéricas no espaço e a avaliação clássica em curvas separadas: compara-se as trajetórias características com a trajetória ótima na nova representação proposta. Os algoritmos têm seus desempenhos avaliados de acordo com a região percorrida no plano de aprendizado. O plano de aprendizado também comporta uma representação do regime permanente dos algoritmos de passo variável, caracterizada pela curva de estabilização. Esta representa o conjunto dos pontos em que as trajetórias tornam-se estacionárias, auxiliando na avaliação da dinâmica e do desempenho após a convergência.

A etapa seguinte foi o desenvolvimento de modelos analíticos para construção do plano de aprendizado e para a geração das trajetórias características. Embora os algoritmos possam ter suas trajetórias características geradas por simulações Monte Carlo, uma base analítica é sempre desejável.

Reunindo o modelo analítico da evolução ótima do algoritmo LMS, o modelo analítico para geração de trajetórias, e a caracterização do processo adaptativo no plano de aprendizado, construiu-se uma metodologia simples e

eficiente para *avaliação e projeto de algoritmos de passo variável* baseados no algoritmo LMS em ambientes estacionários.

Em se tratando de sinais correlacionados, com uma distribuição arbitrária de autovalores, a *direcionalidade* do algoritmo LMS [1] restringe a metodologia proposta à *avaliação* dos algoritmos de passo variável, uma vez que a geração das trajetórias depende de informações não conhecidas a priori, como a solução de Wiener.

Em aplicações nas quais o sinal de referência é ruído branco, como nos importantes casos de identificação de sistemas, os modelos analíticos explanados transformam-se em equações recursivas escalares, permitindo a construção do plano de aprendizado baseando-se em informações estatísticas conhecidas ou estimadas com relativa facilidade dos dados. Nessas aplicações pode-se então avaliar e projetar os algoritmos de passo variável existentes.

Para mostrar a sua potencialidade, no Capítulo 4 a metodologia desenvolvida foi aplicada na avaliação e projeto de alguns algoritmos propostos na literatura [12], [14], [16], [17], [22]. Um dos resultados obtidos foi um novo valor ótimo para o passo fixo que maximiza a taxa inicial de convergência do erro médio quadrático. Além de tornar o LMS mais rápido, relativo aos resultados clássicos [6], [11], [12], [31], o novo passo ótimo derivado determina *desajustes menores* em regime permanente.

Dedicou-se particular atenção ao algoritmo proposto em 1992 por Kwong e Johnston [16] e criticado em [17], [19] e [22]. Pela simplicidade inerente e constituindo uma estrutura que filtra o erro quadrático de predição para controlar o passo de cálculo, o algoritmo VSS atende razoavelmente bem às premissas estatísticas utilizadas no desenvolvimento dos modelos analíticos, sendo um ótimo candidato para aplicar a metodologia proposta.

No primeiro exemplo desenvolvido, modificou-se convenientemente o procedimento original proposto em [16], gerando um novo procedimento no qual pode-se projetar o regime permanente e o regime transitório, assistidos pela metodologia proposta e pela análise realizada pelos autores. Quando projetado

pela nova metodologia, o algoritmo VSS apresenta excelente desempenho, mesmo para especificações de projeto rigorosas, ou quando sujeito a certos tipos de não-estacionaridade, isso com uma complexidade computacional adicional irrisória frente ao caso clássico com passo fixo.

A importância da utilização de resultados analíticos na avaliação de desempenho dos algoritmos de passo variável torna-se clara nos exemplos reproduzidos e avaliados na Seção 4.2.3. Afirmações apoiadas unicamente em resultados de simulação são pouco confiáveis, a ausência de qualquer respaldo analítico pode levar a conclusões equivocadas acerca da natureza dos algoritmos e de seus desempenhos, conforme ocorre em [17].

A superioridade do algoritmo MVSS, descrita em [17] e reafirmada em [19] e [22], não é suportada pelas simulações apresentadas em [17], pelo menos por aquelas que foram possíveis reproduzir com as informações disponíveis. O algoritmo VSS apresenta um desempenho igual ou superior ao apresentado, inclusive em situações que deveriam respaldar o algoritmo MVSS, como nos casos de baixas SNR ("Signal-to-Noise Ratio").

Algumas dificuldades enfrentadas neste trabalho dão margem à críticas que assolam muitos trabalhos da literatura: a omissão de informações essenciais para reproduzir os exemplos explorados [16], [17], [25], [26]. Uma prática comum é apresentar conclusões baseadas em exemplos sem as informações necessárias para reproduzi-los. Para aplicações nas quais o sinal $x(n)$ é ruído branco, pode-se estimar alguns parâmetros a partir das curvas e reproduzir os exemplos, como foi feito na Seção 4.2.3. Contudo, para sinais correlacionados informações como a inicialização do filtro adaptativo e a solução de Wiener empregada nas simulações são cruciais para caracterizar completamente os exemplos explorados. A escassez destas informações impossibilitou a avaliação de alguns exemplos apresentados em [17].

Para testar a qualidade dos modelos gerados quando as premissas estatísticas são atendidas, projetou-se um estrutura analisada em [12], e utilizada de maneira semelhante em [14]. A política proposta resgata a abordagem original

dos algoritmos do gradiente estocástico, na qual o passo evoluía de maneira determinística, inversamente proporcional a alguma potência do número de iterações. Os resultados das simulações revelam uma concordância muito boa com as previsões teóricas dos modelos analíticos.

Foram comparadas também duas filosofias, de certa forma, concorrentes: o projeto da dinâmica dos algoritmos e as estruturas de otimização automática do passo. A comparação foi feita entre o algoritmo VSS [16] e o algoritmo de busca automática mais recente da literatura, proposto por Koike em [22], chegando-se a resultados interessantes. Em aplicações cujo sinal de referência é ruído branco e necessita-se de especificações mais rigorosas de projeto, a correção da dinâmica das estruturas de filtragem, em especial aquela proposta em [16], apresenta vantagens. Além de estruturalmente mais simples, leva a resultados expressivamente melhores, mesmo em ambientes onde se manifestam certos tipos de não-estacionaridades (Seção 4.5).

5.2. Críticas

Talvez mais importante que desenvolver uma teoria seja criticá-la, revelando sua abrangência e desvendando suas limitações e restrições lógicas.

A análise matemática dos algoritmos invariavelmente envolve compromissos entre a tratabilidade matemática e o realismo do modelo a ser produzido. Os modelos analíticos apresentados baseiam-se na popular hipótese da independência estatística. É provavelmente o melhor que pode ser feito do ponto de vista pragmático [9].

A evolução ótima do algoritmo LMS foi obtida a partir do modelo analítico a passo fixo, o qual também utiliza a hipótese da independência. Naturalmente a seqüência ótima será autêntica quando as premissas utilizadas forem atendidas. Ocorre que em diversas aplicações o filtro adaptativo processa séries temporais de dados, para as quais as premissas estatísticas reunidas na seção 3.2 do Capítulo 3 não se verificam. O que em última instância significa, dentro de um rigorismo lógico maior, que a seqüência gerada no Capítulo 3 não é,

estritamente, ótima. Entretanto, os resultados obtidos no Capítulo 4 demonstram que mesmo conceitualmente inválida, ou aproximadamente válida, a seqüência ótima derivada encontra-se além da suficiência para aplicações em geral, levando a resultados bastante expressivos.

Constituindo um problema de análise com complexidade notadamente maior, justificar o modelo utilizado para descrever o comportamento do erro médio quadrático com passo variável exige premissas estatísticas adicionais àquelas que naturalmente incorporam a teoria da independência (Seção 3.6.3).

Vincular o passo de cálculo a alguma variável do processo adaptativo também o torna, a rigor, uma variável aleatória. Pela estrutura do algoritmo LMS, pressupor que o passo $\mu(n)$ é independente de outras variáveis do processo leva em geral a simplificações consideráveis de análise. De acordo com as estruturas de controle do passo utilizadas, são feitas simplificações de análises ligeiramente diferentes. Sob este ponto de vista, as premissas adicionais adotadas neste trabalho (Apêndice), (H.3) e (H.4), são as mínimas possíveis, relativas às utilizadas na literatura [8], [15], [16], [17], [21].

As premissas (H.3) e (H.4) fornecem bons resultados para algumas estruturas [16], [17], [18], [22], nas quais a adaptação do passo é obtida através de filtragem passa-baixas, o que atenua consideravelmente o comportamento aleatório do passo de cálculo, tornando-o aproximadamente independente das outras variáveis do processo adaptativo.

Em outras estruturas, entretanto, constata-se uma evolução do passo bastante condicionada às variáveis das estruturas, como em [1], [8], [20], [25], [26]. Nesses casos, a concordância entre o modelo teórico para a evolução do erro médio quadrático proposto neste trabalho e os resultados de simulação podem ser discrepantes.

Em situações práticas, é comum a manifestação de não-estacionaridades, quando as propriedades estatísticas dos sinais modificam-se no decorrer do tempo. Isso fere as premissas utilizadas no desenvolvimento da metodologia, sendo importante verificar até onde a correção da dinâmica pelo

plano de aprendizado ainda proporciona vantagens. Avaliou-se na Seção 4.5 o impacto das *não-estacionaridades abruptas*, verificando que o projeto pela metodologia proposta não sofre alterações consideráveis. No entanto, é bastante desejável avaliar a qualidade do projeto quando as propriedades estatísticas dos sinais variam continuamente no tempo, o que é comum em muitas aplicações. Isto não foi feito, carecendo de atenção adicional!

5.3. Algumas Conjecturas (ou "Quase Conclusões")

Muitos resultados foram extraídos durante o desenvolvimento do trabalho, mas por circunstâncias diversas, não foram aqui formalizados. Não constituindo uma base formal, mas representando informações importantes no que concerne os algoritmos de passo variável e a escolha das estruturas, traça-se algumas conjecturas a seguir.

As Estruturas Auto-Otimizáveis

Quando o filtro adaptativo trata sinais correlacionados com distribuições genéricas de autovalores, torna-se difícil otimizar *a priori* os parâmetros dos algoritmos de passo variável de maneira satisfatória, dada a direcionalidade do processo de convergência do algoritmo LMS. Uma conjectura possível nesses casos é aconselhar o uso das estruturas de busca automática, menos sensíveis à escolha de seus parâmetros e à natureza dos dados, quando comparadas às outras estruturas. O problema é empregar uma estrutura que implemente de maneira satisfatória a busca automática do passo ótimo, ainda que de maneira aproximada, em situações de projeto mais exigentes, como desajustes pequenos em regime permanente.

A algoritmo analisado em [20] e revisitado em [1] parece apresentar problemas com a polarização do passo. Pelas simulações realizadas, mas não mostradas neste trabalho, o algoritmo apresenta dificuldades para alcançar valores pequenos para o passo, o que certamente limita o desajuste em regime

permanente alcançável pela estrutura.

Mathews *et al.* [21] exploraram uma estrutura que também segue a abordagem gradiente-descendente, assim como a estrutura apresentada em [20]. Abolunasr e Mayyas [17] mostraram que este algoritmo pode-se tornar inaceitavelmente lento quando lhe é exigido operar sob desajustes rigorosos em regime permanente. Isto ocorre também com o algoritmo proposto por Koike [22], como mostrou-se na Seção 4.3.

Para aplicações nas quais o desajuste requerido em regime permanente não é rigoroso, as estruturas de busca automática citadas conseguem otimizar a dinâmica do algoritmo LMS satisfatoriamente, podendo ser uma boa opção. Destaca-se aquela analisada por Mathews *et al.*, representando a estrutura computacionalmente mais simples das analisadas e mostrando bons desempenhos em [21].

Uso de Lógica Difusa na Adaptação do Passo

Com o desenvolvimento do plano de aprendizado e da trajetória ótima, o uso de lógica difusa para adaptar o passo [25], [26] fica descaracterizado. Lá, o sistema de inferência difuso utiliza como variável lingüística de entrada o erro quadrático (também o número de iterações, numa extensão do trabalho), mapeando-o na variável de saída, o passo de cálculo, através de uma base de regras lógicas simples.

Ora, o plano de aprendizado é exatamente o mesmo lugar geométrico onde o sistema difuso opera. A trajetória ótima é precisamente a relação entrada/saída que deveria ser implementada pelo sistema difuso para operação ótima no sentido médio quadrático do erro. Isto enfraqueceria o argumento defendido em [25] e [26] para justificar a estrutura difusa no controle do passo. Existe um modelo analítico formal e razoavelmente preciso, onde o sistema difuso pode ser visto como uma aproximação do modelo formal [32]. Seria eficiente na medida em que implementasse a relação ditada pela trajetória ótima, a qual

poderia ser, obviamente, usada para sintonizar e particionar os conjuntos difusos.

Embora não formalizadas neste trabalho, as simulações realizadas mostram que devido à natureza aleatória do passo gerado, o sistema difuso não consegue atender às exigências do regime permanente e do regime transitório simultaneamente. Quando se sintoniza os conjuntos difusos pela trajetória ótima, a estrutura apresenta problemas sérios de polarização do passo, levando a desajustes inaceitáveis em regime permanente.

Quando se relaxa a sintonia dos conjuntos difusos, empregando passos menores no particionamento e sintonia dos conjuntos, consegue-se desajustes menores, mas a custo da velocidade de convergência. Novamente, a complexidade computacional empregada não se traduz em desempenho.

A potencialidade do sistema de inferência difuso não está convenientemente aproveitada na forma proposta em [25] e [26]; consegue-se resultados muito melhores com estruturas muito mais simples, como a estrutura proposta em [16], quando projetada pela metodologia aqui apresentada.

5.4.Sugestões para Trabalhos Futuros

Conforme diz o aforismo do início do capítulo, alcançar um ideal é ultrapassá-lo. Certamente o trabalho desenvolvido não esgotou a linha de pesquisa explorada. A seguir sugere-se alguns temas em potencial para a continuidade do trabalho.

Incluindo Não-Linearidades nos Modelos Analíticos

Em 1999, Koike [22] proporcionou uma contribuição importante, desenvolvendo um modelo analítico da evolução ótima do algoritmo LMS que inclui não-linearidades na estrutura original do algoritmo, o que permite abordar outros algoritmos derivados diretamente dele. Isto possibilitaria estender a metodologia proposta para avaliar e projetar algoritmos tais como o SRA ("Signed Regressor Algorithm"), o SSA ("Sign-Sign Algorithm"), ou o LMF ("Least Mean

Fourth Algorithm") [23], dentre outros [22], [24].

Extensão para o Algoritmo LMS Normalizado

Dirk Slock derivou em [14] a seqüência ótima de passos para o algoritmo LMS normalizado. Com este resultado, e outras análises adicionais lá feitas, pode-se estender a metodologia proposta para avaliar algoritmos de controle do passo baseados no algoritmo LMS normalizado [14], [17], [18], [19].

Dois Novos Algoritmos

Embora o propósito deste trabalho não seja elaborar novos algoritmos, duas das estruturas mencionadas podem ser convenientemente modificadas, tornando-as exploráveis pela metodologia proposta.

Suprimindo uma das equações do algoritmo proposto por Aboulnasr e Mayyas [17], pode-se utilizar uma única equação que simultaneamente adapta o passo e realiza a estimativa da figura de mérito utilizada, assim como foi feito com o erro quadrático de estimação em [16]:

$$\mu(n+1) = \alpha\mu(n) + \gamma e(n)e(n-1)$$

Assim, gera-se um algoritmo em que seria possível projetar tanto o regime permanente como a dinâmica, semelhante ao já feito na Seção 4.2 com o algoritmo VSS. Inspirando-se nas análises feitas em [16] e [17], pode-se derivar uma expressão do desajuste em regime permanente e corrigir a dinâmica do algoritmo pela metodologia proposta, dado um modelo analítico para a evolução da auto-correlação do erro $E[e(n)e(n-1)]$.

Novamente inspirando-se no algoritmo VSS, pode-se também modificar o algoritmo proposto em [18], fornecendo à estrutura original mais um grau de liberdade. O novo algoritmo possuiria a vantagem não garantida por Aboulnasr e Mayyas em [17], no caso de ruídos de medição coloridos, como mostrado em [19]. O passo de cálculo gerado pela estrutura não seria polarizado no sentido

médio estatístico, dado o princípio da ortogonalidade:

$$\mu(n+1) = \alpha\mu(n) + \gamma e(n) x(n)$$

Com um modelo para a evolução média da quantidade $e(n)x(n)$ e uma expressão para o desajuste, ter-se-ia outro algoritmo que permite o projeto da dinâmica e do regime permanente pela metodologia desenvolvida.

Caso Não-Estacionário

A metodologia desenvolvida pressupõe que o filtro adaptativo processa sinais estacionários. Porém, em princípio pode-se modificá-la para otimizar a resposta dos algoritmos de passo variável à não-estacionaridades abruptas, onde os sinais são estacionários dentro de certos intervalos de tempo, sendo a metodologia aproximadamente válida. Com algumas premissas e informações sobre a natureza da não-estacionaridade, a metodologia poderia ser modificada para avaliar a recuperação dos algoritmos após a manifestação da mudança no ambiente adaptativo, e então otimizar a resposta dos filtros quando inseridos nesses contextos.

Referências Bibliográficas

- [1] HAYKIN, S. *Adaptive Filter Theory*. 3. ed., Prentice Hall, 1996.
 - [2] PAPOULIS, A. *Probability, Random Variables and Stochastic Processes*. 3. ed. McGraw-Hill, 1991.
 - [3] WIDROW, B.; STEARNS, S. D. *Adaptive Signal Processing*. Englewood Cliffs, New Jersey: Prentice Hall, 1985.
 - [4] THERRIEN, C. W. *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1992.
 - [5] COWAN, C. F. N.; GRANT, P. M. *Adaptive Filters*. Englewood Cliffs: Prentice Hall, 1985.
 - [6] HOROWITZ, L.; SENNE, K. Performance Advantage of Complex LMS for Controlling Narrow-Band Adaptive Arrays. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. ASSP-29, n. 3 , p. 722-736, June 1981.
 - [7] WIDROW, B.; McCOOL, J. M.; LARIMORE, M. G.; JOHNSON, C. R. Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter. *Proceedings of the IEEE*, v. 64, n. 8, Aug. 1976.
 - [8] HARRIS, R.; CHABRIES, D. M.; BISHOP, F. A. A Variable Step (VS) Adaptive Filter Algorithm. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. ASSP-34, n. 2, p. 309-316, Apr. 1986.
 - [9] GARDNER, W. A. Learning Characteristics of Stochastic-Gradient-Descent Algorithms: A General Study, Analysis, and Critique. *Signal Processing*, North Holland, v. 6, p. 113-133, 1984.
-

-
- [10] KREYSZIG, E. *Advanced Engineering Mathematics*. 7. ed. John Wiley & Sons, 1993.
- [11] FEUER, A.; WEINSTEIN, E. Convergence Analysis of LMS Filters with Uncorrelated Gaussian Data. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. ASSP-33, n. 1, p. 222-230, Feb. 1985.
- [12] MACCHI, O. *Adaptive Processing – The Least Mean Squares Approach with Applications in Transmission*. John Wiley & Sons, 1995.
- [13] BERSHAD, N. J. On the Optimum Gain Parameter in LMS Adaptation. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. ASSP-35, n. 7, p. 1065-1069, July 1987.
- [14] SLOCK, D. On the Convergence Behavior of the LMS and the Normalized LMS Algorithms. *IEEE Trans. on Signal Processing*, v. 41, n. 9, p. 2811-2825, Sept. 1993.
- [15] EVANS, J. B.; XUE, P.; LIU, B. Analysis and Implementation of Variable Step Size Adaptive Algorithms. *IEEE Trans. on Signal Processing*, v. 41, n. 8, p. 2517-2535, Aug. 1993.
- [16] KWONG, R. H.; JONHSTON, E. W. A Variable Step Size LMS Algorithm. *IEEE Trans. Signal Processing*, v. 40, n. 7, p. 1633-1642, July 1992.
- [17] ABOULNASR, T.; MAYYAS, K. A Robust Variable Step-Size LMS-Type Algorithm: Analysis and Simulations. *IEEE Trans. Signal Processing*, vol. 45, n.3, p 631-639, Mar. 1997.
- [18] SHAN, T. J.; KAILATH, T. Adaptive Algorithms with an Automatic Gain Control Feature. *IEEE Trans. on Circuits and Systems*, v. 35, n.1, p 122-127, Jan. 1988
-

-
- [19] OKELLO, J.; ITOH, Y.; FUKUI, Y.; NAKANISHI, I. *et al.* A New Modified Variable Step Size for the LMS Algorithm. In: INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS (May 1998, Monterey). Monterey, CA, v. 5, pp 170-173.
- [20] KUSHNER, H. J.; YANG, J. Analysis of Adaptive Step-Size SA Algorithms for Parameter Tracking. *IEEE Trans. Automatic Control*, v. 40, n. 8, p 1403-1410, Aug. 1995.
- [21] MATHEWS, J. V.; XIE, Z. A Stochastic Gradient Adaptive Filter with Gradient Adaptive Step Size. *IEEE Trans. Signal Processing*, v. 41, n.6, p 2075-2087, June 1993.
- [22] KOIKE, S. A Novel Adaptive Step Size Control Algorithm for Adaptive Filters. In: INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING (Mar.1999: Phoenix), Phoenix, Arizona, v. 4, paper no. 1054 , 1999.
- [23] WALACH, E.; WIDROW, B. The Least Mean Fourth (LMF) Adaptive Algorithm. *IEEE Transactions on Information Theory*, v. 30, p 275-283. Mar. 1984.
- [24] COWAN, F.; RUSU, C. Novel Cost Function Adaptation Algorithm for Echo Cancellation. In: INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, p. 1501-1504, 1998.
- [25] GAN, W. S. Fuzzy step-size adjustment for the LMS algorithm. *Signal Processing*, 49, p. 145-149, 1996.
- [26] _____. Designing a fuzzy step size LMS algorithm, *IEE Proc. Vis. Image Signal Processing*, v. 144, n. 5, p. 261-266, Oct. 1997.
- [27] KARNI, S.; ZENG, G. A New Convergence Factor for Adaptive Filters. *IEEE Trans. on Circuits and Systems*, v. 36, n. 7, July 1989.
-

-
- [28] KUO, S. M.; MORGAN, D. R. *Active Noise Control Systems*. John Wiley & Sons, 1996.
- [29] DUTRA, L. H. *Introdução à Teoria da Ciência*. Florianópolis, SC: Editora da UFSC, 1998.
- [30] JAGGI, S.; MARTINEZ, B. Upper and Lower Bounds of the Misadjustment in the LMS Algorithm. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, v. 38, n. 1, Jan. 1990.
- [31] CARPES Jr., W. P.; SEARA, R.; BERMUDEZ, J. C. M. Análise dos Efeitos da Quantização no Algoritmo LMS em Filtros Digitais Adaptáveis. In: CONGRESSO BRASILEIRO DE AUTOMÁTICA (8.:1990:UFPA), Belém, PA, Brasil, p. 1114-1119.
- [32] KOSKO, B. *Fuzzy Engineering*. Upper Saddle River, New Jersey: Prentice Hall, 1997.
-

Apêndice

Modelo Analítico Recursivo da Matriz $K(n)$ com Passo Variável

A.1. Considerações Iniciais

Na Seção 3.2, em (3.11), foi demonstrado que o modelo analítico do erro médio quadrático do algoritmo LMS depende da evolução da *matriz de covariância do vetor de erro nos coeficientes*:

$$\xi(n) = \xi_{\min} + \text{Tr}\{RK(n)\} \quad (\text{A.1.1})$$

onde:

$$K(n) = E[V(n)V^T(n)] \quad (\text{A.1.2})$$

A partir de algumas hipóteses e resultados difundidos na literatura, será desenvolvido um modelo analítico recursivo para a matriz $K(n)$ que, em conjunto com (A.1.1), descreve o comportamento do erro médio quadrático do algoritmo LMS com passo variável.

Freqüentemente presentes na literatura, a teoria da independência da Seção 3.1 e as hipóteses simplificadoras adicionais da Seção 3.6.3, listadas a seguir por conveniência, facilitam a determinação do comportamento estatístico da matriz $K(n)$:

- *O conjunto $\{X(n), d(n)\}$ é estatisticamente independente do conjunto $\{X(k), d(k)\}$ para $n \neq k$;*

(H.1)

- *O vetor de entradas $X(n)$ e a resposta desejada $d(n)$ são simples realizações de processos estocásticos estacionários conjuntamente gaussianos e com média nula;*

(H.2)

- O passo de cálculo $\mu(n)$ e $\mu^2(n)$ são estatisticamente independentes do vetor de entradas $X(n)$ e do vetor de erro nos coeficientes $V(n)$;

(H.3)

- O valor esperado do passo de cálculo quadrático é aproximadamente igual ao quadrado do valor médio do passo, $E[\mu^2(n)] \approx E^2[\mu(n)]$.

(H.4)

Em decorrência das hipóteses adotadas, em particular a premissa de sinais $x(n)$ e $d(n)$ gaussianos com média nula, torna-se possível a utilização de um resultado conhecido como *teorema da fatoração gaussiana* [6], facilitando o tratamento estatístico das variáveis envolvidas:

“ Se Z_1, Z_2, Z_3 e Z_4 são quatro variáveis (amostras) de um processo gaussiano real e com média nula, então

$$E[z_1 z_2 z_3 z_4] = E[z_1 z_2]E[z_3 z_4] + E[z_1 z_3]E[z_2 z_4] + E[z_1 z_4]E[z_2 z_3] \quad ”$$

Na literatura um dos mais elegantes resultados existentes, que também leva a simplificações importantes, é o *princípio da ortogonalidade* [1], [4]. Este princípio afirma que o erro mínimo $e_o(n)$ remanescente do filtro de Wiener é *ortogonal* ao sinal de referência $x(n)$ no sentido estatístico:

$$E[e_o(n)x(n-p)] = 0 \quad ; \quad \forall p, n \quad (A.1.3)$$

onde o erro mínimo $e_o(n)$ é dado por:

$$e_o(n) = d(n) - W_o^T X(n) \quad (A.1.4)$$

Utilizando apropriadamente os resultados explanados chega-se, após algum algebrismo, à expressão almejada para descrição do comportamento analítico do algoritmo LMS.

A.2. Equação Recursiva da Matriz $K(n)$

Inicia-se a análise pela definição do vetor de erro nos coeficientes e pela equação de atualização do algoritmo LMS, reescritas a seguir:

$$V(n) = W(n) - W_0 \quad (\text{A.2.1})$$

$$W(n+1) = W(n) + \mu(n)e(n)X(n) \quad (\text{A.2.2})$$

Combinando (A.2.1) e (A.2.2) determina-se a equação recursiva do LMS em função do vetor de erro nos coeficientes:

$$V(n+1) = V(n) + \mu(n)e(n)X(n) \quad (\text{A.2.3})$$

No Capítulo 3, definiu-se uma expressão que relaciona o erro de estimação em função do vetor de entradas $X(n)$ e do vetor $V(n)$, permitindo escrever a equação (A.2.3) de forma mais adequada aos propósitos desta seção:

$$e(n) = e_0(n) - V^T(n)X(n) \quad (\text{A.2.4})$$

Aplicando (A.2.4) em (A.2.3):

$$V(n+1) = V(n) - \mu(n)X(n)X^T(n)V(n) + \mu(n)e_0(n)X(n) \quad (\text{A.2.5})$$

Assim, aplicando a definição (A.1.2) na equação recursiva (A.2.5), chega-se a uma equação-diferenças que descreve a evolução da matriz $K(n)$:

$$\begin{aligned} V(n+1)V^T(n+1) &= V(n)V^T(n) + \mu(n)e_0(n)V(n)X^T(n) - \mu(n)V(n)V^T(n)X(n)X^T(n) \\ &\quad + \mu(n)e_0(n)X(n)V^T(n) - \mu^2(n)e_0(n)X(n)V^T(n)X(n)X^T(n) \\ &\quad - \mu(n)X(n)X^T(n)V(n)V^T(n) - \mu^2(n)e_0(n)X(n)X^T(n)V(n)X^T(n) \\ &\quad + \mu^2(n)e_0^2(n)X(n)X^T(n) + \mu^2(n)X(n)X^T(n)V(n)V^T(n)X(n)X^T(n) \end{aligned} \quad (\text{A.2.6})$$

Rearranjando os termos:

$$\begin{aligned}
 V(n+1)V^T(n+1) &= V(n)V^T(n) - \mu(n)[V(n)V^T(n)X(n)X^T(n) + X(n)X^T(n)V(n)V^T(n)] \\
 &+ \mu^2(n)X(n)X^T(n)V(n)V^T(n)X(n)X^T(n) + \mu^2(n)e_o^2(n)X(n)X^T(n) \\
 &+ [\mu(n)e_o(n)X(n)V^T(n) - \mu^2(n)e_o(n)X(n)V^T(n)X(n)X^T(n)] \\
 &+ [\mu(n)e_o(n)X(n)V^T(n) - \mu^2(n)e_o(n)X(n)V^T(n)X(n)X^T(n)]^T
 \end{aligned}
 \tag{A.2.7}$$

Efetando o valor esperado da expressão (A.2.7) e novamente invocando a definição (A.1.2) obtém-se uma expressão da seguinte forma:

$$K(n+1) = K(n) - A + B + C + (D+F) + (D+F)^T \tag{A.2.8}$$

A expressão recursiva (A.2.8) descreve a evolução da matriz de covariância do vetor erro nos coeficientes, carecendo ainda da determinação dos termos que a constituem, o que será feito na seqüência.

A.3.Termo $A \triangleq E[\mu(n)[V(n)V^T(n)X(n)X^T(n) + X(n)X^T(n)V(n)V^T(n)]:$

Recorrendo às hipóteses da independência, o termo A resulta em:

$$A = E[\mu(n)]E[V(n)V^T(n)]E[X(n)X^T(n)] + E[\mu(n)]E[X(n)X^T(n)]E[V(n)V^T(n)] \tag{A.3.1}$$

Das definições (A.1.2) e (2.7):

$$A = E[\mu(n)][K(n)R + RK(n)] \tag{A.3}$$

A.4.Termo $B \triangleq E[\mu^2(n)X(n)X^T(n)V(n)V^T(n)X(n)X^T(n)]$

Este termo envolve momentos de quarta ordem de $x(n)$. Como a teoria da independência assume $x(n)$ gaussiano, os momentos de quarta ordem podem ser reduzidos ao cálculo de momentos de segunda ordem, utilizando o teorema da fatoração gaussiana.

Escrevendo a matriz B na forma:

$$B = \{b_{ij}\} = E \left[\mu^2(n) \sum_{p=1}^N \sum_{m=1}^N (X(n)X^T(n))_{im} (V(n)V^T(n))_{mp} (X(n)X^T(n))_{pj} \right] \quad (A.4.1)$$

onde:

$$(X(n)X^T(n))_{im} = x(n-i)x(n-m) \quad (A.4.2)$$

$$(V(n)V^T(n))_{mp} = v_m(n)v_p(n) \quad (A.4.3)$$

$$(X(n)X^T(n))_{pj} = x(n-p)x(n-j) \quad (A.4.4)$$

Assim:

$$\begin{aligned} \{b_{ij}\} &= E \left[\mu^2(n) \sum_{p=1}^N \sum_{m=1}^N x(n-i)x(n-m)v_m(n)v_p(n)x(n-p)x(n-j) \right] \\ &= E \left[\mu^2(n) \sum_{p=1}^N \sum_{m=1}^N x(n-i)x(n-m)x(n-p)x(n-j)v_m(n)v_p(n) \right] \end{aligned} \quad (A.4.5)$$

Utilizando as hipóteses listadas na Seção A.1:

$$\{b_{ij}\} = E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N E[x(n-i)x(n-m)x(n-p)x(n-j)] E[v_m(n)v_p(n)] \quad (A.4.6)$$

Na expressão (A.4.6), cabe a aplicação do teorema da fatoração ao termo com momentos de quarta ordem em $x(n)$:

$$\begin{aligned} E[x(n-i)x(n-m)x(n-p)x(n-j)] &= E[x(n-i)x(n-m)] E[x(n-p)x(n-j)] \\ &+ E[x(n-i)x(n-p)] E[x(n-m)x(n-j)] \\ &+ E[x(n-i)x(n-j)] E[x(n-m)x(n-p)] \end{aligned} \quad (A.4.7)$$

Sob a premissa de estacionaridade do sinal de referência $x(n)$:

$$E[x(n-i)x(n-m)x(n-p)x(n-j)] = r(m-i)r(j-p) + r(p-i)r(j-m) + r(j-i)r(p-m) \quad (\text{A.4.8})$$

Substituindo (A.4.8) em (A.4.6):

$$\begin{aligned} \{b_{ij}\} &= E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N [r(m-i)r(j-p) + r(p-i)r(j-m) + r(j-i)r(p-m)] k_{mp}(n) \\ &= E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N r(m-i)r(j-p) k_{mp}(n) + E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N r(p-i)r(j-m) k_{mp}(n) \\ &\quad + E[\mu^2(n)] r(j-i) \sum_{p=1}^N \sum_{m=1}^N r(p-m) k_{mp}(n) \end{aligned} \quad (\text{A.4.9})$$

Considerando que $E[v_m(n)v_p(n)] = k_{mp}(n) = (K(n))_{mp}$ e $r(m-i) = (R)_{mi}$:

$$\begin{aligned} \{b_{ij}\} &= E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N (R)_{im} (R)_{pj} (K(n))_{mp} + E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N (R)_{ip} (R)_{mj} (K(n))_{mp} \\ &\quad + E[\mu^2(n)] (R)_{ij} \sum_{p=1}^N \sum_{m=1}^N (R)_{mp} (K(n))_{mp} \end{aligned} \quad (\text{A.4.10})$$

Dado que as matrizes $K(n)$ e R são simétricas, $(K(n))_{mp} = (K(n))_{pm}$ e $(R)_{mp} = (R)_{pm}$, os termos em (A.4.10) podem ser arranjados apropriadamente:

$$\begin{aligned} \{b_{ij}\} &= E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N (R)_{im} (K(n))_{mp} (R)_{pj} + E[\mu^2(n)] \sum_{p=1}^N \sum_{m=1}^N (R)_{ip} (K(n))_{pm} (R)_{mj} \\ &\quad + E[\mu^2(n)] (R)_{ij} \sum_{p=1}^N \sum_{m=1}^N (R)_{pm} (K(n))_{mp} \end{aligned} \quad (\text{A.4.11})$$

Os dois primeiros termos da expressão (A.4.11) claramente correspondem ao termo $(\cdot)_{ij}$ do produto matricial $RK(n)R$. A última somatória

corresponde ao traço do produto $RK(n)$. A expressão (A.4.11) pode então ser reescrita como:

$$\{b_{ij}\} = E[\mu^2(n)](RK(n)R)_{ij} + E[\mu^2(n)](RK(n)R)_{ij} + E[\mu^2(n)](R)_{ij} \text{Tr}\{RK(n)\} \quad (\text{A.4.12})$$

A matriz B correspondente é:

$$B = E[\mu^2(n)][2RK(n)R + R \text{Tr}\{RK(n)\}] \quad (\text{A.4})$$

A.5.Termo $C \triangleq E[\mu^2(n)e_o^2(n)X(n)X^T(n)]$

Novamente, representa-se a matriz C na forma:

$$\{c_{ij}\} = E[\mu^2(n)]E[(e_o(n)e_o(n)X(n)X^T(n))_{ij}] = E[\mu^2(n)]E[e_o(n)e_o(n)x(n-i)x(n-j)] \quad (\text{A.5.1})$$

Como a teoria da independência assume $d(n)$ e $x(n)$ conjuntamente gaussianos, o erro dado por (A.1.4) também o é, desta forma:

$$\begin{aligned} E[e_o(n)e_o(n)x(n-i)x(n-j)] &= E[e_o(n)e_o(n)]E[x(n-i)x(n-j)] \\ &+ E[e_o(n)x(n-i)]E[e_o(n)x(n-j)] + E[e_o(n)x(n-j)]E[e_o(n)x(n-i)] \end{aligned} \quad (\text{A.5.2})$$

O princípio da ortogonalidade determina que somente o primeiro termo não é nulo:

$$E[e_o(n)e_o(n)x(n-i)x(n-j)] = E[e_o^2(n)]r(j-i) = E[e_o^2(n)](R)_{ij} \quad (\text{A.5.3})$$

Substituindo (A.5.3) em (A.5.1):

$$\{c_{ij}\} = E[\mu^2(n)]E[x(n-i)e_o(n)e_o(n)x(n-j)] = E[\mu^2(n)]E[e_o^2(n)](R)_{ij} \quad (\text{A.5.4})$$

A matriz C correspondente resulta em:

$$C = E[\mu^2(n)] \cdot R \cdot \xi_{\min} \quad (\text{A.5})$$

A.6.Termo $D \triangleq E[\mu(n)e_o(n)X(n)V^T(n)]$

Utilizando o princípio da ortogonalidade e a teoria da independência:

$$D = E[\mu(n)]E[e_o(n)X(n)]E[V^T(n)] = [0] \quad (\text{A.6})$$

A.7.Termo $F \triangleq E[\mu^2(n)e_o(n)X(n)V^T(n)X(n)X^T(n)]$

$$\begin{aligned} \{f_{ij}\} &= E \left[\sum_{p=1}^N \mu^2(n)e_o(n) \left(X(n)V^T(n) \right)_p \left(X(n)X^T(n) \right)_p \right] \\ &= E \left[\sum_{p=1}^N \mu^2(n)e_o(n)x(n-i)v_p(n)x(n-p)x(n-j) \right] \end{aligned} \quad (\text{A.7.1})$$

Sob orientação das hipóteses da independência, (A.7.1) resulta em:

$$\{f_{ij}\} = E[\mu^2(n)] \sum_{p=1}^N E[e_o(n)x(n-i)x(n-p)x(n-j)]E[v_p(n)] \quad (\text{A.7.2})$$

Invocando novamente o teorema da fatoração sobre o termo acima:

$$\begin{aligned} E[e_o(n)x(n-i)x(n-p)x(n-j)] &= E[e_o(n)x(n-i)]E[x(n-p)x(n-j)] \\ &+ E[e_o(n)x(n-p)]E[x(n-i)x(n-j)] + E[e_o(n)x(n-j)]E[x(n-i)x(n-p)] \end{aligned} \quad (\text{A.7.3})$$

Pelo princípio da ortogonalidade, todos os valores esperados acima são nulos, portanto, tem-se:

$$\begin{aligned} f_{ij} &= 0 \quad ; \quad \forall i, j \\ F &= \{f_{ij}\} = [0] \end{aligned} \quad (\text{A.7})$$

A.8. Modelo Analítico Recursivo da Matriz $K(n)$

Reunindo as expressões de A, B, C, D e F, derivadas anteriormente, na expressão inicial (A.2.8), tem-se:

$$K(n+1) = K(n) - E[\mu(n)] \cdot [K(n)R + RK(n)] + E[\mu^2(n)] \cdot [2RK(n)R + R \operatorname{Tr}\{RK(n)\} + R \cdot \xi_{\min}] \quad (\text{A.8.1})$$

As expressões (A.1.1) e (A.8.1) descrevem o *comportamento do erro médio quadrático do algoritmo LMS evoluindo com passo variável*.

Com a premissa (H.4), $E[\mu^2(n)] = E^2[\mu(n)]$, chega-se a uma expressão mais simples, utilizada nos Capítulos 3 e 4 :

$$K(n+1) = K(n) - E[\mu(n)] \cdot [K(n)R + RK(n)] + E^2[\mu(n)] \cdot [2RK(n)R + R \operatorname{Tr}\{RK(n)\} + R \cdot \xi_{\min}] \quad (\text{A.8.2})$$

Caso se utilize passo fixo, com $\mu(n) = \mu$, a expressão (A.8.2) se reduz àquela utilizada no Capítulo 3 para derivar a evolução ótima do algoritmo LMS:

$$K(n+1) = K(n) - \mu [K(n)R + RK(n)] + \mu^2 [2RK(n)R + R \operatorname{Tr}\{RK(n)\} + R \cdot \xi_{\min}] \quad (\text{A.8.3})$$