


UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE COMUNICAÇÕES

Este exemplar corresponde à redação final da tese
defendida por Luis Cláudio Coradine
e aprovada pela Comissão
Julgadora em 30 09 93

Orientador

FILTRAGEM ADAPTATIVA EM CASCATA:

PROPOSTA DE ESTRUTURA E ALGORITMO, ANÁLISE E APLICAÇÕES

Tese apresentada à Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de *Doutor em Engenharia Elétrica*.

Autor: Luis Cláudio Coradine n. 1763 / r.c.
Orientador: Prof. Dr. João Marcos Travassos Romano t

Setembro 1993

1993/09

À

Meus Pats, Sheila e

Meus Familiares

AGRADECIMENTOS

Ao Prof. Dr. João Marcos Travassos Romano, pela amizade e orientação deste trabalho.

Aos Professores da FEE, com os quais tive contato.

Aos Funcionários da FEE, pelo auxílio e paciência.

Aos Colegas da UFES e da UNICAMP, pelo apoio.

Aos Amigos, novos e antigos, pela amizade.

À CAPES-PICD e ao CPqD-TELEBRÁS pelo suporte financeiro.

Graças a Deus

RESUMO

A filtragem adaptativa utilizando estruturas na forma em cascata FIR se mostra mais eficaz do que na forma direta (transversal), quando se quer obter os pólos de um modelo autoregressivo, estimando-se diretamente os zeros do preditor adaptativo, a partir dos parâmetros desse filtro, pela solução de equações de segundo grau.

Este trabalho propõe uma técnica de filtragem adaptativa combinando o uso dos algoritmos dos mínimos quadrados recursivos em conjunto com estruturas na forma em cascata FIR. A abordagem consiste em transladar a original estrutura em cascata para uma configuração direta multidimensional FIR, de modo a atualizar os coeficientes do filtro através de um algoritmo dos mínimos quadrados rápido proposto, FLS-MD.

A aplicação do algoritmo em cascata é de extrema importância em situações onde os parâmetros de interesse são os zeros dos filtros, como em predição de voz, detecção de frequências, etc. Nesse sentido, esse algoritmo é proposto para estimação dos parâmetros LSP (Line Spectrum Pairs), utilizados em análise e modelagem de voz. A partir do estudo das relações entre os parâmetros LSP e a predição linear, chega-se a uma estrutura e mostra-se que o algoritmo multidimensional pode ser usado para adaptar seus coeficientes. Simulações são feitas com sinais de voz digitalizados. Também, no caso de detecção de frequências, simulações mostram o bom desempenho da técnica proposta.

A análise de convergência é feita a partir da relação entre as curvas de erro da estrutura proposta e da estrutura transversal, utilizando a seguir a técnica da equação diferencial associada (ODE).

ABSTRACT

Adaptive filtering, using FIR cascade form, is more efficient than the direct (transversal) form when the poles of an autoregressive model are to be obtained, because it directly estimates the adaptive predictor zeros from the filter parameters by the solution of a second degree equation.

This work proposes an adaptive filtering technique that combines the recursive least square algorithms and the FIR cascade form structures. The approach consists in translating the original structure in cascade form to a FIR multidimensional direct configuration, in such a way to update the filter coefficients by using the proposed fast least square multidimensional algorithm, FLS-MD.

The application of this algorithm is very important in situations where the parameters of interest are the zeros of the filters, as in speech prediction, frequency detection, etc. In this way, this algorithm may be applied to estimate the LSP (Line Spectrum Pairs) parameters, used in speech analysis and speech modeling. From the study of the relation between the LSP parameters and the linear prediction, a structure is reached where the multidimensional algorithm can be used to adapt its coefficients. Simulations are carried out with digital speech signal. In the case of frequency detection, simulations also show a good performance of the proposed technique.

The convergence analysis is made from the relation between the error surface of the proposed structure and the transversal structure, using the associated differential equation technique (ODE).

ÍNDICE

Capítulo 1 - Introdução	01
Capítulo 2 - Filtragem Adaptativa com Estrutura Transversal FIR	05
2.1. Filtragem Adaptativa	06
2.2. Estimação Linear Ótima	10
2.2.1. Filtro de Wiener	11
2.2.2. Filtro de Wiener Não Recursivo (FIR)	13
2.2.3. Predição Linear	17
2.2.4. Comentários	25
2.3. Filtragem por Mínimos Quadrados	26
2.3.1. Propriedade dos Estimadores dos Mínimos Quadrados	28
2.4. Algoritmos Adaptativos pela Técnica de Busca	29
2.4.1. O Algoritmo do Gradiente	33
2.4.2. Algoritmos Quase Newton	36
2.5. Algoritmos Recursivos	39
2.5.1. Algoritmo dos Mínimos Quadrados Recursivos - RLS	40
2.5.2. Algoritmos dos Mínimos Quadrados Rápidos - FLS	43
2.6. Discussão sobre o Desempenho dos Algoritmos	44
2.7. Sinais Multidimensionais em Estruturas FIR	49
2.8. Conclusões	53
Capítulo 3 - Filtragem Adaptativa com Estruturas em Cascata FIR	54
3.1. Filtragem Ótima na Forma em Cascata	55
3.1.1. Caso da Predição Linear em Cascata	61
3.2. Técnicas Iterativas para Sistemas Não Lineares	64
3.3. Filtragem Adaptativa na Forma em Cascata	65
3.3.1. Solução Proposta: Configuração Mutidimensional	66
3.4. Algoritmo Rápido Multidimensional: FLS-MD	72
3.4.1. Aplicação à Predição Linear	75
3.5. Simulações e Discussões	77
3.6. Algoritmo Rápido Modificado	80
3.7. Conclusões	85
Capítulo 4 - Análise de Convergência	88
4.1. Proposta de Algoritmo para Estruturas Arbitrárias	89
4.1.1. A Escolha do Gradiente	90
4.1.2. O Algoritmo para Estrutura Arbitrária	92
4.2. O Filtro em Cascata como uma Estrutura Arbitrária	93
4.3. Análise Assintótica de Convergência para Filtros com estruturas Arbitrárias	94

4.4. Análise da Curva de Erro	96
4.4.1. A Função Custo para Filtros com Estruturas Arbitrárias	97
4.5. Aplicação à Forma Cascata	100
4.6. Conclusões	103
Apêndice 4A - O Algoritmo Geral de Adaptação	105
4A1. Dedução do Algoritmo	105
Apêndice 4B - Análise Assintótica da Convergência	110
4B1. O Método ODE	110
4B2. Condições de Regularidade	116
Apêndice 4C - Análise de Convergência para o Método do Erro de Estimação Recursivo	119
Apêndice 4D - Cálculo do Determinante da Matriz Jacobiana	122
Capítulo 5 - Uma Estrutura Adaptativa e Algoritmo Rápido para Modelagem de Voz por Pares de Raias Espectrais (LSP)	125
5.1. Os Parâmetros LSP	126
5.2. Modelagem Adaptativa LSP	129
5.3. Resultados das Simulações	133
5.4. Conclusões	143
Capítulo 6 - Conclusões	144
Bibliografia	147

CAPÍTULO 1

INTRODUÇÃO

A filtragem adaptativa tem sido largamente usada como técnica de processamento de sinais digitais, tornando-se cada vez mais sofisticada com os avanços dessa teoria e da tecnologia dos circuitos integrados. Diferentes pontos de vista teóricos são usados como forma de introduzir o problema da filtragem adaptativa, destacando-se a abordagem da aproximação estocástica, feita a partir da busca iterativa sobre a superfície de erro, e a técnica recursiva, a partir da solução exata dos estimadores dos mínimos quadrados.

A estrutura do filtro a ser empregada é de fundamental importância no processo adaptativo, pois define as condições de operação do algoritmo como um todo.

Os diversos caminhos para análise e classificação de algoritmos e estruturas na filtragem adaptativa têm originado excelentes trabalhos nas últimas décadas. Os fundamentos teóricos comuns a esses trabalhos também estão bem estabelecidos e recolhidos em diferentes livros-textos, como, por exemplo, as referências [1-3].

O método do gradiente (LMS) é o mais popular dos algoritmos estocásticos, tendo baixa complexidade computacional, mas apresentando, em contrapartida, uma convergência lenta quando o espalhamento da matriz de correlação é elevado. Já o algoritmo dos mínimos quadrados recursivos (RLS) possui uma convergência rápida e boa capacidade de rastreamento, tendo, porém, uma complexidade computacional mais elevada. A sua versão rápida, o FLS, o torna altamente competitivo em relação ao LMS, embora com problemas de precisão finita [4].

Os filtros adaptativos na forma direta de estruturas transversais FIR, como também na forma em treliça FIR, são os mais comumente usados em processamento de sinal, os quais respondem muito bem em termos de convergência e estabi-

lidade. Seus coeficientes são obtidos como solução de um problema de otimização linear, sendo sua curva do erro convexa em relação aos coeficientes, o que permite a busca do mínimo global. Em outros tipos de estruturas, aparece o problema da não linearidade no processo de otimização, levando a soluções aproximadas para a determinação do filtro ótimo. Estruturas recursivas (IIR) têm sido apresentadas com o objetivo de diminuir o custo computacional dos filtros FIR [5,6], reduzindo de maneira considerável a ordem do filtro, embora introduzindo grandes problemas de estabilidade, requerendo monitoramento dos pólos durante o processo de adaptação, além do risco de convergir para mínimos locais. De uma forma geral, é em função de uma aplicação específica que se define o algoritmo e a estrutura de um filtro adaptativo.

Este trabalho propõe uma técnica de filtragem adaptativa combinando o uso dos algoritmos dos mínimos quadrados recursivos em conjunto com estruturas na forma em cascata FIR, estudada preliminarmente nos trabalhos de Jackson e Wood [7]. A abordagem, aqui, consiste em transladar a original estrutura em cascata para uma configuração direta multidimensional FIR, de modo a atualizar os coeficientes do filtro. A extensão para algoritmos rápidos, FLS, é derivada para ambos os casos de predição linear e identificação.

As vantagens da decomposição dos filtros em estruturas na forma em cascata, tanto para filtros IIR quanto para filtros FIR, são conhecidas e comentadas na literatura de processamento de sinais [8]. Pode-se citar a pequena sensibilidade dos coeficientes à quantização, a facilidade no monitoramento da estabilidade, etc. Além disso, sua aplicação é de extrema importância em situações onde os parâmetros de interesse são os zeros dos filtros, ao invés dos coeficientes. Esse é o caso de algumas técnicas de predição de voz [9,10], em detecção adaptativa de senóides ruidosas [11] e, em geral, em análise autoregressiva, quando se deseja acesso aos pólos de um modelo [12].

De fato, o cálculo dos zeros de um filtro transversal FIR, os quais correspondem às raízes da função de transferência, requer complicadas técnicas iterativas se tivermos mais que dois coeficientes. No contexto adaptativo, esses cálculos devem ser feitos a cada iteração n , tornando-se praticamente impossível em muitos casos. A estrutura proposta, então, é formada por secções de segunda ordem em cascata, uma vez que os zeros de cada secção podem ser facilmente obtidos pela solução de equações de segundo grau. Pode-se, em particular, obter os pólos de um modelo autoregressivo, estimando diretamente os zeros do preditor adaptativo, implementado na forma em cascata [13,14].

A relação entre os coeficientes da forma em cascata e os da forma direta é feita por um mapeamento não linear, levando a se resolver, no caso em cascata, equações normais não lineares [7]. Porém, essas equações provêm da mesma minimização da norma do erro, como na forma direta, podendo ser mostrado que a solução é única, a menos de permutações entre as secções. A abordagem multidimensional, proposta neste trabalho, permite a linearização do problema e a conseqüente aplicação de técnicas do tipo FLS.

Como uma interessante conseqüência do estudo dos filtros em cascata, propõe-se, também, uma técnica adaptativa para obter os parâmetros LSP (Line Spectrum Pairs) de um sinal. A modelagem de sinais por coeficientes LSP foi proposta para codificação de voz [15], tendo como motivação a superioridade desses coeficientes com relação ao LPC e ao PARCOR no que se refere à robustez aos erros de quantização [16,17]. Mostra-se que a determinação dos parâmetros LSP passa pela implementação de filtros em cascata, que visam localizar raias espectrais cujas freqüências são obtidas a partir da posição dos zeros desses filtros.

À guisa de apresentar as diferentes contribuições e a base teórica necessária, este trabalho está organizado como descrito a seguir.

No Capítulo 2, é feita uma revisão dos conceitos de filtragem ótima, introduzindo-se a teoria de filtragem adaptativa, tendo em vista tanto os algoritmos baseados nas técnicas de busca como nos métodos de mínimos quadrados. Além disso, faz-se uma discussão das estruturas e apresenta-se o algoritmo rápido multidimensional.

O cálculo do filtro ótimo na estrutura em cascata é desenvolvido no Capítulo 3, com ênfase à predição linear, onde se coloca a proposta da configuração adaptativa multidimensional e, em acordo com essa abordagem, o uso do algoritmo rápido multidimensional, FLS-MD. Propõe-se também um algoritmo híbrido onde o FLS-MD é aplicado em conjunto com técnicas de tipo Gauss-Newton visando melhorar a convergência inicial. A detecção de freqüência em meio ao ruído, utilizando preditores em cascata, é talvez a aplicação mais relevante das técnicas propostas. Nesse sentido, diversos resultados de simulação são apresentados mostrando o desempenho do método em termos de rapidez de convergência, inclusive para os casos críticos de existência de várias senóides.

No Capítulo 4, faz-se a análise de convergência relativa ao algoritmo proposto. Essa análise se baseia nos resultados de Nayeri e Forssén [18, 19] que relacionam a curva de erro de uma estrutura arbitrária à da estrutura transver-

sal. Particulariza-se, então, esse resultado aos filtros em cascata FIR, mostrando que, embora sua curva de erro possua vários mínimos, esses são todos equivalentes ao mínimo global processado. A partir desse resultado, utiliza-se, também, a técnica da equação diferencial associada (ODE) [5] para garantir a convergência do algoritmo proposto.

A aplicação do algoritmo para a estimação adaptativa de parâmetros LSP, utilizados em análise e modelagem de voz é desenvolvida no Capítulo 5. O algoritmo é implementado numa estrutura constituída por uma cascata de secções de 2ª ordem com fase linear. A partir do estudo das relações entre os parâmetros LSP e a predição linear, chega-se à estrutura proposta e mostra-se que o algoritmo multidimensional (FLS-MD) pode ser usado para adaptar seus coeficientes. O método é avaliado através de resultados de simulações obtidos com sinais de voz digitalizados, onde se compara os resultados teóricos conseguidos com uma análise por bloco dos dados, constatando-se que o algoritmo fornece corretamente os parâmetros desejados e apresenta excelentes propriedades de rastreamento e estabilidade.

Finalmente, no Capítulo 6, as conclusões gerais do trabalho são discutidas assim como possíveis extensões.

CAPÍTULO 2

FILTRAGEM ADAPTATIVA COM ESTRUTURA TRANSVERSAL FIR

No contexto da filtragem linear, tem-se a idéia básica de que a versão ruidosa de um determinado sinal é observada e tenta-se estimar o sinal verdadeiro. Essa estimativa é feita através de um filtro, cuja saída é comparada ao sinal desejado. A teoria é apresentada em termos de se desenvolver um estimador, baseado no conhecimento estatístico dos sinais envolvidos, que melhor ajuste um determinado critério. Assim, a otimização desse critério deve levar a uma melhor comparação entre o sinal de referência e a sua estimativa.

A técnica da filtragem adaptativa é introduzida para suprir algumas dificuldades na obtenção do filtro digital ótimo em ambientes de estatísticas desconhecidas. Esta técnica mostra-se imprescindível em certos processamentos em tempo real, bem como na análise de sinais não estacionários.

De uma forma geral, as soluções analíticas podem servir como balizadores e, também, como instrumentos para a definição de critérios, estruturas e algoritmos, a serem aplicados nas técnicas adaptativas. Assim, os conceitos de filtragem de Wiener, predição linear, filtragem de Kalman e estimadores de mínimos quadrados são importantes, não só pelos conceitos propriamente ditos, mas na análise do desempenho dos filtros adaptativos.

A filtragem adaptativa tem como princípio a variação dos parâmetros do filtro, ao longo do processo, de acordo com critérios pré-estabelecidos. A qualidade do sinal de saída é avaliada e essa informação é passada para um sistema, o qual a usa para decidir como modificar os parâmetros do filtro, com a finalidade de melhorar o próprio sinal de saída. A estrutura de filtro a ser usada e o algoritmo empregado em sua adaptação constituem os problemas fundamentais em filtragem adaptativa.

Nesse sentido, o objetivo deste capítulo é colocar de maneira geral e unificada, a base teórica necessária em processamento adaptativo para que se possa abordar as propostas subsequentes. O problema da filtragem linear ótima será discutido, partindo-se da teoria de Wiener e do critério de mínimos quadrados. Em seguida, os algoritmos adaptativos básicos são introduzidos para ambos

os enfoques, usando-se a idéia de aproximação estocástica e a técnica recursiva de mínimos quadrados, mostrando-se as analogias entre elas.

Todo o desenvolvimento apresentado neste capítulo se baseia nas estruturas FIR em sua forma transversal, destacando-se também o caso multidimensional. A partir daí, tem-se os elementos necessários para abordar o caso em cascata.

2.1. FILTRAGEM ADAPTATIVA

Filtros adaptativos são dispositivos auto-ajustáveis, baseados em algoritmos recursivos, que modificam seus parâmetros de acordo com critérios pré-estabelecidos. Em ambiente estacionário, convergem para a solução ótima do problema. Em ambiente não estacionário, acompanham as modificações do sinal envolvido, desde que essas variações sejam suficientemente lentas.

O esquema de uma filtragem adaptativa é apresentado na Figura 2.1,

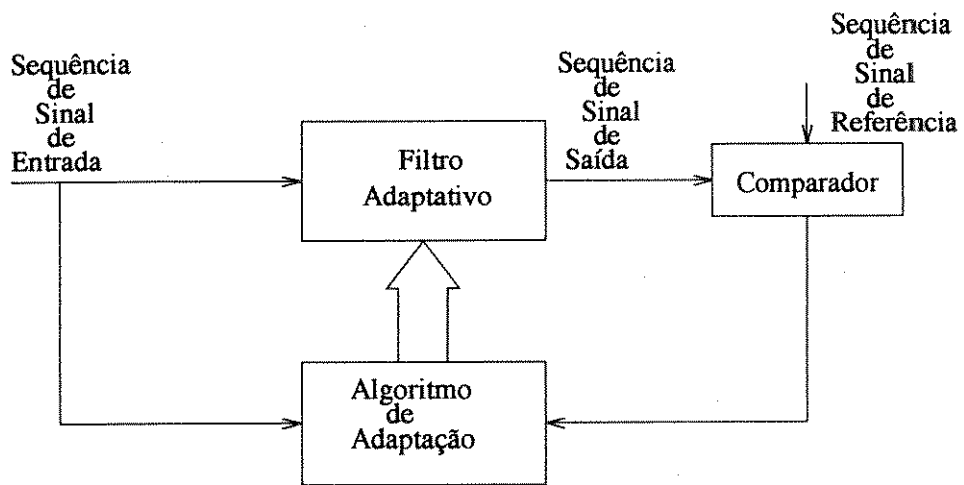


Fig. 2.1- A Filtragem Adaptativa

onde aparecem os quatro elementos básicos de sua operação: o critério de otimização, o tipo de sinal processado, o processo de filtragem e o processo de adaptação.

O critério de otimização é, a princípio, uma função quadrática do erro de estimação. É importante conhecer o comportamento da superfície associada a esse critério, principalmente quando existem mínimos locais ou pontos de sela.

O tipo de sinal processado é inerente a cada problema, podendo ser mono ou multidimensional, estacionário ou não, etc. Qualquer informação sobre o comportamento do sinal se mostra de grande valia na análise do sistema.

O processo de filtragem está diretamente relacionado à estrutura do filtro, a qual busca melhor produzir um sinal de saída como resposta a uma sequência de dados de entrada. O sinal de saída contém informações que balizam o processo de adaptação.

O processo de adaptação define o mecanismo de controle do conjunto de parâmetros do filtro. Esse mecanismo é desenvolvido na forma de algoritmo, com base em algum critério de otimização.

Assim, os processos de adaptação e filtragem interagem durante toda a filtragem adaptativa, sendo que a escolha da estrutura do filtro para o processo de filtragem tem importantes conseqüências sobre a operação do algoritmo como um todo.

Uma variedade de algoritmos recursivos para estimação dos parâmetros de um filtro adaptativo é apresentada na literatura [1, 2, 3], sendo que a sua escolha é determinada por vários fatores:

- Rapidez de convergência
- Medida de desajuste do erro mínimo
- Habilidade de rastreamento
- Robustez ao ruído
- Complexidade computacional
- Consistência da rapidez de convergência
- Estabilidade numérica

Esses fatores fazem parte da análise do desempenho do algoritmo.

A forma geral de adaptação dos parâmetros pode ser descrita como:

$$\begin{bmatrix} \text{novo} \\ \text{parâmetro} \end{bmatrix} = \begin{bmatrix} \text{parâmetro} \\ \text{anterior} \end{bmatrix} + \begin{bmatrix} \text{função de atuali-} \\ \text{zação baseada nos} \\ \text{dados de entrada,} \\ \text{referência e do} \\ \text{erro} \end{bmatrix}$$

onde a função de atualização é definida pelo algoritmo recursivo. Essa visão unificada é bem apresentada por Ljung [5].

De uma forma geral, duas diferentes classes de algoritmos de adaptação são desenvolvidas para resolver o problema da filtragem adaptativa. A primeira advém dos métodos de busca iterativa, representados pelos algoritmos Gauss-Newton e pelo gradiente estocástico (o popular LMS). A outra é a versão adaptativa dos estimadores de mínimos quadrados, ou algoritmos dos *mínimos quadrados recursivos* (RLS).

O LMS (Least Mean Square) é um método aproximado, de baixa complexidade computacional, porém com convergência excessivamente lenta, sendo sensível ao espalhamento da matriz de correlação, ou seja, quanto mais correlacionado é o sinal, menos eficiente é o algoritmo.

O RLS provém da exata solução dos mínimos quadrados (LS), para cada atualização no tempo. Tem convergência rápida e boa capacidade de rastreamento, possuindo porém uma alta complexidade computacional. O desenvolvimento de técnicas computacionais rápidas, por exemplo FLS (Fast Least Square) tornam os algoritmos dos mínimos quadrados altamente competitivos com o LMS, embora apresentem problemas de precisão finita [4].

Além das estruturas transversais FIR (Finite Impulse Response), outras estruturas podem ser exploradas, como por exemplo: FIR cascata, paralelo, treliça e filtros com *resposta impulsiva infinita* (IIR) [8].

Os algoritmos de adaptação apresentados neste capítulo utilizam a estrutura não recursiva (FIR), que tendem a convergir para a solução ótima de Wiener.

A filtragem adaptativa tem sido aplicada com sucesso em diversas áreas como comunicações, controle, engenharia biomédica, etc. Os modos de operação podem ser divididos em quatro grandes grupos, a seguir:

- a- Identificação: Modelagem de um sistema na forma direta
- b- Modelagem Inversa
- c- Predição
- d- Cancelamento de Interferência

os quais são ilustrados na Figura 2.2.

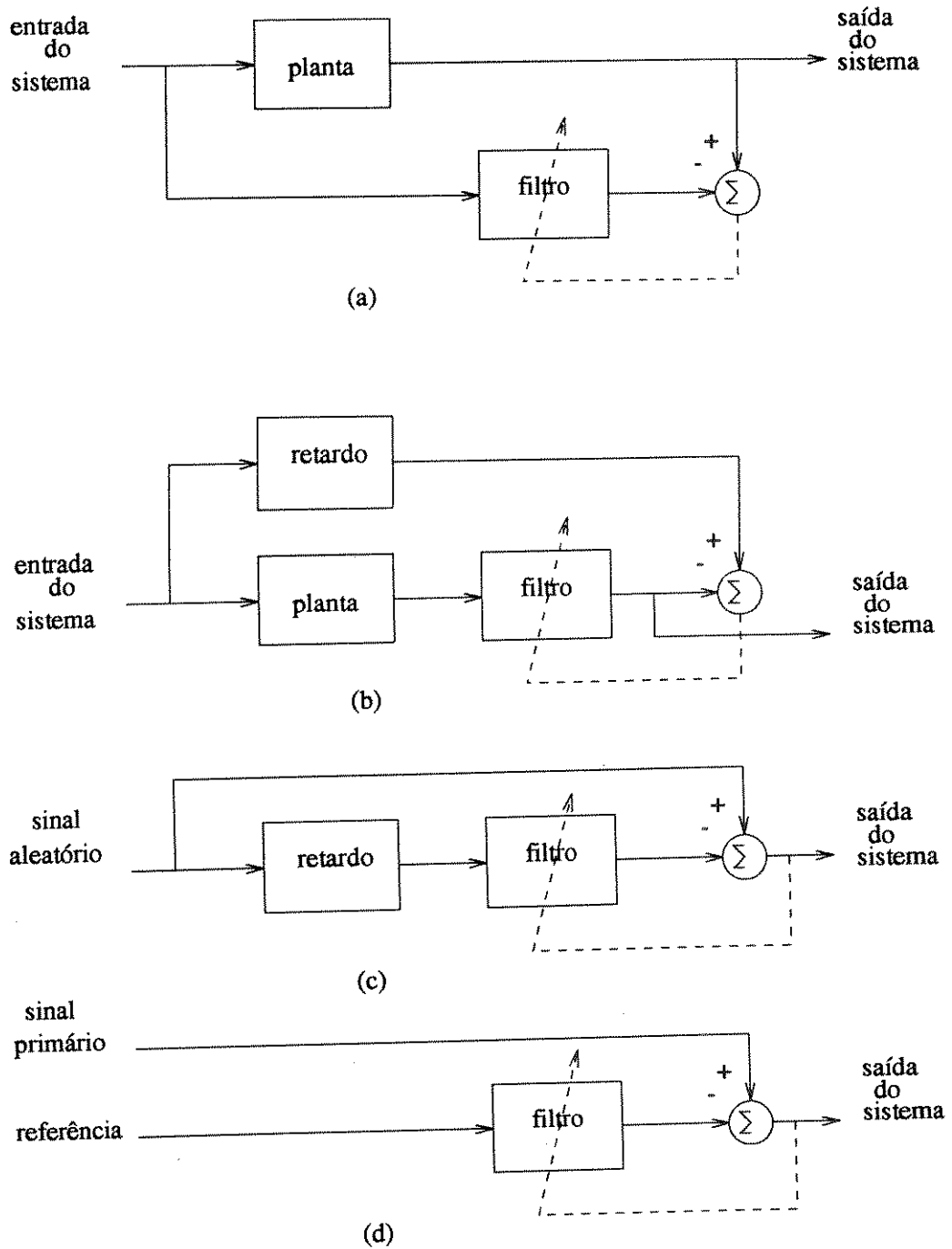


Fig 2.2 - Tipos de aplicação da filtragem adaptativa [1]

Como consequência da aplicação de um algoritmo recursivo para a estimação dos parâmetros do filtro adaptativo, tais parâmetros são dependentes dos sinais processados. Assim, um filtro adaptativo é um dispositivo não linear e de coeficientes variantes no tempo.

2.2. ESTIMAÇÃO LINEAR ÓTIMA

A estrutura de um problema típico de estimação linear discreta de um sinal é ilustrada na Figura 2.3 [1]. A partir de uma seqüência de dados observada $\{x(n)\}$, deve-se encontrar um filtro linear discreto, $H(z)$ que opere sobre $\{x(n)\}$ fornecendo, na saída, a seqüência estimada $\{\hat{y}(n)\}$. O critério de qualidade é uma função, $f(\cdot)$, do erro de estimação $\{e(n)\}$, o qual é a diferença entre a seqüência $\{y(n)\}$, sinal desejado ou de referência, e a seqüência $\{\hat{y}(n)\}$ que é o sinal estimado:

$$e(n) = y(n) - \hat{y}(n) \quad (2.1)$$

Assim, a função perda $f(e(n))$ define a penalidade que se tem quando a estimação é incorreta. Essa função deve ser positiva e não decrescente, por exemplo: função quadrática, função valor absoluto, etc. [20].

Considerando $\{x(n)\}$ e $\{y(n)\}$ como seqüências aleatórias, tem-se que $\{e(n)\}$ também é uma seqüência aleatória. Assim, a melhor escolha do filtro linear é tal que uma função estatística do erro de estimação (função custo) seja minimizada:

$$I(e(n)) = E(f(e(n))) \quad (2.2)$$

onde $E\{\cdot\}$ define o valor esperado.

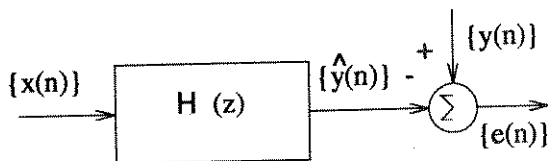


Fig.2.3 - Estimador linear ótimo

Um critério estatístico comumente usado é o valor quadrático médio do erro de estimação:

$$J = E\{e^2(n)\} \quad (2.3)$$

também chamado de erro quadrático médio, que apresenta facilidades no tratamento matemático. O problema, então, se resume em calcular os coeficientes do filtro, que são ótimos no sentido dos mínimos médios quadrados.

2.2.1. Filtro de Wiener

O sistema de filtragem ótima apresentado na Figura 2.3 é conhecido como filtro de Wiener, desde que o critério a ser minimizado seja o erro quadrático médio, e se conheça, a priori, as estatísticas conjuntas dos sinais de entrada $\{x(n)\}$ (recebido), e de referência $\{y(n)\}$ (transmitido), com a necessidade de que ambos sejam estacionários.

A saída $\hat{y}(n)$ do filtro discreto linear causal pode ser descrita como a convolução entre a seqüência de entrada e a seqüência de resposta impulsiva do filtro $\{h_i\}$:

$$\hat{y}(n) = \sum_{i=0}^{\infty} h_i x(n-i) \quad n = 0, 1, 2, \dots \quad (2.4)$$

o qual é, por definição, um filtro de resposta impulsiva infinita [8].

Supondo que a seqüência $\{x(n)\}$ de entrada e a seqüência $\{y(n)\}$ do sinal desejado sejam realizações simples de processos estocásticos conjuntamente estacionários no sentido amplo, o problema está em determinar as condições de operação do filtro tais que a função J atinja o seu valor mínimo. Para se obter o ponto de mínimo da função custo J , toma-se o gradiente dessa função com relação aos coeficientes do filtro, de forma que, nesse ponto da função (ponto estacionário), os componentes do vetor gradiente $\nabla(J)$, dados por:

$$\nabla_i(J) = \partial J / \partial h_i \quad i = 0, 1, 2, \dots \quad (2.5)$$

sejam todos nulos, simultaneamente. Assim, tomando-se a expressão de J da equação 2.3, tem-se o gradiente para cada coeficiente:

$$\frac{\partial J}{\partial h_i} = 2E \left\{ \frac{\partial e(n)}{\partial h_i} \cdot e(n) \right\} \quad (2.6)$$

Devido ao fato de $y(n)$ não ser função dos h_i 's, substituindo a equação 2.4 na equação 2.6, escreve-se:

$$\frac{\partial J}{\partial h_i} = -2E \left\{ \frac{\partial \hat{y}(n)}{\partial h_i} \cdot e(n) \right\} = -2E \{x(n-i) \cdot e(n)\} \quad (2.7)$$

No ponto de equilíbrio da função J:

$$E \{x(n-i) \cdot e(n)\} = 0 \quad i = 0, 1, 2, \dots \quad (2.8)$$

mostrando que o erro, na condição ótima, é ortogonal aos dados de entrada. Escreve-se, assim, "o princípio da ortogonalidade":

"A condição necessária e suficiente para que a função custo J atinja seu ponto de mínimo é que o valor correspondente do erro, $e(n)$, seja ortogonal a cada amostra do sinal de entrada que integra a estimação de uma resposta desejada no instante n".

Como consequência desse princípio, verifica-se que o erro de estimação é ortogonal à saída estimada na condição ótima, ou seja:

$$E \left\{ \hat{y}_{ot}(n) \cdot e_{ot}(n) \right\} = 0 \quad (2.9)$$

Também, a partir do princípio da ortogonalidade, o erro quadrático médio mínimo é expresso, em relação à variância do sinal de referência e à variância do sinal estimado, da forma:

$$J_{\min} = E\{y^2(n)\} - E\{\hat{y}_{ot}^2(n)\} \quad (2.10)$$

Usando a equação 2.4 e a equação 2.8, pode-se escrever:

$$E \left\{ x(n-i) \cdot \left(y(n) - \sum_{k=0}^{\infty} h_{otk} \cdot x(n-k) \right) \right\} = 0 \quad i = 0, 1, 2, \dots \quad (2.11)$$

onde h_{otk} é o k-ésimo coeficiente do filtro ótimo. Expandindo-se essa equação e

rearranjando os termos, tem-se:

$$\sum_{k=0}^{\infty} h_{otk} \cdot E\{x(n-i) x(n-k)\} = E\{x(n-i) \cdot y(n)\} \quad i = 0, 1, 2, \dots \quad (2.12)$$

ou, ainda,

$$\sum_{k=0}^{\infty} h_{otk} \cdot r(k-i) = p(-i) \quad i = 0, 1, 2 \dots \quad (2.13)$$

que são as equações normais, que equivalem, no caso discreto, às equações de Wiener-Hopf no caso contínuo, onde

$$r(k-i) = E\{x(n-i) \cdot x(n-k)\} \quad (2.14)$$

é a função de autocorrelação dos dados de entrada para um atraso (k-i), e

$$p(-i) = E\{x(n-i) \cdot y(n)\} \quad (2.15)$$

a correlação cruzada entre a entrada $x(n-i)$ e o sinal desejado. Dessa forma as soluções das equações 2.13 definem o filtro ótimo.

2.2.2. Filtro de Wiener Não Recursivo (FIR)

Não é possível implementar o filtro de Wiener na forma descrita anteriormente, a não ser através de filtros de *resposta impulsiva infinita* (IIR) com recursão do sinal de saída, que serão discutidos posteriormente.

Seja $H(z)$ um filtro transversal (não recursivo) com $\hat{y}(n)$ baseado em um número finito de observações do sinal de entrada $x(n)$:

$$\hat{y}(n) = \sum_{i=0}^{M-1} h_i x(n-i) \quad (2.16)$$

onde h_i são os coeficientes do filtro. Na forma vetorial, tem-se o produto interno:

$$\hat{y}(n) = \underline{H}^T \cdot \underline{X}(n) \quad (2.17)$$

onde \underline{H} é o vetor coluna dos coeficientes h_i ,

$$\underline{H} = [h_0, h_1, \dots, h_{M-1}]^T \quad (2.18)$$

e $\underline{X}(n)$, o vetor coluna dos dados passados,

$$\underline{X}(n) = [x(n), x(n-1), \dots, x(n-M+1)]^T \quad (2.19)$$

O erro de estimação é dado por:

$$e(n) = y(n) - \underline{H}^T \cdot \underline{X}(n) \quad (2.20)$$

A Figura 2.4 mostra a estrutura do filtro de Wiener transversal FIR, composto de (M-1) armazenamentos, M multiplicações e uma adição. Diz-se que é um filtro de ordem (M-1).

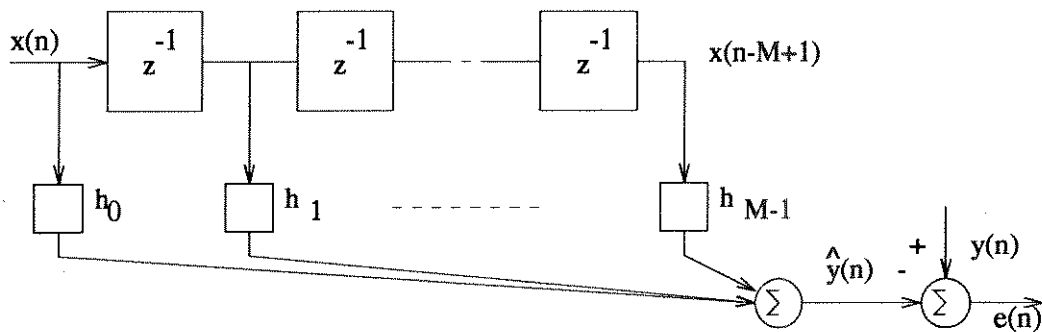


Fig. 2.4 - Filtro de Wiener FIR

- Observação:

Em alguns casos, tem-se um filtro com múltiplas entradas, de forma que

$$\hat{y}(n) = \sum_{k=0}^{L-1} h_k \cdot x_{kn} \quad (2.21)$$

ou, na forma vetorial,

$$y(n) = \underline{H}^T \cdot \underline{X}_n \quad (2.22)$$

com \underline{X}_n sendo um vetor espacial da forma

$$\underline{X}_n = [x_{0n}, x_{1n}, \dots, x_{(L-1)n}]^T \quad (2.23)$$

tal que o erro seja, como na equação 2.20,

$$e(n) = y(n) - \underline{H}^T \cdot \underline{X}_n \quad (2.24)$$

A figura 2.5 ilustra esse caso:

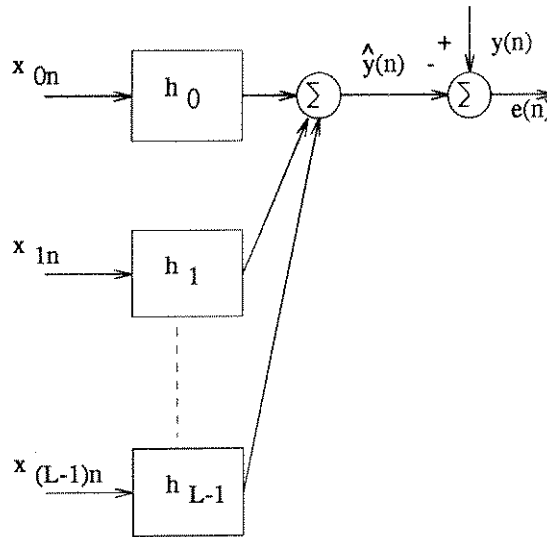


Fig. 2.5 - Filtro de Wiener Espacial

Os critérios para a obtenção do filtro ótimo são os mesmos do filtro FIR.

□

Como no caso geral do filtro de Wiener, tem-se a determinação do filtro ótimo a partir do ponto de mínimo da função de custo $J(\underline{H})$, observando as mesmas condições de $\{x(n)\}$ e $\{y(n)\}$:

$$J(\underline{H}) = E\{(y(n) - \underline{H}^T \cdot \underline{X}(n))^2\} \quad (2.25)$$

ou ainda,

$$J(\underline{H}) = E\{y^2(n)\} + \underline{H}^T \underline{R}_{xx} \cdot \underline{H} - 2\underline{H}^T \cdot \underline{R}_{xy} \quad (2.26)$$

onde

$$\mathbf{R}_{xx} = E\{\underline{X}(n) \cdot \underline{X}^T(n)\} = \begin{bmatrix} r(0) & \dots & r(M-1) \\ r(-M+1) & \dots & r(0) \end{bmatrix} \quad (2.27)$$

é a matriz de autocorrelação (MxM) do sinal de entrada e

$$\mathbf{R}_{xy} = E\{\underline{X}(n) \cdot y(n)\} = [p(0) \dots p(1-M)]^T \quad (2.28)$$

o vetor de correlação cruzada, (Mx1), entre a amostra instantânea do sinal de referência e as amostras do sinal de entrada.

Nota-se que, para o filtro FIR, a função custo $J(\underline{H})$ é uma função quadrática dos coeficientes do filtro, também descrita por:

$$J(\underline{H}) = r_{yy}(0) + \sum_{k=0}^{M-1} \sum_{i=0}^{M-1} h_k h_i r(k-i) - 2 \sum_{i=0}^{M-1} h_i p(-i) \quad (2.29)$$

onde

$$r_{yy}(0) = \sigma^2 \quad (2.30)$$

é a variância do sinal de referência, supondo a média nula. A função J é uma superfície convexa de dimensão (M+1).

O gradiente da função custo é determinado diretamente da equação 2.26 por:

$$\underline{\nabla}(J) = 2\mathbf{R}_{xx} \underline{H} - 2\mathbf{R}_{xy} \quad (2.31)$$

e o filtro ótimo é dado pela solução única do sistema de equações normais, no ponto do gradiente nulo:

$$\mathbf{R}_{xx} \underline{H}_{ot} = \mathbf{R}_{xy} \quad (2.32)$$

ou seja

$$\underline{H}_{ot} = \mathbf{R}_{xx}^{-1} \cdot \mathbf{R}_{xy} \quad (2.33)$$

onde \underline{H}_{ot} é o vetor de coeficientes do filtro ótimo.

O valor mínimo da função custo é obtido substituindo-se a equação 2.33 na equação 2.26 de forma que:

$$J_{\min} = E\{y^2(n)\} - \underline{H}_{ot}^T \cdot \underline{R}_{xy} \quad (2.34)$$

É importante frisar que, para se determinar a solução da equação 2.33, é necessário obter a inversa da matriz de autocorrelação. Considerando que a sequência $\{x(n)\}$ é um processo estocástico estacionário, a matriz de autocorrelação \underline{R}_{xx} é simétrica, Toeplitz e positiva definida (para um sistema persistentemente excitado, $\underline{X} \neq 0$) [1]. Assim, devido ao fato de ser positiva definida, a matriz \underline{R}_{xx} é também não-singular, existindo necessariamente a matriz inversa \underline{R}_{xx}^{-1} .

O cálculo do vetor \underline{H}_{ot} a partir da equação 2.32 requer a solução de M equações lineares simultâneas com M incógnitas. Para uma matriz não-singular, o método mais eficiente de solução é o de Eliminação Gaussiana, que requer operações computacionais da ordem de M^3 .

Um método recursivo de solução direta das equações normais, sem necessidade da inversão da matriz de autocorrelação, é o algoritmo de Levinson [21] que utiliza o fato da matriz de autocorrelação ser simétrica e Toeplitz. A propriedade da matriz ser Toeplitz implica em que todos os seus elementos possam ser gerados a partir do conhecimento da sua primeira linha. Assim, o algoritmo de Levinson é uma técnica eficiente para se obter a solução ótima, requerendo operações na ordem de M^2 .

Métodos iterativos indiretos também são apresentados, levando-se em conta que o problema da filtragem ótima pode ser visto como a minimização de uma função não linear $J(\underline{H})$. Algoritmos de busca iterativa, como o de máxima descida ("steepest descent"), e o de Newton, podem ser utilizados independentes do fato da matriz de autocorrelação ser Toeplitz [22].

2.2.3. Predição Linear

A predição linear pode ser vista como um caso particular da filtragem ótima, sendo uma das ferramentas básicas utilizadas em vários aspectos da análise de voz, estimação espectral, modelagem de sinal e filtragem adaptativa [1, 9, 10, 11, 12].

Na predição linear busca-se estimar um sinal estocástico $\{x(n)\}$, sinal recebido, diretamente a partir de suas amostras passadas e de suas propriedades estatísticas. Dois pontos são fundamentais para essa análise: (i) a estrutura do filtro usado e (ii) o desempenho do sistema, caracterizado pelo comportamento do erro de predição.

Uma revisão dos conceitos e propriedades da predição linear é feita, a seguir, ressaltando sua importância como técnica de análise de sinal.

2.2.3.1. Conceitos Básicos

Tomando as mesmas considerações anteriores da seqüência de dados $\{x(n)\}$, a predição linear pode ser vista como um caso particular de filtragem de Wiener, sendo que o sinal de referência é o próprio sinal de entrada do filtro, chamado filtro preditor. Dessa forma, o sinal estimado (sinal predito) é comparado com o sinal observado, tendo, na saída, o erro de predição. A Figura 2.6 ilustra o sistema, conhecido como filtro de erro de predição.

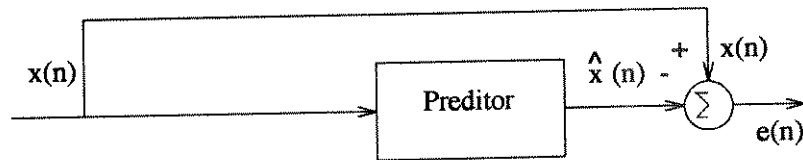


Fig 2.6 - Filtro de Erro de Predição

Assim, o erro de predição, com passo unitário e passado infinito é descrito por:

$$e(n) = x(n) - \hat{x}(n) = x(n) - \sum_{i=1}^{\infty} a_i x(n-i) \quad (2.35)$$

onde os a_i 's ótimos, coeficientes do filtro preditor, são obtidos a partir da minimização da função custo $J = E\{e(n)\}$, potência do erro, tal que:

$$\frac{\partial J}{\partial a_i} = 0 \quad i = 1, 2, \dots \quad (2.36)$$

o que implica em

$$E\{x(n-i) e(n)\} = 0 \quad (2.37)$$

que, como foi visto anteriormente, indica a ortogonalidade entre as amostras $x(n-i)$ e o erro $e(n)$. Levando-se em conta que o erro $e(n-i)$ é descrito como combinação linear infinita da seqüência $\{x(n-i)\}$, obtém-se que:

$$E\{e(n-i) e(n)\} = 0 \quad i = 1, 2, \dots \quad (2.38)$$

ou seja, que a saída do filtro de erro de predição infinito é um sinal totalmente decorrelatado. O filtro de erro de predição funciona, então, como um filtro branqueador.

A partir da transformada Z da equação 2.35, escreve-se a função de transferência do filtro de erro de predição

$$P(z) = 1 - A(z) = 1 - \sum_{i=1}^{\infty} a_i z^{-i} \quad (2.39)$$

com $A(z)$ sendo o filtro preditor.

A variância do erro de predição, J (função custo) pode ser calculada a partir da densidade espectral de potência dos dados $S_x(e^{j\omega})$, pela expressão:

$$J = \frac{1}{2\pi} \int_{-\pi}^{\pi} |P(e^{j\omega})|^2 S_x(e^{j\omega}) d\omega \quad (2.40)$$

ou ainda, em termos da transformada Z,

$$J = \frac{1}{2\pi} \int_{-\pi}^{\pi} P(z) P(z^{-1}) S_x(z) \frac{dz}{z} \quad (2.41)$$

Como os coeficientes do filtro dependem apenas do sinal de entrada, a potência do erro pode ser expressa apenas como uma função de $S_x(e^{j\omega})$. Pode-se derivar essa expressão, considerando que o filtro de erro de predição é de fase mínima [23]. Da equação 2.40, chega-se ao resultado:

$$J = \exp \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln [S_x(e^{j\omega})] d\omega \right\} \quad (2.42)$$

conhecido como fórmula de Kolmogoroff-Szego.

O filtro de erro de predição é também chamado de filtro inverso, supondo-se que seja o inverso do filtro que modela o sinal $x(n)$, sendo essa modelagem um processo autoregressivo [24]. As aplicações dessa versão são muito utilizadas em síntese de voz [9]. A Figura 2.7 mostra a visão geral do processo de síntese e análise de um sinal.

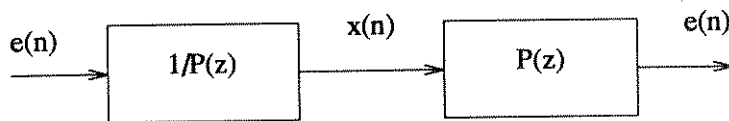


Fig. 2.7 - Modelagem e Predição Linear

2.2.3.2. Predição Linear com Filtros FIR

Na prática, o preditor linear é descrito com um número finito de coeficientes. Assim, a predição linear é uma filtragem onde a seqüência de entrada $\{x(n)\}$ é estimada como combinação linear das amostras anteriores:

$$\hat{x}(n) = \sum_{i=1}^M a_i x(n-i) \quad (2.43)$$

com $\hat{x}(n)$ sendo o sinal predito de $x(n)$ e a_i , os coeficientes do filtro preditor FIR. Novamente, o erro de predição é tal que:

$$e(n) = x(n) - \sum_{i=1}^M a_i x(n-i) \quad (2.44)$$

O filtro de erro de predição é descrito, como na equação 2.39, ressaltando que $A(z)$, o filtro preditor, é da forma:

$$A(z) = \sum_{i=1}^M a_i z^{-i} \quad (2.45)$$

Esse esquema é mostrado na Figura 2.8.

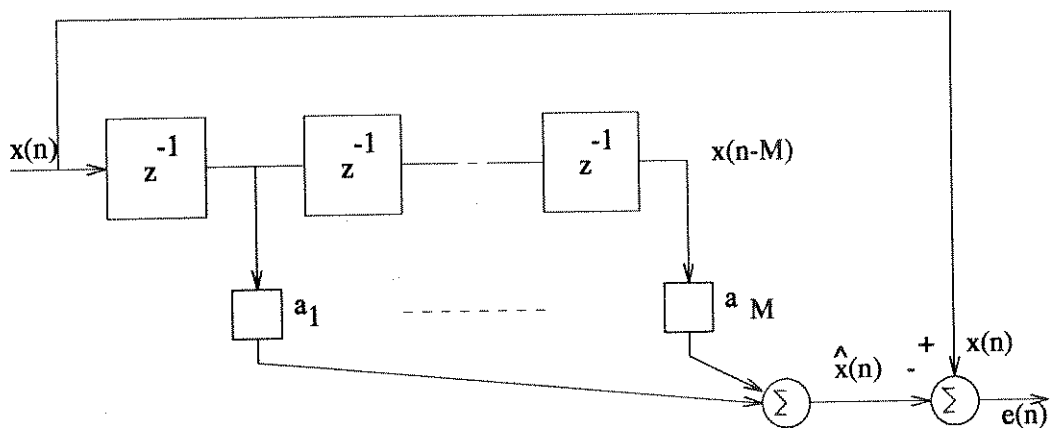


Fig. 2.8 - Filtro de Erro de Predição FIR

A determinação dos coeficientes ótimos do filtro de erro de predição FIR, no sentido do erro médio quadrático, reduz-se a resolver as equações:

$$\frac{\partial}{\partial a_i} [E\{e^2(n)\}] = 0 \quad i = 1, \dots, M \quad (2.46)$$

Sejam $\underline{X}(n-1)$, vetor de M amostras passadas do sinal de entrada, e \underline{A} , vetor dos coeficientes do filtro preditor, tais que:

$$\underline{X}(n-1) = [x(n-1), x(n-2), \dots, x(n-M)]^T \quad (2.47)$$

e

$$\underline{A} = [a_1 \dots a_M]^T \quad (2.48)$$

então,

$$e(n) = x(n) - \underline{A}^T \underline{X}(n-1) \quad (2.49)$$

Tomando-se \mathbf{R}_{xx} como sendo a matriz de autocorrelação do vetor de entrada (MxM), escrita da forma:

$$\mathbf{R}_{xx} = E\{\underline{X}(n).\underline{X}^T(n)\} = E\{\underline{X}(n-1).\underline{X}^T(n-1)\} \quad (2.50)$$

e \mathbf{R}_{xx} , o vetor correlação cruzada entre o novo dado que chega e o vetor das amostras anteriores, ou seja:

$$\underline{\mathbf{R}}_{xx} = E\{x(n).\underline{X}(n-1)\} = [p_x(-1) \dots p_x(-M)]^T \quad (2.51)$$

chega-se a solução ótima do problema resolvendo-se o conjunto de equações normais, chamadas também, nesse caso, de equações de Yule-Walker [1]:

$$\mathbf{R}_{xx} \underline{\mathbf{A}}_{ot} = \underline{\mathbf{R}}_{xx} \quad (2.52)$$

onde $\underline{\mathbf{A}}_{ot}$ é o vetor ótimo, tal que:

$$\underline{\mathbf{A}}_{ot} = \mathbf{R}_{xx}^{-1} \underline{\mathbf{R}}_{xx} \quad (2.53)$$

Os métodos para a solução do problema são os mesmos apresentados para o filtro de Wiener, onde o método de Levinson tem algumas modificações devido às particularidades da predição linear (método de Durbin).

2.2.3.3. Filtros de Predição Progressiva e Regressiva

O processo de predição linear em estruturas FIR, apresentado no item anterior, é também chamado de predição linear progressiva, pois representa a predição da amostra seguinte $x(n)$ dado a observação das M amostras anteriores. Assim, o erro de predição progressiva é dado por:

$$e_a(n) = x(n) - \sum_{i=1}^M a_i x(n-i) \quad (2.54)$$

ou, ainda

$$e_a(n) = x(n) - \underline{\mathbf{A}}_a^T \underline{\mathbf{X}}(n-1) \quad (2.55)$$

na forma vetorial, onde o vetor \underline{A} tem dimensão M .

A potência do erro de predição progressiva é dado a partir das equações 2.34 e 2.52 da forma:

$$E_a = r_{xx}(0) - \underline{A}_{ot}^T \underline{R}_{xx}^T \quad (2.56)$$

onde $r_{xx}(0)$ é a variância do sinal de entrada.

A combinação da equação normal (2.52), com a equação 2.56, fornece a equação normal aumentada

$$\underline{R}_{xx(M+1)} \cdot \begin{bmatrix} 1 \\ \underline{A} \end{bmatrix} = \begin{bmatrix} E_a \\ \underline{0} \end{bmatrix} \quad (2.57)$$

onde $\underline{R}_{xx(M+1)}$ pode ser expressa, usando as propriedades da matriz de autocorrelação, como segue:

$$\underline{R}_{xx(M+1)} = \begin{bmatrix} r_{xx}(0) & \underline{R}_{xx}^T \\ \underline{R}_{xx} & \underline{R}_{xx} \end{bmatrix} \quad (2.58)$$

que é uma expressão de grande utilidade no desenvolvimento de técnicas iterativas para obtenção do filtro preditor ótimo.

O erro de predição progressiva é, também, chamado de processo de inovação pelo fato de que, a cada instante, uma nova informação é adquirida. No caso dos filtros FIR, no qual se tem um número finito de dados armazenados, o dado mais antigo, $x(n-M)$, é descartado quando uma nova amostra é adquirida. No entanto, pode-se determinar a amostra mais antiga por um processo análogo chamado predição linear regressiva.

O erro de predição linear regressiva é definido por:

$$e_b(n) = x(n-M) - \sum_{i=1}^M b_i x(n-M+i) \quad (2.59)$$

ou, na forma vetorial:

$$e_b(n) = x(n-M) - \underline{B}^T \underline{X}(n) \quad (2.60)$$

onde \underline{B} é o vetor dos coeficientes regressivos de dimensão M .

O conjunto dos filtros de predição progressiva e regressiva é mostrado na Figura 2.9.

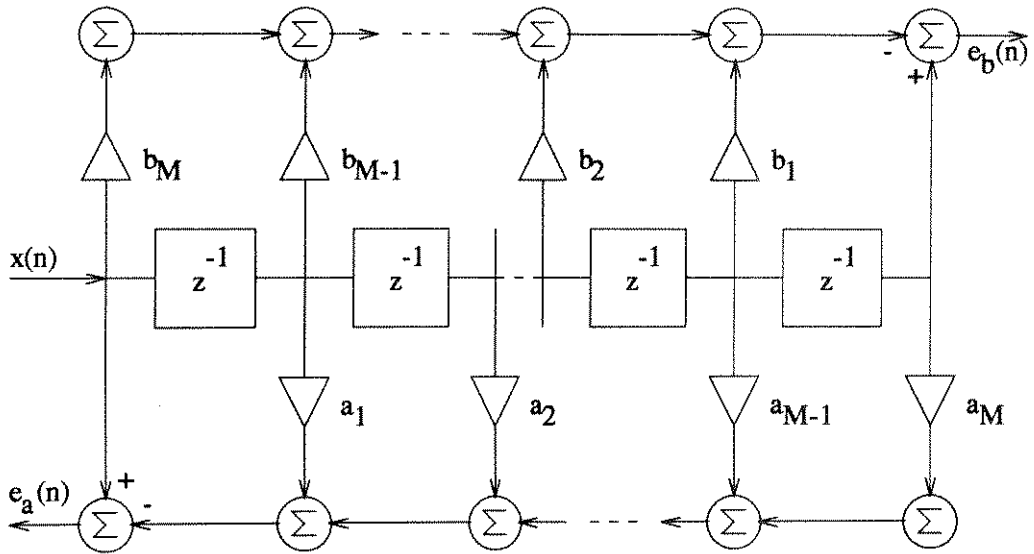


Fig.2.9 - Filtro de Erro de Predição Progressiva e Regressiva

A minimização do critério $E_b = E\{ e_b^2(n) \}$ leva à obtenção do vetor \underline{B} ótimo, a partir da equação normal:

$$\mathbf{R}_{xx} \underline{B}_{ot} = \mathbf{R}_{xx}^b \quad (2.61)$$

onde \mathbf{R}_{xx}^b é o vetor de correlação entre $\underline{X}(n)$ e a amostra $x(n-M)$, ou seja:

$$\mathbf{R}_{xx}^b = E\{x(n-M) \cdot \underline{X}(n)\} = [p_x(M) \dots p_x(1)]^T \quad (2.62)$$

Novamente, pode-se chegar à equação normal aumentada

$$\mathbf{R}_{xx(M+1)} \begin{bmatrix} -\underline{B} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ E_b \end{bmatrix} \quad (2.63)$$

Agora, multiplicando-se a equação 2.63 pela matriz de co-identidade $\mathbf{J}_{(M+1)}$,

onde:

$$\mathbf{J} = \begin{bmatrix} 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \end{bmatrix} \quad (2.64)$$

obtém-se

$$\mathbf{J}_{(M+1)} \cdot \mathbf{R}_{xx(M+1)} \begin{bmatrix} -\underline{\mathbf{B}} \\ 1 \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{E}}_b \\ \underline{\mathbf{0}} \end{bmatrix} \quad (2.65)$$

Levando-se em conta que o sinal é estacionário:

$$\mathbf{R}_{xx(M+1)} \begin{bmatrix} 1 \\ -\mathbf{J}_{(M)} \underline{\mathbf{B}} \end{bmatrix} = \begin{bmatrix} \underline{\mathbf{E}}_b \\ \underline{\mathbf{0}} \end{bmatrix} \quad (2.66)$$

e comparando-se a equação 2.57 com a equação anterior, pode-se constatar que:

$$\underline{\mathbf{A}} = \mathbf{J}_{(M)} \underline{\mathbf{B}} \quad (2.67)$$

e, sobre o erro total

$$\mathbf{E}_a = \mathbf{E}_b = \mathbf{E} \quad (2.68)$$

Assim, para um processo estacionário, a potência do erro da predição progressiva é a mesma que a da regressiva e seus coeficientes estão em ordem inversa. Além disso, o filtro de erro de predição regressiva é de fase máxima [1].

2.2.4. Comentários

Para se determinar o filtro de Wiener é fundamental a hipótese de estacionariedade dos sinais. Os algoritmos de Levinson e Levinson-Durbin (para predição linear) são técnicas iterativas eficientes para a solução direta das equações normais, sendo desenvolvidas a partir das equações normais aumentadas, fazendo o uso do fato que a matriz de autocorrelação é Toeplitz.

O procedimento recursivo do algoritmo de Levinson-Durbin, para predição linear, desenvolvido por Burg, usando simultaneamente as equações do filtro de erro de predição progressiva e regressiva, está presente em [1], bem como sua generalização, considerando sinais não estacionários, pelo algoritmo de Cholesky.

Uma combinação da operação dos filtros de erro de predição progressiva e regressiva, feita numa mesma estrutura, também pode ser usada iterativamente na solução dos problemas de filtragem ótima. São os chamados preditores em treliça [1].

Uma outra técnica de estimação ótima é o filtro de Kalman. Essa técnica é baseada na estimação de estado do sistema [24], sendo um estimador recursivo, não importando se os sinais são ou não estacionários. O filtro de Kalman e sua realização estão descritos em [1].

É importante ressaltar que a filtragem ótima foi apresentada nesta seção a partir de uma visão estocástica dos sinais. Uma outra visão do problema, embora restrita com relação ao sentido de "ótimo", é a do estimador dos mínimos quadrados (versão determinística). O estimador dos mínimos quadrados supera o problema da não estacionariedade, imperativo no filtro de Wiener, usando uma janela ponderada por um fator de esquecimento, sendo uma base para os algoritmos adaptativos. Na seção seguinte abordaremos esse método de forma sucinta, uma vez que a derivação dos resultados é semelhante à da filtragem de Wiener.

2.3 FILTRAGEM POR MÍNIMOS QUADRADOS

O método de estimação dos mínimos quadrados tem como premissa resolver os problemas da filtragem linear sem necessitar das estatísticas dos sinais de entrada e de referência dos filtros. Basicamente, a partir das observações de um bloco de dados disponíveis do sinal de entrada e do sinal desejado, pretende-se minimizar uma função do erro (função custo) expressa em termos do somatório sobre os dados disponíveis, da forma:

$$J(\underline{H}) = \sum_{p1}^{p2} e^2(p) = \sum_{p1}^{p2} \left[y(p) - \hat{y}(p) \right]^2 \quad (2.69)$$

Assim, deseja-se obter os coeficientes do filtro que minimizem a soma dos quadrados do erro de estimação, o que caracteriza a filtragem ótima no sentido dos mínimos quadrados (Least Square - LS).

O método pode ser apresentado como uma alternativa à filtragem de Wiener. Como visto, os filtros de Wiener são derivados das médias estatísticas resultando em um filtro ótimo, no sentido probabilístico, para todas as realizações do processo, suposto estacionário. Em contrapartida, o método dos mínimos quadrados envolve o uso de médias temporais, ao invés de médias de "ensemble", resultando na dependência do filtro ao número de amostras usadas. Ou seja, dife-

rentes filtros são obtidos para diferentes blocos de dados, a não ser que o processo $\{x(n)\}$ seja ergódico e se disponha de infinitos dados ($n \rightarrow \infty$). Nesse caso, a solução dos mínimos quadrados tende à de Wiener.

Para um dado filtro Transversal FIR, a estimativa $\hat{y}(p)$ pode ser escrita na forma da equação 2.16:

$$\hat{y}(p) = \sum_{i=0}^{M-1} h_i x(p-i) \quad (2.70)$$

A função custo, então, é dada em termos dos coeficientes do filtro por:

$$J(\underline{H}) = \sum_{p1}^{p2} \left[y(p) - \sum_{i=0}^{M-1} h_i x(p-i) \right]^2 \quad (2.71)$$

Em particular, quatro diferentes formas de estimação são possíveis, dependendo das hipóteses feitas sobre o janelamento dos dados disponíveis. São elas [1]:

- Método da Covariância- Nenhuma hipótese é feita sobre os dados fora do intervalo $[1, N]$, com os limites de interesse para a matriz de dados sendo $p1=M$ e $p2=N$;
- Método da Autocorrelação- Dados fora do intervalo são nulos. Tem-se $p1=1$ e $p2=N+M-1$;
- Método do Pré-janelamento- Dados anteriores ao intervalo são nulos, com $p1=1$ e $p2=N$;
- Método do Pós-janelamento- Dados posteriores ao intervalo são nulos, com $p1=M$ e $p2=N+M-1$.

A partir da minimização da função custo dada pela equação 2.71, chega-se às equações normais, observando-se um determinado conjunto de dados. A anulação do gradiente em $J(\underline{H})$ implica em

$$\hat{\mathbf{R}}_{xx} \cdot \hat{\underline{H}} = \hat{\mathbf{R}}_{xy} \quad (2.72)$$

onde

$$\hat{\mathbf{R}}_{xx} = \sum_{p1}^{p2} \underline{X}(p) \underline{X}^T(p) \quad (2.73)$$

é a matriz de autocorrelação ($M \times M$) estimada para um bloco de dados. O vetor de correlação cruzada é do tipo:

$$\hat{\underline{R}}_{xy} = \sum_{p=1}^{P^2} \underline{X}(p) y(p) \quad (2.74)$$

O vetor $\hat{\underline{H}}$ é composto pelos valores ótimos dos coeficientes de filtro para um determinado bloco de dados, ou seja:

$$\hat{\underline{H}} = [\hat{h}_0, \hat{h}_1 \dots \hat{h}_{n-1}]^T \quad (2.75)$$

sendo determinado, então, a partir da equação normal por:

$$\hat{\underline{H}} = \hat{\underline{R}}_{xx}^{-1} \cdot \hat{\underline{R}}_{xy} \quad (2.76)$$

supondo-se que a matriz $\hat{\underline{R}}_{xx}$ estimada seja não singular. Essa hipótese, porém, dependerá do conjunto de dados disponíveis. Uma forma de garantir a não singularidade é o uso da pseudo-inversa de Moore-Pensore [1], tal que:

$$\underline{R}_{xx} = \sum_{p=M}^N \underline{X}(p) \underline{X}^T(p) + \delta \mathbf{I} \quad (2.77)$$

O filtro ótimo de predição linear também pode ser obtido pelo método de mínimos quadrados, de forma semelhante a (2.76), utilizando a estimação da correlação cruzada entre o sinal de entrada $x(n)$ e suas amostras passadas [25].

2.3.1. Propriedade dos Estimadores dos Mínimos Quadrados

Para um processo estacionário ergódico, as médias estatísticas correspondem às médias temporais. Assim,

$$J = E[e^2(p)] = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{p=0}^n e^2(p) \quad (2.78)$$

Vê-se a relação com o estimador dos mínimos quadrados a menos de um fator $1/n$, o que não afeta a solução das equações normais. Assim, o estimador dos mínimos quadrados converge assintoticamente para a solução H_{ot} de Wiener. Três resultados básicos são aplicáveis no caso de se ter seqüências de grande comprimento [3], $M < n < \infty$:

- (i) A solução $\hat{\underline{H}}$ é uma estimativa não polarizada da solução de Wiener;

- (ii) A covariância do erro entre a estimativa e \underline{H}_{ot} decresce com o aumento da quantidade de dados disponíveis;
- (iii) O excesso de erro quadrático, quantidade de erro acima de J_{\min} , aumenta linearmente com o número de coeficientes e decresce com o aumento da quantidade de dados disponíveis.

A análise geral das propriedades do estimador dos mínimos quadrados está bem apresentada por Goodwing e Payne [26].

2.4. ALGORITMOS ADAPTATIVOS PELA TÉCNICA DE BUSCA

Em diversos casos práticos, torna-se difícil a solução analítica de um problema de estimação linear tal como visto nas seções precedentes. Além de não se dispor, na prática, das estatísticas dos sinais, soma-se problemas de não estacionariedade e a necessidade da estimação em tempo real. Isso faz com que a filtragem adaptativa seja não só interessante como por vezes necessária. Os algoritmos adaptativos são implementados de maneira que os coeficientes do filtro, variando de acordo com um determinado critério, tendam para a solução do filtro ótimo a medida que os dados vão sendo recebidos. Nos casos não-estacionários, esses algoritmos devem também possuir uma capacidade de rastreamento.

Desse modo, uma primeira alternativa para a determinação de um algoritmo consiste em partir das técnicas clássicas de otimização (busca do mínimo de uma função custo) e adequá-las ao tratamento "on line". Essas técnicas, ditas de busca iterativa, conduzem a soluções simples e implementáveis na prática quando se emprega a idéia da aproximação estocástica. Os princípios gerais e os principais algoritmos serão discutidos a seguir.

Busca Iterativa

As técnicas de busca iterativa são utilizadas em problemas de otimização para encontrar os parâmetros que minimizam uma determinada função objetivo [22]. Assim, a busca é feita sobre a superfície de uma curva de erro médio quadrático, $J(\underline{H})$, conhecida, que é função não linear das variáveis h_i . As técnicas se baseiam no desenvolvimento em série de Taylor da função custo $J(\underline{H})$, seguindo a restrição que:

$$\hat{\underline{H}}_i \rightarrow \hat{\underline{H}}_{i+1} \quad \text{tal que} \quad J|_{\hat{\underline{H}}_{i+1}} < J|_{\hat{\underline{H}}_i} \quad (2.79)$$

Essas técnicas consideram blocos de dados sobre os quais, a partir de uma condição inicial \underline{H}_0 , são feitas iterações na busca da convergência dos coeficientes, na forma:

$$\hat{\underline{H}}_{i+1} = \hat{\underline{H}}_i + \left[\gamma f[J(\underline{H})] \right] \Big|_{\hat{\underline{H}}_i} \quad (2.80)$$

onde a função $f(\cdot)$ indica a direção de busca do ponto de mínimo, baseado em informações de $J(\underline{H})$, e γ é uma constante positiva.

O método da máxima descida, "steepest descent", é baseado na expansão de $J(\underline{H})$ limitada à primeira ordem, ou seja:

$$J(\underline{H})|_{\hat{\underline{H}}_{i+1}} = J(\underline{H})|_{\hat{\underline{H}}_i} + \frac{\partial J(\underline{H})}{\partial \underline{H}^T} \Big|_{\hat{\underline{H}}_i} \Delta \hat{\underline{H}}_{i+1} \quad (2.81)$$

onde,

$$\Delta \hat{\underline{H}}_{i+1} = \hat{\underline{H}}_{i+1} - \hat{\underline{H}}_i \quad (2.82)$$

Vê-se que a variação da função custo na busca do mínimo, quando se introduz os novos coeficientes, é igual ao produto interno entre o gradiente da própria função custo e o vetor diferença entre os coeficientes. Tomando-se esse vetor diferença colinear e em sentido oposto ao gradiente, observa-se que a função custo irá diminuir a cada iteração. O processo iterativo para obtenção dos novos valores do vetor de coeficientes é descrito a seguir:

$$\hat{\underline{H}}_{i+1} = \underline{H}_i - \gamma \left[\frac{\partial J}{\partial \underline{H}^T} \right] \Big|_{\hat{\underline{H}}_i} \quad (2.83)$$

onde γ é uma constante que controla a estabilidade e a rapidez de convergência. Esse método também é chamado de método do gradiente.

O método de Newton-Raphson limita a expansão à componente de segunda ordem. Assim,

$$J(\underline{H})|_{\hat{\underline{H}}_{i+1}} = J(\underline{H})|_{\hat{\underline{H}}_i} + \frac{\partial J(\underline{H})}{\partial \underline{H}^T} \bigg|_{\hat{\underline{H}}_i} \Delta \hat{\underline{H}}_{i+1} + \frac{1}{2} \Delta \hat{\underline{H}}_{i+1}^T \cdot \frac{\partial^2 J(\underline{H})}{\partial \underline{H} \cdot \partial \underline{H}^T} \bigg|_{\hat{\underline{H}}_i} \Delta \hat{\underline{H}}_{i+1} \quad (2.84)$$

onde a busca do mínimo tem como fator principal a derivada segunda da função custo, chamada Hessiana, definida como:

$$\mathbf{H}(\underline{H}) = \frac{\partial^2 J(\underline{H})}{\partial \underline{H} \partial \underline{H}^T} \quad (2.85)$$

Dessa forma, o processo iterativo é descrito por:

$$\hat{\underline{H}}_{i+1} = \hat{\underline{H}}_i - \gamma \left[\mathbf{H}^{-1}(\underline{H}) \cdot \frac{\partial J(\underline{H})}{\partial \underline{H}} \right]_{\hat{\underline{H}}_i} \quad (2.86)$$

Esse método de estimação não linear é muito eficiente, porém sua região de convergência, dependendo da superfície, pode ser muito pequena, necessitando de precaução na escolha do fator γ ($0 < \gamma < 1$) para não extrapolar a região de validade.

Uma aproximação ao método de Newton, dita quase-Newton, é feita de maneira que, expandindo-se a Hessiana, despreza-se o termo de segunda ordem, passando-se a ter uma matriz aproximada \mathbf{B} ao invés da Hessiana.

De uma forma geral, o processo iterativo é escrito, para os três métodos, na forma:

$$\hat{\underline{H}}_{i+1} = \hat{\underline{H}}_i - \gamma \left[\mathbf{B}^{-1} \cdot \underline{S} \right]_{\hat{\underline{H}}_i} \quad (2.87)$$

sendo γ uma constante, \underline{S} o vetor gradiente e \mathbf{B} a matriz que define a direção de busca, de acordo com o método escolhido, ou seja:

- .Matriz identidade \mathbf{I} , para o Gradiente;
- .Hessiana \mathbf{H} , para o Newton Raphson e
- .Matriz de correlação \mathbf{B} , para o Quase Newton.

Esses algoritmos são determinísticos pois percorrem a curva de erro baseados em direções determinadas.

Aproximação Estocástica

É uma classe de algoritmos que busca a solução para certos tipos de

problemas de estimação sequencial. A partir de uma seqüência de dados $\{x(n)\}$, e uma função conhecida $Q(x(n), \underline{H})$, deseja-se a solução para equações do tipo:

$$g(\underline{H}) = E\left\{Q(x(n), \underline{H})\right\} = 0 \quad (2.88)$$

onde \underline{H} é um vetor de parâmetros desconhecidos. Como as estatísticas da seqüência $\{x(n)\}$ não são conhecidas, a solução do sistema é desenvolvida pelo algoritmo proposto por Robbin-Monro [5], que dispõe da atualização desse vetor \underline{H} na forma:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) + \gamma(n+1) Q(x(n+1), \hat{\underline{H}}(n)) \quad (2.89)$$

onde $\{\gamma(n)\}$ é uma seqüência escalar, positiva e tendendo a zero, escolhida sob algum critério.

Propriedades de convergência do algoritmo são mostradas para certas condições de $\{\gamma(n)\}$, $\{x(n)\}$ e $Q(\cdot)$ [27].

Assim, para a minimização da função custo:

$$J = E\{G(x(n), \underline{H})\} \quad (2.90)$$

na forma:

$$-\frac{\partial J(\underline{H})}{\partial \underline{H}} = E\left\{-\frac{\partial}{\partial \underline{H}} G(x(n), \underline{H})\right\} = E\left\{Q^T(x(n), \underline{H})\right\} = 0 \quad (2.91)$$

a recursão é dada por:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) + \gamma(n+1) \frac{\partial G}{\partial \underline{H}}(x(n+1), \hat{\underline{H}}(n)) \quad (2.92)$$

sendo, na média, o ajuste feito via o gradiente da função custo.

Os algoritmos de busca, vistos anteriormente, são determinísticos (isto é, percorrem um caminho pré-determinado na superfície de erro em direção ao mínimo) pois supõem conhecidos o vetor gradiente e, no caso de Newton Raphson, a matriz Hessiana. Isto significa que as estatísticas de 2ª ordem dos sinais envolvidos são conhecidas, o que torna essas técnicas inviáveis na maioria dos casos práticos.

A partir da técnica de aproximação estocástica chega-se a algoritmos que substituem o conhecimento exato das estatísticas por estimativas simples. O percurso de descida da curva torna-se aleatório porém factível de ser efetuado a

tempo real.

Na continuidade, apresenta-se as principais técnicas de busca e seus duais estocásticos para os casos em que se trabalha apenas com o gradiente e com o gradiente em conjunto com a Hessiana.

2.4.1. O Algoritmo do Gradiente

O algoritmo do gradiente busca atingir de forma iterativa o ponto de mínimo da curva de erro médio quadrático, definida por:

$$J(\underline{H}) = \frac{1}{2} E \{e^2(n)\} = \frac{1}{2} \left\{ r_{yy}(0) - 2\underline{R}_{xy}^T \underline{H} + \underline{H}^T \underline{R}_{xx} \underline{H} \right\} \quad (2.93)$$

usando a expressão do gradiente

$$\underline{\nabla}(J(\underline{H})) = \underline{R}_{xx} \underline{H} - \underline{R}_{xy} \quad (2.94)$$

ou ainda:

$$\underline{\nabla}(J(\underline{H})) = \underline{R}_{xx} (\underline{H} - \underline{H}_{ot}) \quad (2.95)$$

onde o vetor \underline{H}_{ot} corresponde aos valores de coeficientes ótimos que anulam o gradiente.

A técnica começa com um vetor arbitrário $\underline{H}(0)$ e vai se atualizando a cada novo instante de tempo, conforme já explicitado em (2.94). Fazendo-se as iterações no tempo, tem-se após a aquisição de novos dados no instante $n+1$:

$$\underline{H}(n+1) = \underline{H}(n) + \underline{\Delta} \underline{H}(n) \quad (2.96)$$

Da técnica iterativa, vê-se que a variação dos coeficientes é sempre no sentido contrário ao vetor gradiente do erro quadrático. O incremento de $\underline{H}(n)$ para $\underline{H}(n+1)$ é então dado por:

$$\underline{\Delta} \underline{H}(n) = \mu [-\underline{\nabla}(J(n))] \quad (2.97)$$

sendo o escalar μ , o fator de convergência que controla a estabilidade e a taxa de adaptação. A recursão

$$\underline{H}(n+1) = \underline{H}(n) - \mu(\underline{R}_{xx} \underline{H}(n) - \underline{R}_{xy}) \quad (2.98)$$

é o algoritmo do gradiente determinístico, pois a cada iteração de $\underline{H}(n)$, o gradiente caminha por lugares determinados pelas estatísticas. Essa expressão corresponde à expressão (2.87) onde as iterações são efetuadas no tempo e a matriz \underline{B} é a identidade.

O método convergirá para o ótimo na condição que o passo de adaptação μ esteja na intervalo:

$$0 < \mu < \frac{2}{\lambda_{\max}} \quad (2.99)$$

onde λ_{\max} é o maior autovalor da matriz de autocorrelação, mostrando a influência dessa matriz na estabilidade do algoritmo. Haykin [1] mostra que o tempo de convergência é diretamente proporcional ao espalhamento dos autovalores da matriz e ao passo de adaptação no algoritmo.

2.4.1.1. O Gradiente Estocástico (LMS)

O algoritmo do gradiente estocástico é o mais utilizado em processamento adaptativo devido a sua simplicidade, robustez e estabilidade numérica. Tem como princípio o fato de que, na maioria das vezes, não se conhece as estatísticas do sinal de entrada, e, em consequência, o valor exato do gradiente da curva de erro médio quadrático. Trabalha-se, então, com estimativas estocásticas desse vetor gradiente, $\hat{\underline{V}}(J(n))$. Assim, o algoritmo de recursão é dado pela expressão:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) - \mu \hat{\underline{V}}(J(n)) \quad (2.100)$$

a ser comparada com a equação (2.92). Daí tira-se que o gradiente é estimado por:

$$\hat{\underline{V}}(J(n)) = \frac{\partial e^2(n)}{\partial \underline{H}(n)} = 2e(n) \frac{\partial e(n)}{\partial \underline{H}(n)} = -2 e(n) \underline{X}(n), \quad (2.101)$$

Supondo $\gamma(n+1)$ constante e igual a $\mu/2$, chega-se que o algoritmo do gradiente estocástico, mais conhecido como algoritmo LMS, pode, ser escrito como:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) + \mu e(n) \underline{X}(n) \quad (2.102)$$

com

$$e(n) = y(n+1) - \hat{\underline{H}}^T(n) \underline{X}(n) \quad (2.103)$$

Para um determinado vetor $\hat{\underline{H}}(n)$, o gradiente estimado é não polarizado [23], ou seja:

$$E[\hat{\underline{V}}] = \underline{V} \quad (2.104)$$

Por sua vez, ao se comparar a aproximação efetuada com o algoritmo determinístico, conclue-se que ela equivale a supor as seguintes estimativas:

$$\hat{\underline{R}}_{xx}(n) = \underline{X}(n) \underline{X}^T(n) \quad (2.105)$$

e

$$\hat{\underline{R}}_{xy}(n) = y(n) \underline{X}(n) \quad (2.106)$$

que embora bastante simples, por contarem apenas com as amostras atuais, são também estimativas não polarizadas.

A simplicidade do algoritmo LMS mostra que cada coeficiente do filtro é atualizado pela adição de um erro ponderado,

$$h_i(n+1) = h_i(n) + \mu e(n) x(n-i) \quad (2.107)$$

sendo $e(n)$ comum a todos os coeficientes, enquanto a ponderação $\{\mu x(n-i)\}$ é proporcional aos dados armazenados para o i -ésimo coeficiente.

Como se pode ver, o algoritmo do LMS executa, a cada passo, uma busca do ponto de mínimo através de caminhos aleatórios, devido ao fato de se trabalhar com uma versão ruidosa do gradiente. Além disso, o vetor de coeficientes não consegue atingir o ponto ótimo, ficando a oscilar em torno desse, caracterizando a presença de um erro residual.

Verifica-se que o passo μ é um fator de extrema importância, pois seu valor influencia tanto na velocidade de convergência como no erro residual. Por exemplo, aumentando-se o passo, aumenta-se a velocidade de convergência, ao custo porém de um maior erro residual. Esse excesso de erro médio quadrático é definido como a diferença entre o limite de convergência, $J(\omega)$, e o ponto de mínimo da função custo, J_{\min} , ou seja:

$$J_{exc} = J(\omega) - J_{min} \quad (2.108)$$

Como condição de estabilidade, o passo μ está limitado pelo fator:

$$0 < \mu < 2/M\sigma_x^2 \quad (2.109)$$

onde $M\sigma_x^2$ é a potência total de entrada.

Finalmente, no caso particular da predição linear adaptativa, o algoritmo LMS é dado por:

$$e(n) = x(n) - \hat{\underline{A}}^T(n) \underline{X}(n-1) \quad (2.110)$$

e

$$\hat{\underline{A}}(n+1) = \hat{\underline{A}}(n) - \mu \underline{X}(n-1) e(n) \quad (2.111)$$

2.4.2. Algoritmos Quase Newton

Uma forma de expressar esse algoritmo é a partir da equação do gradiente dada pela equação 2.95. A multiplicação dessa equação por um fator (\mathbf{R}_{xx}^{-1}) em ambos os lados, resulta em :

$$\mathbf{R}_{xx}^{-1} \underline{\nabla}(J(\underline{H})) = \underline{H} - \underline{H}_{ot} \quad (2.112)$$

ou ainda:

$$\underline{H}_{ot} = \underline{H} - \mathbf{R}_{xx}^{-1} \underline{\nabla}(J(\underline{H})) \quad (2.113)$$

que é o método das raízes de Newton, sendo adaptativamente aproximado, considerando-se a recursividade no tempo, a partir da equação 2.87, por:

$$\underline{H}(n+1) = \underline{H}(n) - \gamma \mathbf{R}_{xx}^{-1} \underline{\nabla}(J(n)) \quad (2.114)$$

isto é:

$$\underline{H}(n+1) = \underline{H}(n) - \gamma \mathbf{R}_{xx}^{-1} (\mathbf{R}_{xx} \underline{H}(n) - \underline{R}_{xy}) \quad (2.115)$$

Widrow e Stearn [2] mostram que $\underline{H}(n)$ converge para \underline{H}_{ot} somente se o termo $|1-\gamma|$ for menor que a unidade, ou seja, para:

$$0 < \gamma < 2 \quad (2.116)$$

Do mesmo modo que o gradiente determinístico, a evolução dos coeficientes em (2.115) também é determinística e necessita do conhecimento das estatísticas de segunda ordem. Por outro lado, a multiplicação do gradiente pela matriz de autocorrelação inversa provoca uma rotação sobre aquele vetor em direção ao ponto de mínimo, acelerando o processo de convergência.

2.4.2.1. O Algoritmo de Gauss-Newton

Uma das aproximações do algoritmo de Newton na forma estocástica é o Gauss-Newton. Aqui, se utiliza não só as estimativas do gradiente, como também as da matriz de autocorrelação, considerando que essa aproximação seja positiva definida. Como visto, a direção de Newton é mais refinada que a do gradiente. Assim, a recursão é escrita como segue:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) - \gamma \underline{R}_{xx}^{-1}(n) \hat{\underline{V}}(J(n)) \quad (2.117)$$

onde a estimativa $\underline{R}_{xx}(n)$ pode ser obtida recursivamente da aproximação estocástica:

$$E\{\underline{X}(n).\underline{X}^T(n) - \underline{R}(n)\} = 0 \quad (2.118)$$

pela solução iterativa de Robbins-Monro, na forma:

$$\underline{R}_{xx}(n) = \underline{R}_{xx}(n-1) + \alpha(n) [\underline{X}(n).\underline{X}^T(n) - \underline{R}_{xx}(n-1)] \quad (2.119)$$

onde $\alpha(n)$ é uma seqüência positiva de ganho, $0 < \alpha \ll 1$.

Para se obter a inversa da matriz em 2.119, também de forma recursiva, deve-se começar reescrevendo:

$$\underline{R}_{xx}(n) = (1-\alpha) \underline{R}_{xx}(n-1) + \alpha [\underline{X}(n) \underline{X}^T(n)] \quad (2.120)$$

com α constante. Pré-multiplicando a equação 2.120 por $\underline{R}_{xx}^{-1}(n)$ e pós-multiplicando por $\underline{R}_{xx}^{-1}(n-1)$, tem-se:

$$\mathbf{R}_{xx}^{-1}(n-1) = (1-\alpha) \mathbf{R}_{xx}^{-1}(n) + \alpha \mathbf{R}_{xx}^{-1}(n) \underline{\mathbf{X}}(n) \underline{\mathbf{X}}^T(n) \mathbf{R}_{xx}^{-1}(n-1) \quad (2.121)$$

e, ainda, pós-multiplicando pelo vetor $\underline{\mathbf{X}}(n)$, escreve-se:

$$\mathbf{R}_{xx}^{-1}(n-1) \underline{\mathbf{X}}(n) = \mathbf{R}_{xx}^{-1}(n) \underline{\mathbf{X}}(n) (1 - \alpha + \alpha \underline{\mathbf{X}}^T(n) \mathbf{R}_{xx}^{-1}(n-1) \underline{\mathbf{X}}(n)) \quad (2.122)$$

Agora, define-se o vetor auxiliar:

$$\underline{\mathbf{S}}(n) = \mathbf{R}_{xx}^{-1}(n-1) \underline{\mathbf{X}}(n) \quad (2.123)$$

com transposta na forma:

$$\underline{\mathbf{S}}^T(n) = \underline{\mathbf{X}}^T(n) \mathbf{R}_{xx}^{-1}(n-1) \quad (2.124)$$

Rearranjando a equação 2.121, tem-se como resultado:

$$\mathbf{R}_{xx}^{-1}(n) = \frac{1}{(1-\alpha)} \left[\mathbf{R}_{xx}^{-1}(n-1) - \frac{\underline{\mathbf{S}}(n) \underline{\mathbf{S}}^T(n)}{(1-\alpha) + \alpha \underline{\mathbf{X}}^T(n) \underline{\mathbf{S}}(n)} \right] \quad (2.125)$$

As equações 2.123 e 2.125 definem a recursão da matriz inversa $\mathbf{R}_{xx}^{-1}(n)$, com valor inicial $(1/\sigma^2)\mathbf{I}$. Dessa forma, a atualização do vetor de coeficientes pode ser expressa por:

$$\hat{\underline{\mathbf{H}}}(n+1) = \hat{\underline{\mathbf{H}}}(n) + \gamma e(n) \mathbf{R}_{xx}^{-1}(n) \underline{\mathbf{X}}(n) \quad (2.126)$$

Novamente, uma análise de convergência é de fundamental importância. O fator α da recursão da matriz de autocorrelação aproximada deve ser suposto muito pequeno, a fim de se ter

.Convergência na média

$$E\{\hat{\underline{\mathbf{H}}}(n)\} \rightarrow \hat{\underline{\mathbf{H}}}_{ot} \quad p/ \quad n \rightarrow \infty \quad (2.127)$$

.Convergência na média quadrática

$$\mathbf{J}(n) \rightarrow \mathbf{J}(\infty) \quad p/ \quad n \rightarrow \infty \quad (2.128)$$

onde o erro residual, sob certas condições [2], é expresso por:

$$J_{\text{ex}}(n) = \frac{\gamma J_{\text{min}}}{1-\gamma} \cdot M \quad (2.129)$$

2.5. ALGORITMOS RECURSIVOS

O método dos mínimos quadrados é estendido, aqui, para o desenvolvimento de algoritmos recursivos para o projeto de filtros adaptativos. Essa abordagem é tal que, dada a estimativa, no sentido dos mínimos quadrados, do vetor de coeficientes no instante n , pode-se computar a estimativa atualizada desse vetor em $n+1$, com a chegada de novos dados.

Um filtro ótimo pode ser obtido para cada novo conjunto de dados formado quando o próximo dado de entrada é observado. Como uma técnica recursiva para computar o vetor de coeficientes $\underline{H}(n)$, o método dos mínimos quadrados é implementado de forma adaptativa visando, principalmente, aplicação em tempo real. Porém, o problema está em se fazer a inversão da matriz a cada novo conjunto de dados.

É importante inserir, no contexto adaptativo, um fator de esquecimento que possibilite ao filtro acompanhar as possíveis variações estatísticas ao longo do tempo. Assim, a função custo a ser minimizada torna-se uma soma ponderada do erro quadrático. Usando o método do pré-janelamento, com um fator de esquecimento exponencial, essa função custo expressa-se na forma:

$$J(n) = \sum_{p=1}^n W^{(n-p)} e^2(p) \quad (2.130)$$

onde $0 \ll W < 1$ é o fator de esquecimento e n é o comprimento variável dos dados observáveis.

O valor ótimo do vetor de coeficientes, $\underline{H}(n)$, para o qual a função custo $J(n)$ atinge o valor mínimo é obtido pelo conjunto de equações normais:

$$\hat{\mathbf{R}}_{xx}(n) \hat{\underline{H}}(n) = \hat{\mathbf{R}}_{xy}(n) \quad (2.131)$$

onde a matriz de autocorrelação estimada $\hat{\mathbf{R}}_{xx}(n)$, de dimensão $(M \times M)$, e o vetor de correlação cruzada estimado $\hat{\mathbf{R}}_{xy}(n)$, $M \times 1$, são definidos como:

$$\hat{\mathbf{R}}_{xx}(n) = \sum_{p=1}^n W^{(n-p)} \underline{X}(p) \underline{X}^T(p) \quad (2.132)$$

e

$$\hat{\mathbf{R}}_{xy}(n) = \sum_{p=1}^n W^{(n-p)} \underline{\mathbf{X}}(p) y(p) \quad (2.133)$$

respectivamente.

Isolando-se os termos correspondentes a $p=n$ do resto do somatório da equação 2.132, obtém-se:

$$\hat{\mathbf{R}}_{xx}(n) = W \left[\sum_{i=1}^{n-1} W^{(n-1-i)} \underline{\mathbf{X}}(i) \underline{\mathbf{X}}^T(i) \right] + \underline{\mathbf{X}}(n) \underline{\mathbf{X}}^T(n) \quad (2.134)$$

sendo que o termo entre colchetes, $\hat{\mathbf{R}}_{xx}(n-1)$, se refere aos dados anteriores, enquanto o segundo termo diz respeito ao vetor de amostras observado. Logo, a recursão para a atualização dos valores da matriz estimada de autocorrelação é dada por:

$$\hat{\mathbf{R}}_{xx}(n) = W \hat{\mathbf{R}}_{xx}(n-1) + \underline{\mathbf{X}}(n) \underline{\mathbf{X}}^T(n) \quad (2.135)$$

Da mesma forma, no caso do vetor de correlação cruzada:

$$\hat{\mathbf{R}}_{xy}(n) = W \hat{\mathbf{R}}_{xy}(n-1) + \underline{\mathbf{X}}(n) y(n) \quad (2.136)$$

Portanto, as novas estimativas da matriz de autocorrelação e do vetor de correlação cruzada são obtidas de uma forma recursiva a fim de se atualizar o vetor dos coeficientes do filtro, tal que:

$$\hat{\mathbf{H}}(n) = \hat{\mathbf{R}}_{xx}^{-1}(n) \hat{\mathbf{R}}_{xy}(n) \quad (2.137)$$

2.5.1. Algoritmo dos Mínimos Quadrados Recursivos - RLS

O algoritmo dos mínimos quadrados recursivos prevê a estimativa do vetor $\hat{\mathbf{H}}(n)$ a partir da estimativa anterior $\hat{\mathbf{H}}(n-1)$, dada então por:

$$\hat{\mathbf{H}}(n-1) = \hat{\mathbf{R}}_{xx}^{-1}(n-1) \hat{\mathbf{R}}_{xy}(n-1) \quad (2.138)$$

Toma-se, assim, um procedimento recursivo para atualização do vetor dos coeficientes do filtro a partir da substituição das equações (2.138),

(2.135) e (2.136) na equação (2.137), obtendo-se:

$$\hat{\underline{H}}(n+1) = \hat{\underline{H}}(n) + \hat{\underline{R}}_{xx}^{-1}(n+1)\underline{X}(n+1) \left[y(n+1) - \underline{X}^T(n+1) \hat{\underline{H}}(n) \right] \quad (2.139)$$

É de extrema importância dispor de um algoritmo que suprima a necessidade de inversão da matriz de autocorrelação a cada iteração. Uma forma de se fazer isso é através do lema da inversão de matrizes.

Lema: Sejam A e B matrizes positivas definidas (MxM) tal que:

$$A = B^{-1} + C D^{-1} C^T \quad (2.140)$$

onde D é uma matriz positiva definida (NxN) e C é uma matriz (MxN). Então a inversa da matriz A é dada por:

$$A^{-1} = B - B C (D + C^T B C)^{-1} C B \quad (2.141)$$

A partir da equação 2.134, fazendo $A=R(n)$, $B=W R(n-1)$, $C=\underline{X}(n)$ e $D=1$, pode-se escrever:

$$\hat{\underline{R}}_{xx}^{-1}(n) = W^{-1}\hat{\underline{R}}_{xx}^{-1}(n-1) - \frac{W^{-2}\hat{\underline{R}}_{xx}^{-1}(n-1) \underline{X}(n)\underline{X}^T(n) \hat{\underline{R}}_{xx}^{-1}(n-1)}{1 + W^{-1}\underline{X}^T(n) \hat{\underline{R}}_{xx}^{-1}(n-1)\underline{X}(n)} \quad (2.142)$$

onde, considerando o vetor ganho de adaptação, de dimensão (Mx1), dado por:

$$\underline{G}(n+1) = \frac{\hat{\underline{R}}_{xx}^{-1}(n)\underline{X}(n+1)}{W + \underline{X}^T(n+1) \hat{\underline{R}}_{xx}^{-1}(n)\underline{X}(n+1)} \quad (2.143)$$

pode-se escrever a recursão da inversa:

$$\hat{\underline{R}}_{xx}^{-1}(n+1) = \frac{1}{W} \left[\hat{\underline{R}}_{xx}^{-1}(n) - \underline{G}(n+1)\underline{X}^T(n+1) \hat{\underline{R}}_{xx}^{-1}(n) \right] \quad (2.144)$$

que é a equação de Ricatti para o RLS [1].

Reorganizando a equação 2.143 e observando a equação 2.144, chega-se a

$$\underline{G}(n+1) = \hat{\underline{R}}_{xx}^{-1}(n+1)\underline{X}(n+1) \quad (2.145)$$

levando o processo de atualização do vetor de coeficientes do filtro, da equação 2.139, a ser escrito da forma :

$$\hat{\underline{H}}(n+1) = \underline{H}(n) + \underline{G}(n+1)e(n+1) \quad (2.146)$$

onde

$$e(n) = y(n) - \underline{X}^T(n) \underline{H}(n-1) \quad (2.147)$$

é definido como o erro de estimação à priori, lembrando-se que $e(n)$ é o erro de estimação à posteriori, ou seja:

$$e(n) = y(n) - \underline{X}^T(n) \underline{H}(n) \quad (2.148)$$

O procedimento recursivo envolvendo as equações 2.143, 2.147, 2.146 e 2.144, nessa ordem, é o algoritmo RLS, no qual nenhuma operação de inversão de matriz é realizada. A inicialização do processo requer um matriz inversa diferente de matriz nula. Uma matriz inicial é proposta [23] na forma:

$$\hat{\underline{R}}_{xx}^{-1}(0) = \frac{1}{E_0} \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & W & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & W^{(N-1)} \end{bmatrix} \quad (2.149)$$

A aplicação do algoritmo RLS em predição linear leva à substituição, nas equações 2.143, 2.144, do vetor $\underline{X}(n+1)$ pelo vetor $\underline{X}(n)$, além das considerações de predição na equação 2.147.

O fator de esquecimento W determina a quantidade de memória do algoritmo. Quando $W=1$, RLS não ponderado, o algoritmo é de memória infinita. Para $W < 1$, pode-se verificar, a partir das equações 2.120, 2.125, 2.134 e 2.137, que os algoritmos RLS e Gauss-Newton são matematicamente idênticos, desde que se tome $\gamma = \alpha = (1-W)$ [5]. A relação entre α e W é feita a partir do fato que tanto $\underline{R}_{xx}(n)$ quanto $\hat{\underline{R}}_{xx}(n)$ são estimativas da matriz de autocorrelação, sendo $\underline{R}_{xx}(n)$ a estimativa normalizada no tempo, podendo-se escrever:

$$\underline{R}_{xx}(n) = \frac{\hat{\underline{R}}_{xx}(n)}{\sum_{p=1}^n W^{n-p}} = \alpha \hat{\underline{R}}_{xx}(n) \quad (2.150)$$

Assim, o algoritmo RLS pode ser visto como uma derivação da aproximação estocástica (Gauss-Newton), com um fator de esquecimento $W = (1-\alpha) < 1$. Os problemas de convergência devem ser analisados nessa ótica sob os aspectos da convergência na média e na média quadrática [1]. Verifica-se, pois, que o algoritmo converge na média e, para um processo estacionário, o melhor resultado de convergência na média quadrática é obtido para $W=1$, onde o fator de desajuste é nulo. Seus resultados são muito superiores ao LMS.

A técnica do algoritmo do RLS pode, também, ser vista como uma derivação do filtro de Kalman. Essa visão é demonstrada em [1].

Um problema relevante do algoritmo RLS é a sua complexidade numérica, um total de $2M^2+4M$ multiplicações e 1 divisão são requeridas à cada iteração [23]. Para reduzir esse problema, algoritmos rápidos foram propostos na última década [28, 29, 30].

2.5.2. Algoritmo dos Mínimos Quadrados Rápidos - FLS

Como visto, o custo computacional do algoritmo RLS é muito elevado, comparado com o LMS, mesmo em sistemas de ordem baixa. Por esse motivo, versões rápidas foram desenvolvidas levando-se em conta a propriedade de deslocamento do vetor $\underline{X}(n)$, na qual, um novo vetor $\underline{X}(n+1)$ é formado a partir do vetor $\underline{X}(n)$, bastando para isso o deslocamento de seus elementos, com a introdução da nova amostra $x(n+1)$ e a remoção da amostra $x(n-M+1)$.

Os algoritmos rápidos realizam, rigorosamente, o critério dos mínimos quadrados, fazendo uma adaptação interna no ganho de adaptação, também chamado ganho de Kalman. Em 1978, Ljung *et al.* [28], derivaram uma recursão para o cálculo do ganho sem efetuar o procedimento recursivo dado pelas equações 2.143, 2.144 e 2.145.

O algoritmo se baseia no fato que toda a informação contida na matriz de autocorrelação, que intervém no cálculo do ganho de adaptação, pode ser completamente determinada através dos erros de predição e dos respectivos coeficientes dos filtros de predição linear progressiva e regressiva. Assim, é possível calcular $\underline{G}(n+1)$ a partir de $\underline{G}(n)$ sem que seja necessária a manipulação matricial da equação 2.143, usando apenas os parâmetros de predição linear, que são vetores ou escalares. Mostra-se também que tal procedimento só é possível quando os dados de entrada possuem a propriedade de deslocamento, o que é próprio dos filtros transversais FIR [28].

Dessa forma, esses algoritmos apresentam a mesma rapidez de convergência, no sentido do número de iterações necessárias, que a técnica dos mínimos quadrados recursivos RLS, porém são rápidos por terem menos cálculo computacional em cada iteração.

O algoritmo dos mínimos quadrados rápidos FLS, também chamado de algoritmo rápido de Kalman [18], utilizando apenas o erro de predição à priori, é mostrado na Tabela 2.1, com sua inicialização decorrente de 2.149. Em termos de complexidade computacional, o algoritmo requer cerca de 10M multiplicações e 2 divisões por iteração.

Outras versões de algoritmos rápidos, com um pouco mais de eficiência, foram posteriormente desenvolvidas, como o FAEST [29] e o FTF[30], que empregam também o erro de predição a posteriori e reduzem a complexidade para cerca de 8M multiplicações e 3 divisões [23].

Apesar de sua grande eficiência, esses algoritmos apresentam um problema crítico no que se refere à sensibilidade aos efeitos de precisão finita. [31]. Esse problema foi objeto de intensa pesquisa nos últimos anos sendo que se destacam os resultados apresentados nas referências [32] e [33].

A inicialização do algoritmo é obtida fazendo-se:

$$\underline{A}(0) = \underline{B}(0) = \underline{g}(0) = \underline{0} \quad \text{e} \quad E_a = E_o \quad (2.151)$$

Uma observação importante do algoritmo FLS é que sua aplicação a predição linear é direta pela determinação do vetor $\underline{A}(n)$.

2.6. DISCUSSÃO SOBRE O DESEMPENHO DOS ALGORITMOS

O algoritmo do LMS é o mais popular dentre os algoritmos de adaptação, sendo tomado como referência para comparações. É um algoritmo simples, não requerendo medidas da correlação, tendo grau de complexidade igual a 2M multiplicações e 2M adições/subtrações. Como todo algoritmo recursivo, requer controle da estabilidade, no seu caso, feito pelo passo μ , limitado pelos autovalores da matriz de autocorrelação dos dados de entrada. A velocidade de convergência é lenta, dependendo, novamente de μ e do espalhamento dos autovalores da matriz. Devido ao fato de se trabalhar com uma versão ruidosa do gradiente, percorrendo caminhos aleatórios, ao contrário do método de máxima descida que é determinís-

Quantidades disponíveis no instante n:

- $\underline{H}(n)$ - Vetor de coeficientes do filtro adaptativo
 $\underline{A}(n)$ - Vetor de coeficientes de predição progressiva
 $\underline{B}(n)$ - Vetor de coeficientes de predição regressiva
 $\underline{X}(n)$ - Vetor de dados de entrada
 $\underline{g}(n)$ - Vetor ganho de adaptação
 $E_a(n)$ - Erro de predição acumulado

Novos dados no instante (n+1):

- $x(n+1)$ - Dado recebido
 $y(n+1)$ - Sinal de referência

Atualização do vetor ganho de adaptação:

- $e_a(n+1) = x(n+1) - \underline{A}^T(n)\underline{X}(n)$
(2) Coeficientes de predição progressiva
 $\underline{A}(n+1) = \underline{A}(n) + \underline{g}(n)e_a(n+1)$
(3) Erro de predição progressiva a posteriori
 $\epsilon_a(n+1) = \underline{X}(n+1) - \underline{A}^T(n+1)\underline{X}(n)$
(4) Erro de predição progressiva acumulado
 $E_a(n+1) = W E_a(n) + e_a(n+1)\epsilon_a(n+1)$
(5) Ganho de adaptação intermediário
$$\underline{g}_1(n+1) = \begin{bmatrix} 0 \\ \underline{g}(n) \end{bmatrix} + \frac{\epsilon_a(n+1)}{E_a(n+1)} \begin{bmatrix} 1 \\ -\underline{A}(n+1) \end{bmatrix} = \begin{bmatrix} \underline{m}(n+1) \\ \underline{m}(n+1) \end{bmatrix}$$

(6) Erro de predição regressiva a priori
 $e_b(n+1) = \underline{X}(n+1-N) - \underline{B}^T(n)\underline{X}(n+1)$
(7) Ganho de Adaptação
$$\underline{g}(n+1) = \frac{1}{1 - \underline{m}(n+1) e_b(n+1)} [\underline{m}(n+1) + \underline{m}(n+1)\underline{B}(n)]$$

(8) Coeficientes de predição regressiva
 $\underline{B}(n+1) = \underline{B}(n) + \underline{g}(n+1)e_b(n+1)$
Atualização dos coeficientes do filtro adaptativo
(9) Sinal de erro
 $e(n+1) = y(n+1) - \underline{H}^T(n)\underline{X}(n+1)$
(10) Coeficientes do filtro
 $\underline{H}(n+1) = \underline{H}(n) + \underline{g}(n+1)e(n+1)$

TABELA 2.1: O algoritmo rápido de Kalman

tico, o algoritmo não atinge seu ponto de mínimo, ficando oscilando em torno desse. Esse é o preço pago por usar estimativas para atualizar os coeficientes do filtro. A redução do fator de desajuste (relação entre o excesso de erro médio quadrático, residual, e o J_{\min}) também depende da escolha do μ . Assim, diminuindo o passo, diminui-se o erro residual, J_{exc} , ao custo porém de uma redução na velocidade de convergência. Deve-se levar em conta, na escolha do μ muito pequeno, os problemas de arredondamento, que interferem na adaptação. No contexto não estacionário, além de se buscar o mínimo da função custo, o algoritmo deve ser capaz de rastrear a troca de posição desse mínimo. Para um μ pequeno e uma não estacionariedade branda, o LMS tem bom desempenho. O LMS supõe trabalhar com sinais de espectro plano (branco), piorando seu desempenho quanto mais correlacionado for o sinal.

O algoritmo RLS leva em consideração todas as informações contidas no sinal. Sua grande vantagem sobre o LMS é a extrema rapidez de convergência, sendo insensível ao espalhamento da matriz. Essa rapidez de convergência é fundamental em aplicações onde as estatísticas do sinal variam de forma rápida com o tempo, de modo que o RLS se adapta bem ao rastreamento com variações bruscas. Sua grande desvantagem é a complexidade computacional elevada, sendo proporcional a M^2 . Apresenta alguns problemas quanto ao ruído de arredondamento nas recursões internas. Alguns algoritmos são desenvolvidos para remediar esse problema, por exemplo a fatorização de Cholesky. No sentido de superar o problema de complexidade numérica, algoritmos rápidos foram desenvolvidos, aumentando porém os problemas de arredondamento em aritmética de precisão finita, podendo levar, em alguns casos, à instabilidade.

É notória a proporcionalidade do fator de memória do LMS, μ , com o fator de memória do RLS, $(1-W)$, em situações não estacionárias. A relação entre esses fatores permite a realização de análise e simulação comparativa. Quando se tem grande quantidade de ruído e, em conseqüência, um fator de espalhamento pequeno, o desempenho de ambos os algoritmos é equiparado.

Devido à dificuldade de se obter resultados analíticos rigorosos para as propriedades de convergência de uma extensa seleção de algoritmos adaptativos, a comparação experimental é de grande importância, usando-se simulações computacionais.

Como exemplo de uma comparação do desempenho entre o LMS e o FLS, a predição linear de senóides em meio ao ruído branco gaussiano é colocada. Esse problema é de particular interesse, quando se busca localizar a posição dos polos de um modelo, a partir dos coeficientes do filtro preditor [11]. Deve-se

ressaltar que, no FLS, as operações de filtragem não são necessárias, obtendo-se os coeficientes do filtro preditor direto das operações de atualização do ganho, diminuindo, nesse caso, a complexidade computacional.

Considera-se a predição de duas senóides muito próximas: $f_1=f_e/10$ e $f_2=f_e/6$, onde f_e é a frequência de amostragem. O sinal contendo as duas senóides mais ruído com potência 10^{-6} , é aplicado na entrada do preditor FIR de ordem 4. Os algoritmos estão especificados com fator de esquecimento $W=0,9$, para o FLS, e passo de adaptação $\mu=0.1$, para o LMS, que são razoavelmente equivalentes [23].

A Figura 2.10 mostra a evolução dos coeficientes e dos zeros do preditor para ambos os algoritmos. Vê-se que, após mil iterações, o gradiente ainda é incapaz de discernir as duas frequências, sendo que a posição dos zeros reflete a presença, apenas, de uma frequência intermediária ruidosa. Ao contrário, o FLS localiza as duas frequências em menos de cem iterações. O LMS só alcança o resultado após mais de dez mil iterações devido ao fato do sinal senoidal ser fortemente correlacionado. O algoritmo FLS se mostra como sendo muito interessante na localização rápida de frequências por predição linear.

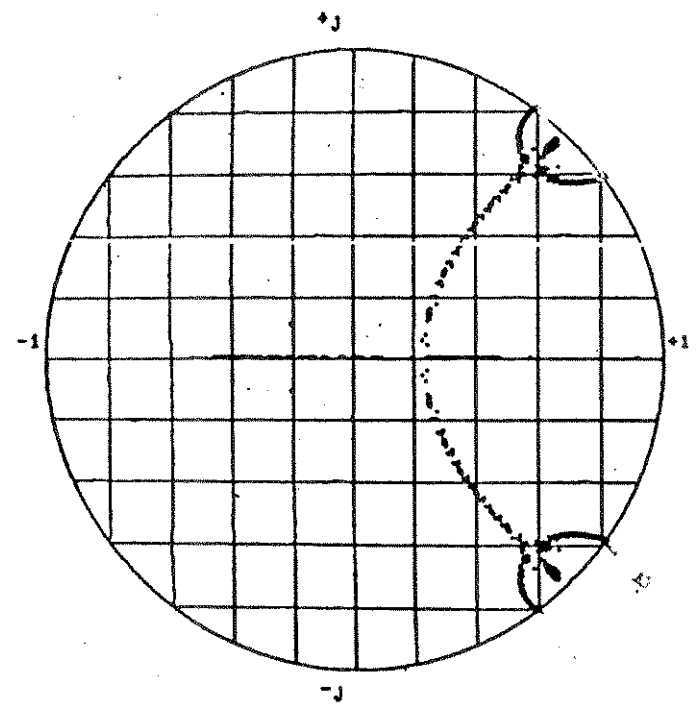
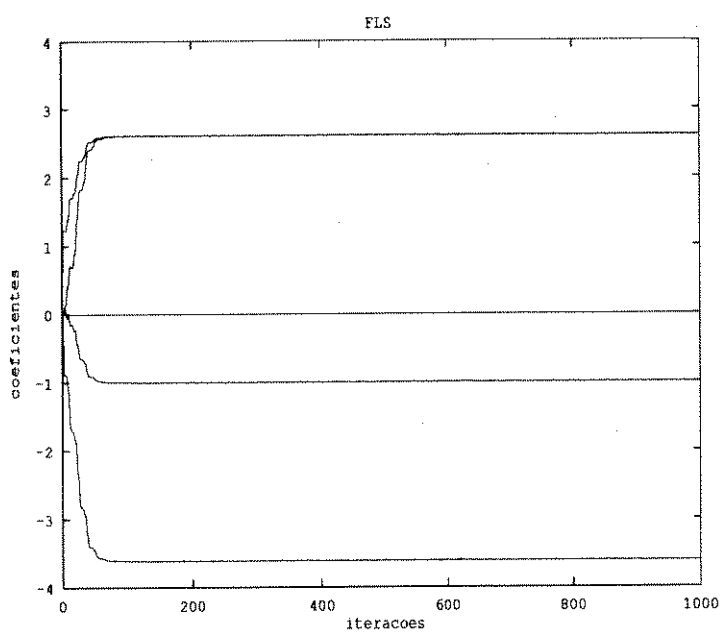
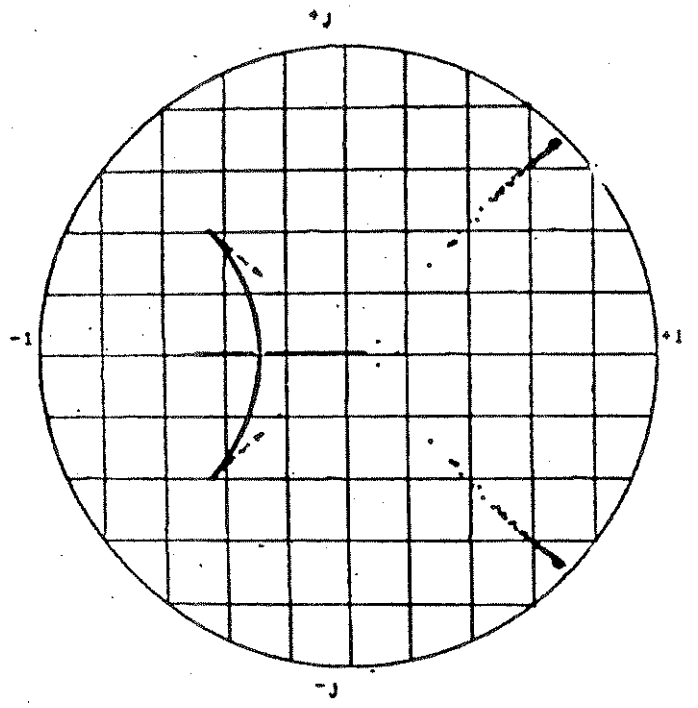
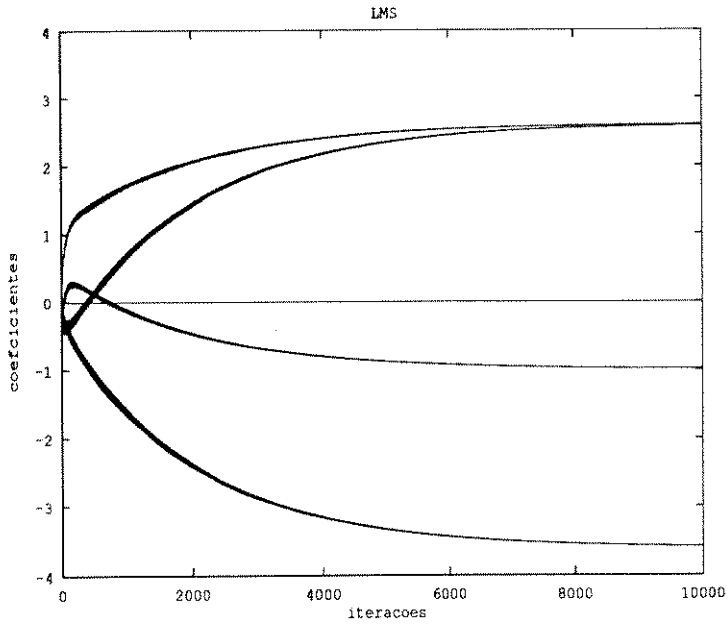


Fig. 2.10 - Predição de duas senóides com estrutura direta

2.7. SINAIS MULTIDIMENSIONAIS EM ESTRUTURAS FIR

Uma situação de grande importância no desenvolvimento da proposta deste trabalho é considerar um sistema com entrada multidimensional. Seja o caso de um sinal vetorial de entrada consistindo de K elementos $x_k(n)$, com $1 < k < K$, e um sinal de referência escalar $y(n)$. Essa situação está ilustrada na Figura 2.11.

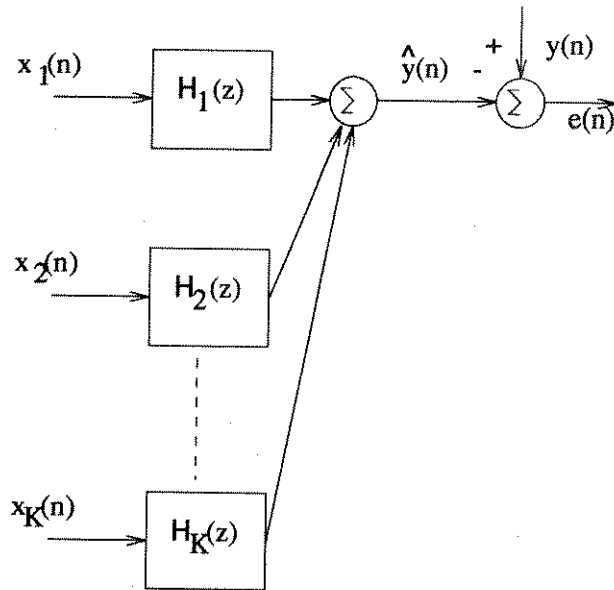


Fig.2.11 - Filtro Adaptativo com Estrutura Multidimensional e Referência Escalar

O filtro programável, cuja saída $y(n)$ é também escalar, consiste de um conjunto de K diferentes filtros com vetores de coeficientes $\underline{H}_k(n)$, sendo seus coeficientes calculados para minimizar uma função custo em tempo real.

A adaptação dos filtros pode ser feita por meio do algoritmo do gradiente, com cada filtro sendo adaptado separadamente, como segue:

$$h_{ik}(n) = h_{ik}(n-1) + \mu x_k(n-1) e(n) \quad \begin{array}{l} k=1, \dots, K \\ i=0, \dots, M-1 \end{array} \quad (2.152)$$

Para a adaptação com o algoritmo dos mínimos quadrados rápidos, FLS, seja o vetor $\underline{X}_E(n)$, tal que:

$$\underline{X}_E(n) = \left[x_1(n), \dots, x_K(n) \right]^T \quad (2.153)$$

Supondo-se que cada vetor de coeficientes dos filtros $H_k(n)$ tenha M elementos, define-se $\underline{X}_T(n)$ como sendo o seguinte vetor de entrada, com KM elementos:

$$\underline{X}_T(n) = \left[\underline{X}_E^T(n), \underline{X}_E^T(n-1), \dots, \underline{X}_E^T(n+1-M) \right]^T \quad (2.154)$$

e considera-se ainda $\underline{H}_T(n)$ como sendo o vetor de todos os coeficientes dos filtros em paralelo, com KM elementos:

$$\underline{H}_T(n) = \left[h_{11}(n), \dots, h_{K1}(n), h_{12}(n), \dots, h_{K2}(n), \dots, h_{1M}(n), \dots, h_{KM}(n) \right]^T \quad (2.155)$$

O sinal de erro pode ser definido, então, como na equação 2.24, ou seja:

$$e(n) = y(n) - \underline{H}_T^T(n) \cdot \underline{X}_T(n) \quad (2.156)$$

A função custo a ser minimizada, para o método do pré-janelamento exponencial é da forma:

$$J(n) = \frac{1}{2} \sum_{p=1}^n W^{n-p} e^2(p) \quad (2.157)$$

o que leva, na condição ótima, à equação normal:

$$\hat{\underline{H}}_T(n) = \hat{\underline{R}}_{(KM)}^{-1}(n) \hat{\underline{R}}_{(KM)}(n) \quad (2.158)$$

com a matriz de correlação e o vetor de correlação cruzada estimados, respectivamente, por:

$$\hat{\underline{R}}_{(KM)}^{-1} = \sum_{p=1}^n W^{n-p} \underline{X}_T(p) \underline{X}_T^T(p) \quad (2.159)$$

$$\hat{\mathbf{R}}_{(KM)}(n) = \sum_{p=1}^n W^{n-p} y(p) \underline{\mathbf{X}}_T^T(p) \quad (2.160)$$

Como visto na seção 2.5, a recursão do vetor de coeficientes pode ser expressa por:

$$\underline{\mathbf{H}}_T(n+1) = \underline{\mathbf{H}}_T(n) + \hat{\mathbf{R}}_{(KM)}^{-1}(n+1) \underline{\mathbf{X}}_T(n+1) e(n+1) \quad (2.161)$$

onde se define o ganho de adaptação

$$\underline{\mathbf{G}}_{(K)}(n) = \hat{\mathbf{R}}_{(KM)}^{-1}(n) \underline{\mathbf{X}}_T(n) \quad (2.162)$$

como um vetor de KM elementos, o qual pode ser adaptado através de um procedimento similar ao do FLS monodimensional, utilizando o erro de predição progressivo e regressivo. Esse procedimento é desenvolvido por Bellanger [23, Cap. 7].

O algoritmo dos mínimos quadrados rápidos, para o caso multidimensional, é mostrado na Tabela 2.2.

O valor inicial da matriz de erro quadrático é tomado como a matriz diagonal

$$\mathbf{E}_a(0) = E_0 \mathbf{I} \quad (2.163)$$

onde E_0 é um escalar positivo. Todos os outros valores iniciais envolvidos podem ser considerados nulos.

A inversão da matriz da equação (5) da Tabela 2.2 é feita a parte, com a ajuda do lema de inversão matricial, obtendo a recursão da matriz de erro quadrático inversa na forma:

$$\mathbf{E}_a^{-1}(n+1) = W^{-1} \left[\mathbf{E}_a^{-1}(n) - \frac{\mathbf{E}_a^{-1}(n) \underline{\mathbf{e}}_a(n+1) \underline{\mathbf{e}}_a^T(n+1) \mathbf{E}_a^{-1}(n)}{W + \underline{\mathbf{e}}_a^T(n+1) \mathbf{E}_a^{-1}(n) \underline{\mathbf{e}}_a(n+1)} \right] \quad (2.164)$$

A complexidade computacional dessa expressão está em torno de $3K^2+2K$ multiplicações e uma divisão. Para as operações relacionadas com o filtro de ordem M, o algoritmo apresentado na Tabela 2.2 despende $7K^2M+KM$ multiplicações para o ganho de adaptação e $2KM$ multiplicações para o processo de filtragem.

Quantidades disponíveis no instante n:

$\underline{H}_T(n)$ - Vetor de coeficientes do filtro adaptativo com KM elementos

M - número de coeficientes por secção

A(n) - Matriz de predição progressiva (KM linhas, K colunas)

B(n) - Matriz de predição regressiva (KM linhas, K colunas)

$\underline{X}_T(n)$ - Vetor de dados com KM elementos

$\underline{G}(n)$ - Vetor ganho de adaptação com KM elementos

$\underline{E}_a(n)$ - Matriz erro de predição quadrático (K,K)

Novos dados no instante (n+1):

x(n+1) - Dado recebido

y(n+1) - Sinal de referência

$\underline{X}_E(n+1)$ - Entrada multidimensional

Atualização do vetor ganho de adaptação:

(1) Vetor erro de predição progressiva à priori (K elementos)

$$\underline{e}_a(n+1) = \underline{X}_E(n+1) - \underline{A}^T(n)\underline{X}_T(n)$$

(2) Matriz de predição progressiva

$$\underline{A}(n+1) = \underline{A}(n) + \underline{G}(n) \underline{e}_a^T(n+1)$$

(3) Vetor erro de predição progressiva à posteriori (K elementos)

$$\underline{\varepsilon}_a(n+1) = \underline{X}_E(n+1) - \underline{A}^T(n+1)\underline{X}_T(n)$$

(4) Matriz erro de predição quadrático

$$\underline{E}_a(n+1) = W \underline{E}_a(n) + \underline{e}_a(n+1)\underline{\varepsilon}_a^T(n+1)$$

(5) Vetor ganho de adaptação intermediário (KM + K elementos)

$$\underline{G}_1(n+1) = \begin{bmatrix} 0 \\ \underline{G}(n) \end{bmatrix} \begin{bmatrix} I \\ -\underline{A}(n+1) \end{bmatrix} + \underline{E}_a^{-1}(n+1) \underline{\varepsilon}_a(n+1) = \begin{bmatrix} \underline{M}(n+1) \\ \underline{m}(n+1) \end{bmatrix}$$

(6) Vetor erro de predição regressiva à priori

$$\underline{e}_b(n+1) = \underline{X}_E(n+1-N) - \underline{B}^T(n) \underline{X}_T(n+1)$$

(7) Vetor ganho de adaptação

$$\underline{G}(n+1) = [1 - \underline{e}_b^T(n+1) \underline{m}(n+1)]^{-1} [\underline{M}(n+1) + \underline{B}(n) \underline{m}(n+1)]$$

(8) Matriz de progressão regressiva

$$\underline{B}(n+1) = \underline{B}(n) + \underline{G}(n+1) \underline{e}_b^T(n+1)$$

Atualização dos coeficientes do filtro adaptativo cascata

(9) Sinal de erro

$$e(n+1) = y(n+1) - (1/K) [\underline{H}_T^T(n) \underline{X}_T(n+1)]$$

(10) Coeficientes

$$\underline{H}_T(n+1) = \underline{H}_T(n) + \underline{G}(n+1)e(n+1)$$

TABELA 2.2: O algoritmo rápido multidimensional

2.8. CONCLUSÕES

A teoria de filtragem adaptativa está hoje bem estabelecida, em particular no que se refere às estruturas FIR, transversais e a treliça, as quais conduzem a um problema de otimização linear e apresentam boas propriedades de convergência e estabilidade.

Procurou-se apresentar neste capítulo os princípios fundamentais de filtragem adaptativa e a derivação dos algoritmos mais importantes. É interessante observar que diferentes pontos de partida, como teoria de Wiener, métodos de mínimos quadrados, técnicas de otimização, etc., conduzem a resultados que na prática são semelhantes. Em última análise, pode-se dizer que os algoritmos LMS e RLS (com sua versão rápida) são os de maior relevância para as estruturas FIR, sendo que esse último pode ser visto também como um caso particular da técnica de Gauss-Newton. Por outro lado, em muitos casos o uso de outro tipo de estrutura (cascata, paralela, IIR) pode ser crucial para a aplicação desejada. Entretanto, a análise do problema torna-se mais complicada e aparecem dificuldades de convergência e estabilidade.

Nossa intenção nos capítulos subsequentes é fornecer uma contribuição ao algoritmo de mínimos quadrados, no sentido de aplicá-lo à estrutura cascata, além de propor novas técnicas de adaptação com sua análise de convergência. Para isto, é fundamental a base teórica disposta neste capítulo.

CAPÍTULO 3

FILTRAGEM ADAPTATIVA COM ESTRUTURAS EM CASCATA FIR

Grande parte das aplicações práticas utilizam filtros digitais adaptativos com estruturas transversais FIR. Isso se justifica pelas conhecidas propriedades de convergência e estabilidade desse tipo de estrutura. Essas propriedades, por sua vez, estão estreitamente ligadas ao fato de que, como visto no capítulo anterior, tais estruturas conduzem a um problema de otimização linear.

Por outro lado, outras motivações podem levar a se decidir por estruturas alternativas. Por exemplo, o uso de filtros IIR é interessante quando se deseja operar com filtros de menor ordem, reduzindo a complexidade computacional.

Em muitas outras aplicações, a posição dos zeros e pólos de um filtro ou preditor adaptativo é o parâmetro de maior importância [11]. Como exemplo, pode-se citar detecção de frequências, monitoramento dos pólos de um modelo AR ou de um sistema a ser identificado, testes de estabilidade, etc. Nesses casos, é altamente conveniente o uso de estruturas em cascata, onde a função de transferência é decomposta em polinômios de 2ª ordem cujas raízes podem então ser facilmente obtidas. Além disso, a estrutura em cascata também diminui consideravelmente a sensibilidade do filtro à quantização de seus coeficientes [8].

A relação entre os coeficientes das formas direta e em cascata do filtro FIR é feita por um mapeamento não linear, levando a se resolver, no caso em cascata, equações normais não lineares [7]. Porém, pode-se mostrar que a solução é única, a menos de permutações entre as seções de segunda ordem. Essa afirmativa vem do fato que essas equações normais não lineares provêm da mesma minimização da norma do erro, que resulta nas equações normais da forma direta, é claro, se essas tiverem solução única. No entanto, no caso em cascata, a curva de erro quadrático não é mais unimodal, o que evidentemente terá conseqüências na convergência, se o filtro for implementado na forma adaptativa.

Propõe-se neste capítulo uma técnica de filtragem adaptativa combinando o uso dos algoritmos dos mínimos quadrados recursivos em conjunto com estruturas na forma em cascata FIR. A abordagem consiste em transladar a original estrutura em cascata para uma forma direta multidimensional FIR, de forma a

atualizar os coeficientes do filtro. A extensão para algoritmos rápidos, FLS, é derivada para ambos os casos de predição linear e identificação.

A partir dos resultados obtidos em [13], chega-se a uma análise mais global do algoritmo, estendendo-a também ao caso Gauss-Newton e LMS, e conseguindo um bom desempenho para diferentes sinais de entrada considerados.

Simulações exaustivas mostram uma certa instabilidade do algoritmo dos mínimos quadrados multidimensional, no início do processamento, devido à aproximação proposta ao problema da não linearidade, como nos casos mais críticos de predição de senóides [14,34]. Uma modificação é feita para superar essa instabilidade inicial, trabalhando-se com o algoritmo de Gauss-Newton, de uma forma mais geral, sendo de grande eficácia, pois no algoritmo proposto, a estabilidade passa a ser controlada por um passo $\delta < 1$, principalmente, no período de transição inicial. Tem-se, assim, desenvolvida uma versão conjunta do algoritmo Gauss-Newton com o FLS-Multidimensional [34].

Antes de apresentar os algoritmos, faz-se a discussão da formulação do problema da filtragem adaptativa na forma cascata.

3.1. FILTRAGEM ÓTIMA NA FORMA EM CASCATA

Deve-se, primeiramente, situar as modificações existentes na análise da filtragem digital ótima quando se trabalha com estruturas de filtros em cascata.

Em geral, são usadas duas formulações. O filtro FIR pode ser considerado como uma fatoração de filtros de primeira ou de segunda ordem em cascata.

Considera-se, aqui, um filtro transversal FIR de ordem $2K$ com uma função de transferência $H(z)$, realizado na forma em cascata, contendo K secções de segunda ordem, ou seja:

$$H(z) = \prod_{k=1}^K \left(h_{0k} + h_{1k} z^{-1} + h_{2k} z^{-2} \right) = \prod_{k=1}^K H_k(z) \quad (3.1)$$

onde h_{ik} ($i=0,1,2$) são os coeficientes de cada secção k ($k=1,\dots,K$), os quais deverão ser calculados de forma a minimizar um determinado critério sobre o erro.

O critério utilizado neste trabalho é o dos mínimos quadrados, onde se considera um conjunto de n dados disponíveis (amostras). Assim, deve-se escolher o filtro ótimo de forma a minimizar a função custo dada por:

$$J(\underline{H}_c) = \frac{1}{2} \sum_{p=1}^n e^2(p) \quad (3.2)$$

que é o critério dos mínimos quadrados, onde \underline{H}_c é o vetor de todos os coeficientes do filtro em cascata, ou seja:

$$\underline{H}_c = [h_{01}, \dots, h_{0K}, h_{11}, \dots, h_{1K}, \dots, h_{21}, \dots, h_{2K}]^T \quad (3.3)$$

e $e(p)$, o erro de estimação.

A Figura 3.1 mostra essa nova estrutura no contexto da filtragem ótima.

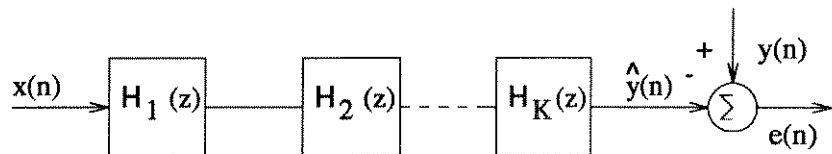


Fig.3.1 - Filtragem Ótima em Cascata

Comparando a Figura 3.1 com a Figura 2.3, observa-se que o erro de estimação é, justamente, o mesmo da filtragem na forma direta.

O conjunto ótimo dos coeficientes h_{ik} , do filtro em cascata, é obtido a partir da solução das equações:

$$\frac{\partial J(\underline{H}_c)}{\partial h_{ik}} = 0 \quad i = 0,1,2 \quad e \quad k = 1,2,\dots, K \quad (3.4)$$

ou seja, a partir da anulação do gradiente da função custo com relação aos coeficientes do filtro, como visto no capítulo anterior. Para se chegar à solução em termos de coeficientes ótimos, deve-se desenvolver a equação 3.4, levando-se

em conta a estrutura da Figura 3.1, onde se expressa o erro $e(n)$, novamente, pela equação 2.1, isto é:

$$e(n) = y(n) - \hat{y}(n) \quad (3.5)$$

Assim, o gradiente para cada coeficiente é tal que:

$$\frac{\partial J(\underline{Hc})}{\partial h_{ik}} = \sum_{p=1}^n \left\{ \frac{\partial e(p)}{\partial h_{ik}} \cdot e(p) \right\} \quad i=0,1,2 \text{ e } k=1,\dots,K \quad (3.6)$$

ou, ainda,

$$\frac{\partial J(\underline{Hc})}{\partial h_{ik}} = - \sum_{p=1}^n \left\{ \frac{\partial \hat{y}(p)}{\partial h_{ik}} \cdot [y(p) - \hat{y}(p)] \right\} \quad i=0,1,2 \text{ e } k=1,\dots,K \quad (3.7)$$

já que o sinal desejado $y(n)$ não é função dos coeficientes do filtro.

Deve-se ressaltar que, nesse caso, o sinal estimado $\hat{y}(n)$ não é mais expresso como regressão linear dos coeficientes do filtro, como no caso transversal. O problema é encontrar uma expressão para o seu gradiente. Para isso, toma-se o sinal filtrado $\hat{y}(n)$ como a transformada Z inversa do sinal de saída, do filtro $H(z)$, cuja entrada tem como transformada $X(Z)$, ou seja [8]:

$$\hat{y}(p) = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} H(z) X(z) dz \quad (3.8)$$

ou, ainda, considerando a fatoração descrita na equação 3.1,

$$\hat{y}(p) = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} \left(\prod_{k=1}^K H_k(z) \right) X(z) dz \quad (3.9)$$

onde Γ é um contorno de integração conveniente.

Assim, a partir da equação acima, pode-se calcular a derivada do sinal $\hat{y}(p)$ em relação aos coeficientes e aplicá-la na expressão (3.7). Tem-se então [22]:

$$\frac{\partial \hat{y}(p)}{\partial h_{ik}} = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} \prod_{\substack{j=1 \\ j \neq k}}^K \left[h_{0j} + h_{1j} z^{-1} + h_{2j} z^{-2} \right] z^{-i} X(z) dz \quad (3.10)$$

ou, ainda:

$$\frac{\partial \hat{y}(p)}{\partial h_{ik}} = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} \frac{H(z)}{H_k(z)} z^{-1} X(z) dz \quad (3.11)$$

Definindo-se, então, o sinal:

$$u_k(p-i) = \frac{\partial \hat{y}(p)}{\partial h_{ik}} \quad i = 0,1,2 \quad e \quad k = 1,\dots,K \quad (3.12)$$

constata-se, de acordo com a equação 3.10, que o sinal $u_k(p-i)$ pode ser visto como sendo a convolução do sinal de entrada $x(n)$ com um filtro em cascata composto por todas as secções de segunda ordem, com exceção da k -ésima secção. De outra forma, pela equação 3.11, pode ser considerado como a passagem do sinal $\hat{y}(n)$ por um filtro IIR inverso à k -ésima secção FIR, $H_k^{-1}(Z)=1/H_k(Z)$. Essa segunda visão é descrita por Jackson e Wood [7] como forma de obtenção dos coeficientes de sensibilidade $u_k(p-i)$.

De qualquer maneira, o sinal $\hat{y}(n)$ pode ser obtido pela filtragem do sinal $u_k(n)$ pela secção $H_k(z)$, cuja expressão matemática é descrita na forma:

$$\hat{y}(p) = \sum_{i=0}^2 h_{ik} u_k(p-i) \quad (3.13)$$

sendo visualizada na Figura 3.2.

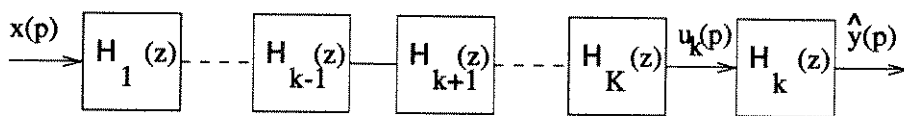


Fig. 3.2 - Obtenção de $u_k(p)$ e $\hat{y}(p)$

Fazendo-se a composição para $k=1,\dots,K$, obtém-se:

$$\hat{y}(p) = \sum_{k=1}^K \sum_{i=0}^2 h_{ik} u_k(p-i) \quad (3.14)$$

O sinal $u_k(p-i)$ se constitui numa das componentes do vetor gradiente $\underline{\nabla}_{\underline{H}_c} [\hat{y}(p)]$, dado por:

$$\underline{U}(p) = \left[u_1(p), \dots, u_K(p), u_1(p-1), \dots, u_K(p-1), u_1(p-2), \dots, u_K(p-2) \right]^T \quad (3.15)$$

Considerando o vetor \underline{H}_c dos coeficientes dos filtros em cascata, definido em (3.3), pode-se escrever a equação 3.14 na forma vetorial:

$$\hat{y}(p) = \frac{\underline{H}_c^T \cdot \underline{U}(p)}{K} \quad (3.16)$$

Quando se considera \underline{H}_c ótimo, que satisfaz (3.4), a consequente anulação da equação (3.7) leva à igualdade:

$$\sum_{p=1}^n \hat{y}_{ot}(p) \cdot \underline{U}(p) = \sum_{p=1}^n y(p) \cdot \underline{U}(p) \quad (3.17)$$

para todos os elementos do vetor gradiente definido em (3.15).

Substituindo $\hat{y}(p)$ da equação 3.16 na equação 3.17, obtém-se:

$$\sum_{p=1}^n \underline{U}(p) \cdot \underline{U}^T(p) \cdot \underline{H}_{c_{ot}} = K \sum_{p=1}^n y(p) \underline{U}(p) \quad (3.18)$$

que é um sistema de equações normais não lineares, podendo ser reescrito como:

$$\underline{R}_{uu} \cdot \underline{H}_{c_{ot}} = \underline{R}_{yu} \quad (3.19)$$

donde se calcula:

$$\underline{H}_{c_{ot}} = \underline{R}_{uu}^{-1} \cdot \underline{R}_{yu} \quad (3.20)$$

sendo

$$\underline{R}_{uu} = \sum_{p=1}^n \underline{U}(p) \underline{U}^T(p) \quad (3.21)$$

a matriz de intercorrelação dos elementos do sinal intermediário e

$$\underline{R}_{yu} = \sum_{p=1}^n K y(p) \underline{U}^T(p) \quad (3.22)$$

o vetor de correlação entre os elementos do vetor gradiente e o sinal de referência. A não linearidade da equação 3.19 está expressa no fato que o vetor gradiente e, conseqüentemente, a matriz \underline{R}_{uu} e o vetor \underline{R}_{yu} , são funções dos próprios parâmetros dos filtros. Essa não linearidade é uma conseqüência direta da filtragem com estruturas em cascata.

A solução da equação 3.20 está associada àquela da estrutura transversal. Embora os parâmetros h_{ik} não sejam os mesmos, a solução da equação 3.20 deve corresponder aos mesmos pontos de mínimo da superfície de erro da estrutura transversal, pois a minimização é feita em relação à mesma norma do erro. Assim, os coeficientes devem ser tais que:

$$\prod_{k=1}^K (h_{0k} + h_{1k} z^{-1} + h_{2k} z^{-2}) = \sum_{i=0}^M h_i z^{-i} \quad (3.23)$$

onde h_i são os coeficientes do filtro transversal. A análise da curva de erro dos filtros em cascata será vista no capítulo seguinte.

Pode-se normalizar os estágios do filtro em função de h_{0k} , de tal forma que:

$$H(z) = H_0 \left[\prod_{k=1}^K (1 + h_{1k} z^{-1} + h_{2k} z^{-2}) \right] \quad (3.24)$$

onde H_0 é o ganho do filtro e h_{ik} a posição dos zeros da resposta em freqüência. Esses filtros são chamados mônicos. Dessa forma, para coeficientes reais, considerando as raízes z_k complexas, tem-se:

$$h_{1k} = -2\text{Re}(z_k), \quad h_{2k} = |z_k|^2 \quad (3.25)$$

sendo que essas raízes estarão dentro do círculo unitário, ou seja, os filtros serão de fase mínima se:

$$|h_{2k}| < 1, \quad |h_{1k}| < 1 + h_{2k} \quad (3.26)$$

3.1.1. Caso da Predição Linear em Cascata

Tomando-se, agora, o filtro $H(z)$ como sendo um filtro de erro de predição linear, tem-se, para cada secção de segunda ordem, a função de transferência:

$$P_k(z) = 1 - a_{1k}z^{-1} - a_{2k}z^{-2} \quad (3.27)$$

de sorte que:

$$P(z) = \prod_{k=1}^K P_k(z) = 1 - \sum_{i=1}^{M=2K} a_i z^{-i} \quad (3.28)$$

onde a_i são os coeficientes do filtro predictor transversal de ordem $2K$, aplicados ao mesmo sinal $x(n)$. Observa-se que os zeros de $P(z)$ são facilmente obtidos da fatoração quadrada da equação 3.27. Essa formulação é apropriada para sinais de valor real.

A mesma formulação apresentada na seção anterior para o caso geral da filtragem ótima pode ser explorada considerando na equação 3.24, $a_{ik} = -h_{ik}$ ($i = 1, 2; k = 1, \dots, K$) e $H_0 = 1$. Dessa forma, para a localização dos pontos estacionários da curva de erro quadrático da equação 3.2, deve-se usar a expressão do erro de predição no desenvolvimento da equação 3.4, tal que:

$$\frac{\partial J(\underline{A}_c)}{\partial a_{ik}} = \sum_{p=1}^n \left\{ \frac{\partial e(p)}{\partial a_{ik}} \cdot e(p) \right\} \quad (3.29)$$

onde \underline{A}_c é o vetor de coeficientes de predição do filtro em cascata, tal que:

$$\underline{A}_c = \left[a_{11} \dots a_{1K} a_{21} \dots a_{2K} \right]^T \quad (3.30)$$

Assim, $e(n)$, da equação 2.35, é a transformada Z inversa da saída $E(z)$, do filtro de erro de predição $P(z)$ quando se aplica um sinal $x(n)$, ou seja

$$e(p) = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} P(z) x(z) dz \quad (3.31)$$

Observa-se, utilizando o mesmo raciocínio da seção anterior, que o sinal que constitui a derivada do erro, em relação a um dado parâmetro a_{ik} , é obtido pela filtragem de $x(n)$ por um conjunto de secções, exceto pela k -ésima. Assim:

$$u_k(p-1) = \frac{\partial e(p)}{\partial a_{1k}} = \frac{1}{2\pi j} \int_{\Gamma} z^{(p-1)} \frac{P(z)}{P_k(z)} z^{-1} X(z) dz \quad (3.32)$$

e, conseqüentemente, filtrando-se $u_k(p)$ por essa k -ésima secção, tem-se, agora, o erro de predição:

$$e(p) = u_k(p) - a_{1k} u_k(p-1) - a_{2k} u_k(p-2) \quad (3.33)$$

e estendendo para todas as secções, com $k=1..K$, de forma análoga à equação 3.14, chega-se a:

$$K e(p) = \sum_{k=1}^K u_k(p) - \sum_{k=1}^K \left[a_{1k} u_k(p-1) + a_{2k} u_k(p-2) \right] \quad (3.34)$$

A geração dos componentes gradientes dos filtros de erro de predição linear em cascata pode ser vista, na forma da equação 3.32, como uma filtragem IIR de acordo com a Figura 3.3 [7].

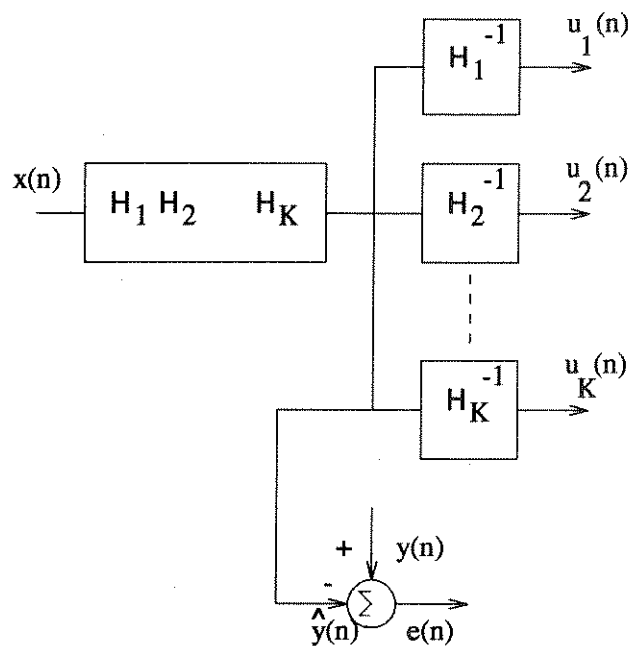


Fig.3.3 - Geração das seqüências gradientes por recursão

Considerando, agora, o vetor $\underline{U}(p)$ dos sinais intermediários com $2K$ elementos:

$$\underline{U}(p) = \left[u_1(p) \dots u_K(p) u_1(p-1) \dots u_K(p-1) \right]^T \quad (3.35)$$

e reescrevendo a equação do erro de predição, a partir da equação 3.34, tomando o sinal de referência :

$$y(p) = \frac{1}{K} \sum_{k=1}^K u_k(p) \quad (3.36)$$

e o sinal filtrado,

$$\hat{y}(p) = \frac{1}{K} \underline{A}_c^T \underline{U}(n-1) \quad (3.37)$$

pode-se chegar às equações do preditor ótimo, a partir das equações normais, de acordo com a equação (3.17), agora escritas como:

$$\underline{R}_{uu} \underline{A}_c = \underline{R}_{\underline{u}u} \quad (3.38)$$

com:

$$\underline{R}_{uu} = \sum_{p=1}^n \underline{U}(p) \underline{U}^T(p) \quad (3.39)$$

e

$$\underline{R}_{\underline{u}u}^T = \sum_{p=1}^n \left[u_1(p) + \dots + u_K(p) \right] \underline{U}(p-1) \quad (3.40)$$

donde,

$$\underline{A}_c = \underline{R}_{uu}^{-1} \cdot \underline{R}_{\underline{u}u} \quad (3.41)$$

que são as equações normais não lineares, já que os parâmetros a_{ik} do filtro preditor interferem na obtenção dos sinais gradientes $u_k(p)$ e, por conseguinte, na matriz \underline{R}_{uu} e no vetor $\underline{R}_{\underline{u}u}$. Embora essas equações sejam não lineares sobre os parâmetros $\{a_{ik}\}$, suas soluções devem corresponder aos mesmos pontos na superfície de erro (com respeito às novas coordenadas do sistema \underline{A}_c) da solução do filtro ótimo na forma direta, pois ambas as equações normais utilizam o gradiente da mesma função custo.

3.2. TÉCNICAS ITERATIVAS PARA SISTEMAS NÃO LINEARES

Como visto, no capítulo anterior, algumas técnicas iterativas de otimização são usadas para obtenção dos coeficientes ótimos, ao invés da solução analítica das equações normais, principalmente neste caso, onde as equações normais são não lineares. Assim, a partir da expansão em séries de Taylor da função custo, tendo como relação básica a restrição que:

$$\underline{H}_c \longrightarrow \underline{H}_{c_{i+1}} \quad \text{tal que} \quad J \Big|_{\underline{H}_{c_{i+1}}} < J \Big|_{\underline{H}_{c_i}} \quad (3.42)$$

define-se, de uma forma geral, o processo iterativo da forma:

$$\underline{H}_{c_{i+1}} = \underline{H}_{c_i} - \gamma \left[\underline{B}^{-1} \cdot \underline{S} \right]_{\underline{H}_{c_i}} \quad (3.43)$$

onde \underline{H}_c é o vetor de coeficientes do filtro em cascata, definido pela equação 3.3, γ é um escalar definido na seção 2.4, \underline{S} é o vetor gradiente da função custo em relação aos coeficientes de \underline{H}_c que, a partir da equação 3.7, pode ser escrito na forma

$$\underline{S} = \sum_{p=1}^n \underline{U}(p) \cdot e(p) \quad (3.44)$$

e \underline{B} é a matriz escolhida de acordo com o método de aproximação, ou seja, define a direção de busca. Se \underline{B} é a matriz identidade, tem-se o método do Gradiente. Escolhendo-se \underline{B} como a Hessiana da função custo, em relação ao vetor \underline{H}_c , define-se, novamente o método de Newton-Raphson. Nesse caso, o fato da superfície não ser convexa leva a se ter uma região de convergência muito pequena.

O método Quase-Newton é uma aproximação do método de Newton no qual, toma-se a matriz \underline{B} a partir da expansão da hessiana,

$$\frac{\partial J}{\partial \underline{H}_c \partial \underline{H}_c^T} = \sum_{p=1}^n \frac{\partial e(p)}{\partial \underline{H}_c} \frac{\partial e(p)}{\partial \underline{H}_c^T} + \sum_{p=1}^n \frac{\partial^2 e(p)}{\partial \underline{H}_c \partial \underline{H}_c^T} \cdot e(p) \quad (3.45)$$

desprezando-se o termo de segunda ordem, ou seja:

$$\underline{B} = \sum_{p=1}^n \underline{U}(p) \cdot \underline{U}^T(p) = \underline{R}_{uu} \quad (3.46)$$

sendo aproximada como a matriz de intercorrelação dos elementos do vetor gradiente.

Deve-se lembrar que a expansão em série de Taylor é somente uma aproximação localizada da superfície do erro, sendo necessário incluir, sempre, o fator $0 < \gamma < 1$ para garantir a não extrapolação da região de validade.

Vale ressaltar que, como a superfície $J(\underline{H}_c)$ não é convexa, em termos dos coeficientes do filtro em cascata, existem $K!$ mínimos (devido à permutação entre as secções), implicando em um número proporcional de pontos de sela. A Figura 3.4 mostra os contornos dessa superfície para o filtro em cascata com duas secções de primeira ordem, correspondendo ao caso de duas raízes reais de um polinômio quadrático [7]. Essa discussão será tratada com maior rigor no próximo capítulo.

As técnicas iterativas, lembradas brevemente nesta seção, foram aplicadas no caso cascata em [7]. Nessa referência, é proposto um tratamento em bloco dos dados, partindo de uma determinada condição inicial e efetuando as iterações em (3.43), sendo \underline{S} e \mathbf{B} calculados segundo o método escolhido.

Tendo em vista uma aplicação em tempo real ("on line"), o interesse nesta tese é obter os coeficientes ótimos do filtro em cascata através de um algoritmo adaptativo, que atualize os coeficientes a cada nova entrada recebida. A derivação e análise desse algoritmo é exposta a seguir.

3.3. FILTRAGEM ADAPTATIVA NA FORMA EM CASCATA

Um algoritmo adaptativo a ser aplicado à filtragem em cascata deve levar em conta o fato da superfície de erro, nesse caso, não ser convexa. Apesar de a filtragem em cascata se caracterizar por um problema de otimização não-linear, é possível se desenvolver algoritmos a partir de aproximações que linearizem o problema de otimização, como extensão em tempo real do processo iterativo de busca. Nesse sentido, algumas técnicas já foram propostas na literatura, usando o algoritmo LMS, de acordo com a Figura 3.3 [23].

A técnica a ser proposta aqui tem sua originalidade no fato de fazer intervir uma estrutura multidimensional e possibilitar o uso de algoritmos do tipo mínimos quadrados e Gauss-Newton, melhorando em diversos aspectos o desempenho em relação ao LMS.

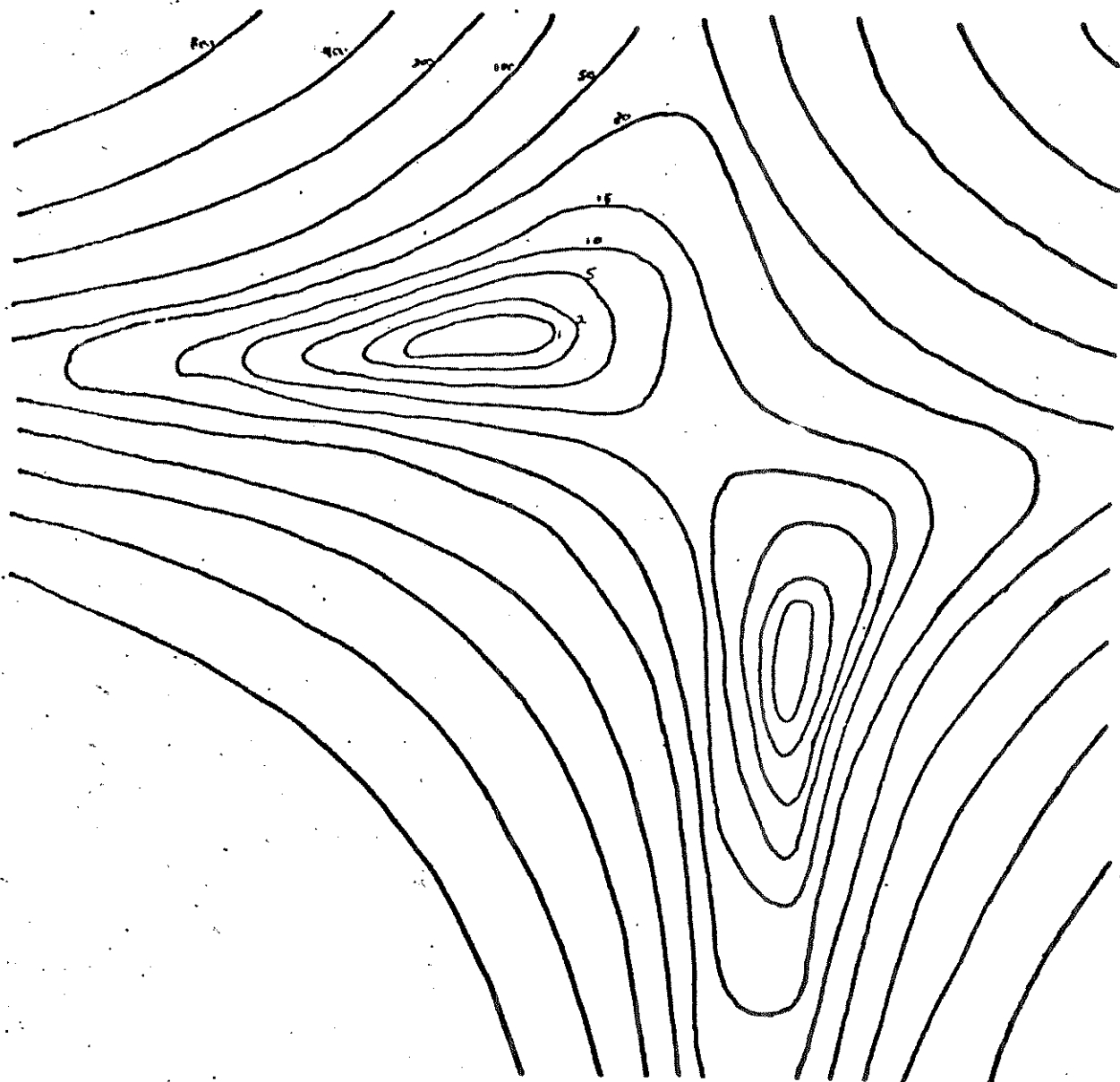


Fig. 3.4. Contornos Típicos da Superfície de Erro para Polinômio Quadrático [7]

Primeiramente, vai-se desenvolver a proposta a partir do método de mínimos quadrados e, em seguida, analisar também o problema sob a ótica da aproximação estocástica.

3.3.1. Solução Proposta: Configuração Multidimensional

De modo a estabelecer uma ligação entre o filtro cascata e a estrutura multidimensional proposta neste trabalho, considere o esquema da Figura 3.2 onde, para cada $k = 1, \dots, K$, tem-se a geração de um sinal $u_k(n)$ e da saída $\hat{y}(n)$ como nas equações (3.12) e (3.13). Fazendo a composição de todos os K índices, pode-se representar essas operações de forma paralela, sendo que o esquema total da filtragem é mostrado na Figura 3.5.

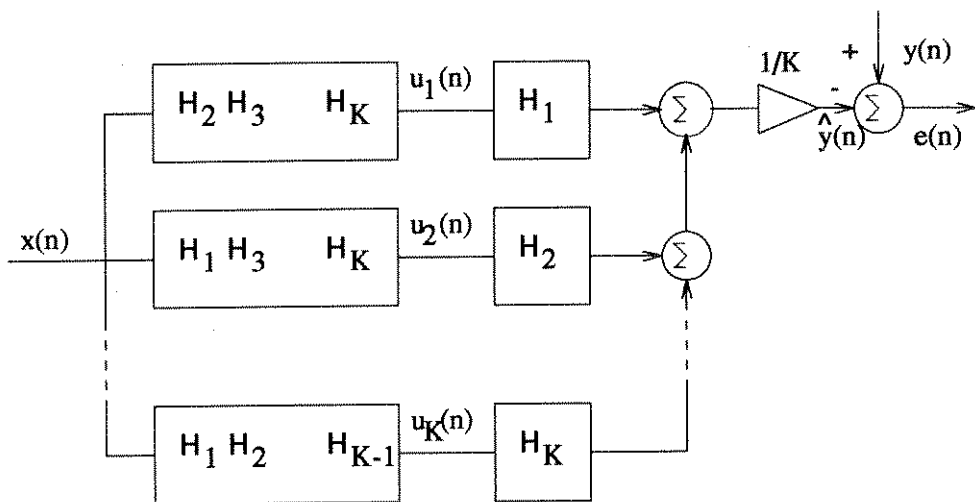


Figura 3.5. Estrutura de Filtragem Multidimensional Proposta

Nesse esquema, visualiza-se os sinais $u_k(n)$ como elementos de uma entrada vetorial $\underline{Y}(n)$, de dimensão K , e o conjunto de secções de segunda ordem como um filtro FIR multidimensional adaptativo, o qual pode ser atualizado a partir do sinal de erro obtido na sua saída.

O vetor definido na equação 3.3 pode ser visto como o conjunto de coeficientes da estrutura multidimensional, devendo ser atualizado, a partir do vetor de dados de entrada $\underline{Y}(n)$, na busca do filtro ótimo.

O algoritmo dos mínimos quadrados, aplicado a essa configuração, nos leva a recursão

$$\underline{H}_c(n+1) = \underline{H}_c(n) + \mathbf{R}_{uu}^{-1}(n+1) \underline{U}(n+1) e(n+1) \quad (3.47)$$

a ser complementada pela estimação recursiva da matriz $\mathbf{R}_{uu}^{-1}(n)$, dada por

$$\mathbf{R}_{uu}(n+1) = W \mathbf{R}_{uu}(n) + \underline{U}(n+1) \underline{U}^T(n+1) \quad (3.48)$$

onde W é o fator de esquecimento.

De fato, utilizando o mesmo desenvolvimento já apresentado na seção 2.7 desta tese, poder-se-ia obter as recursões (3.47 e 3.48) diretamente a partir das equações normais 3.19, 3.21 e 3.22 já que essas podem ser vistas como similares às de uma configuração FIR multidimensional (equações 2.158, 2.159 e 2.160). Assim, ter-se-ia para n dados considerados:

$$\underline{H}_c(n) = \mathbf{R}_{uu}^{-1}(n) \cdot \underline{R}_{yn}(n) \quad (3.49)$$

obtendo, por recorrência, no instante $n+1$:

$$\underline{H}_c(n+1) = \mathbf{R}_{uu}^{-1}(n+1) \cdot \underline{R}_{yu}(n+1) \quad (3.50)$$

com

$$\mathbf{R}_{uu}(n) = \sum_{p=1}^n W^{n-p} \underline{U}(p) \cdot \underline{U}^T(p) \quad (3.51)$$

como a matriz estimada de intercorrelação dos sinais $u_k(n)$ e

$$\underline{R}_{yu}(n) = \sum_{p=1}^n W^{n-p} y(p) \cdot \underline{U}^T(p) \quad (3.52)$$

o vetor de correlação cruzada.

Contudo, devido à não linearidade inerente da formulação em cascata, não se pode desenvolver as manipulações de recursão nessas equações, uma vez que $\mathbf{R}_{uu}(n)$ é tomado com os dados de $\underline{U}(p)$ ($0 < p < n$), obtidos a partir dos coeficientes $\underline{H}_c(n)$, sendo que $\underline{U}(n+1)$ é função de $\underline{H}_c(n+1)$.

A aproximação básica no sentido de linearizar o problema é supor que o conjunto de coeficientes sofre variações suficientemente pequenas, do instante n para $n+1$, de forma que se permita a recursão das equações 3.47 e 3.48. Parte-se, então da hipótese que o vetor gradiente, nas condições expostas, não depende do vetor de coeficientes. De fato, esse tipo de aproximação está necessariamente presente quando se utiliza algoritmos de aproximação estocástica em quaisquer outras estruturas que não conduzam a um problema de otimização linear, como por exemplo as IIR [3,5]. Assim, a equação (3.47) também pode ser vista como derivada da equação de Robbin-Monro em (2.89) onde a escolha da matriz define o algoritmo.

Assim, uma primeira aproximação para a atualização dos coeficientes do filtro seria a utilização do algoritmo do gradiente [25], mostrada na Fig. 3.6. Nesse caso, não se considera a intercorrelação entre os elementos do vetor gradiente, $u_k(p-1)$, $i=0,1,2$ e $k=1,\dots,K$, supondo $R_{uu}(n) = I$, de forma que cada seção de segunda ordem é adaptada separadamente, sendo a adaptação efetuada como segue:

$$h_{ik}(n) = h_{ik}(n-1) + \mu u_k(n-1) e(n) \quad i = 0,1,2 \quad e \quad k = 1,\dots,K \quad (3.53)$$

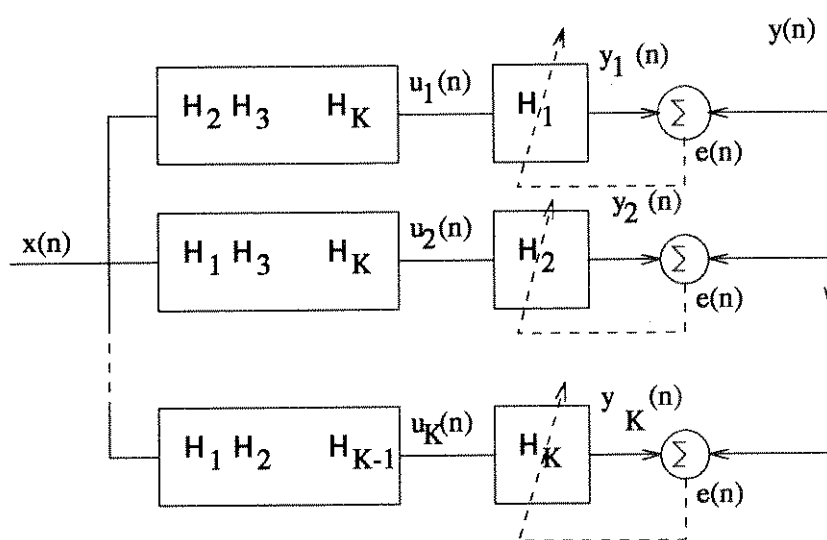


Fig.3.6 - Filtro Adaptativo LMS em Cascata

Alguns problemas importantes são evidenciados no algoritmo gradiente na forma em cascata. Primeiro, é de se esperar que quanto mais os sinais forem bem correlacionados, mais degradado será o desempenho do algoritmo. Além disso, problemas de implementação da estrutura em cascata podem levar a situações de dificuldades no processo de adaptação. Por exemplo, devido ao fato da função custo não ser convexa, pode haver problemas de convergência. A permutação entre as secções de segunda ordem nos leva a uma curva de erro formada por várias "cúias", podendo fazer com que um algoritmo estocástico, a cada instante, busque caminhos de descida diferentes, tendo novamente problemas de convergência. Além disso, o fato das secções serem adaptadas separadamente, pode induzir a que mais de uma secção, que está sendo adaptada, tenda a mesmos valores de coeficientes. Estudos sobre a superfície de erro quadrático, que será abordado no próximo capítulo, mostram que tal situação corresponde a pontos de sela da curva. Devido a esses problemas, o algoritmo do gradiente em estruturas em cascata requer, obrigatoriamente, que as condições iniciais sejam diferentes, devendo-se ter cuidado na escolha dessas.

Em contrapartida, o fato de se trabalhar com secções de segunda ordem garante, em cada estágio, um espalhamento pequeno comparado com a estrutura transversal, levando-se a considerar que o algoritmo do gradiente, nesse caso, tenha uma convergência bem mais rápida que o seu correspondente na forma direta, desde que se inicie o processo de adaptação devidamente.

Outra maneira de linearizar o problema levantado nas recursões (3.47) e (3.48) é através do algoritmo de Gauss-Newton multidimensional, onde se teria:

$$\underline{H}c(n+1) = \underline{H}c(n) + \gamma \mathbf{R}_{uu}^{-1}(n+1) \underline{U}(n+1) e(n+1), \quad (3.54)$$

sendo $\mathbf{R}_{uu}(n+1)$ a matriz de intercorrelação, positiva definida, obtida recursivamente a partir do algoritmo da aproximação estocástica, Robbins-Monro, da forma:

$$\mathbf{R}_{uu}(n+1) = \mathbf{R}_{uu}(n) + \alpha \left[\underline{U}(n+1) \underline{U}^T(n+1) - \mathbf{R}_{uu}(n) \right] \quad (3.55)$$

similar à equação 2.119. Considera-se, agora, que duas aproximações são requeridas. A primeira, com relação ao gradiente, da mesma forma do LMS em cascata. A segunda aproximação é para se garantir a recorrência da estimativa da matriz de intercorrelação, considerando α muito pequeno. Dessa forma, com a validação de equação 3.49, obtém-se, pelo lema de inversão de matrizes, a matriz inversa $\mathbf{P}_{uu}(n) = \mathbf{R}_{uu}^{-1}(n)$, resultando na seguinte atualização dessa matriz:

$$P_{uu}(n+1) = \frac{1}{1-\alpha} P_{uu}(n) - \frac{\alpha}{1-\alpha} \left(\frac{P_{uu}(n) \underline{U}(n) \underline{U}^T(n) P_{uu}(n)}{(1-\alpha) + \alpha \underline{U}^T(n) R_{uu}(n) \underline{U}(n)} \right) \quad (3.56)$$

É de se esperar um algoritmo com maior rapidez de convergência, comparando-se com o algoritmo do gradiente, embora com maior complexidade computacional. Porém, devido às aproximações feitas, deve-se ter um controle da estabilidade do algoritmo muito mais rigoroso durante o período transiente inicial.

A Figura 3.7 mostra a estrutura em cascata utilizando o princípio dos mínimos quadrados multidimensional, como consequência direta da Figura 3.5, considerando a adaptação em todas as secções, ao mesmo tempo, similar ao desenvolvido na seção 2.7 do capítulo anterior.

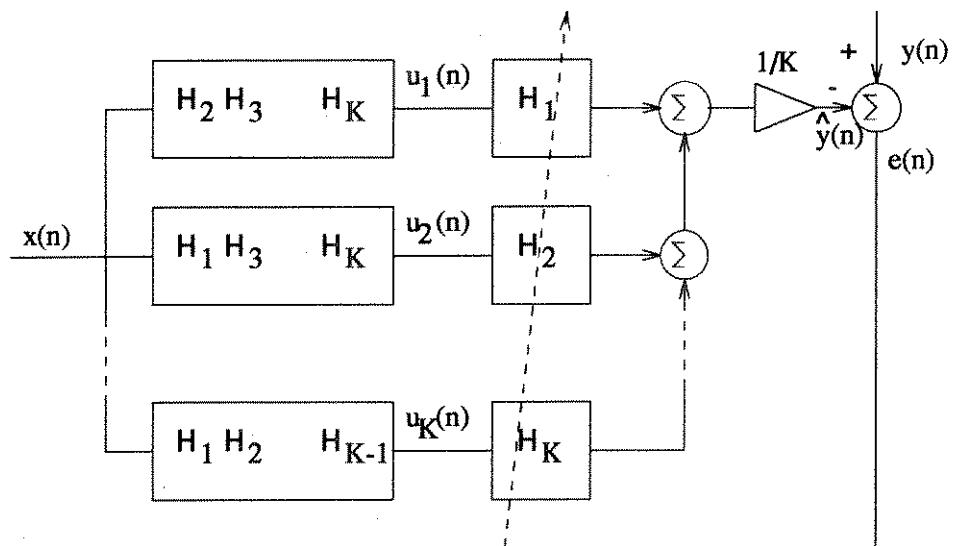


Fig.3.7 - O Filtro Adaptativo LS em Cascata

Finalmente, supondo o procedimento adaptativo da equação 3.47, um algoritmo rápido pode ser derivado para essa estrutura proposta, reduzindo a complexidade computacional e fornecendo uma interessante solução para o caso da predição linear.

3.4. ALGORITMO RÁPIDO MULTIDIMENSIONAL: FLS-MD

Como mostrado na Figura 3.7, as secções são atualizadas quando o novo vetor de dados de entrada $\underline{Y}(n+1)$ é disponível. Um algoritmo rápido pode ser implementado para prover a atualização do vetor $\underline{H}_c(n)$, uma vez que o vetor $\underline{U}(n)$ tem a propriedade de deslocamento de blocos [23], ou seja, pode-se obter o vetor $\underline{U}(n+1)$ a partir do vetor $\underline{U}(n)$ com a chegada do novo vetor de entrada $\underline{Y}(n+1)$, descartando o conjunto de dados mais antigos, $\underline{Y}(n-2)$.

Assim, deve-se observar o comportamento do vetor gradiente $\underline{U}(n)$, onde, a partir da definição 3.15, pode-se escrever:

$$\underline{U}(n) = \left[u_1(n), \dots, u_K(n), \dots, u_1(n-M+1), \dots, u_K(n-M+1) \right]^T \quad (3.57)$$

cujos elementos são obtidos a partir dos coeficientes de $\underline{H}_c(n)$. Com as suposições feitas anteriormente, obtém-se o vetor deslocado:

$$\underline{U}(n+1) = \left[u_1(n+1), \dots, u_K(n+1), \dots, u_1(n-M+2), \dots, u_K(n-M+2) \right]^T \quad (3.58)$$

Assim, a chegada de um novo dado $x(n+1)$ leva ao aparecimento de K novos valores $u_k(n+1)$, verificando-se a propriedade de deslocamento de blocos de K dados, fundamental para o algoritmo dos mínimos quadrados rápido [23]. Com isso, pode-se desenvolver uma versão rápida seguindo os mesmos passos do algoritmo dos mínimos quadrados recursivo multidimensional, fazendo uma adaptação interna do vetor de ganho $\underline{G}(n)$, de dimensão KM , definido a partir da equação 3.47, como na equação 2.162, na forma:

$$\underline{G}(n) = \mathbf{R}_{uu}^{-1}(n) \underline{U}(n) \quad (3.59)$$

Tal como mostrado na Figura 3.7, uma iteração do algoritmo é efetuada com a chegada de K dados $u_k(n)$, que são filtragens do sinal de entrada $x(n)$ pelo conjunto de secções, exceto a k -ésima. Os passos e as definições do processo de adaptação da filtragem em cascata, utilizando o algoritmo dos mínimos quadrados rápido multidimensional (FLS-MD), estão dispostos de acordo com a seção 2.7 do capítulo anterior, incluindo a recursão da inversa da matriz de erro quadrático, $\mathbf{E}_a(n)$. Novamente, é importante destacar algumas definições antes da apresentação do algoritmo completo.

O algoritmo utiliza os parâmetros de predição linear como desenvolvimento da recursão do vetor ganho. Assim, tomando a chegada de um novo vetor de dados filtrados $\underline{Y}(n+1)$, como na Figura 3.7, o erro de predição progressiva é calculado como um vetor de K elementos,

$$\underline{e}_a(n+1) = \underline{Y}(n+1) - \underline{A}^T(n) \underline{U}(n) \quad (3.60)$$

com os coeficientes de predição constituindo uma matriz $\underline{A}(n)$ de dimensão $(MK) \times K$, onde M é o número de coeficientes por secção. Essa matriz $\underline{A}(n)$ é calculada de maneira a minimizar a energia do erro de predição, dada por:

$$E_a(n) = \sum_{p=1}^n \underline{e}_a^T(p) \cdot \underline{e}_a(p) \quad (3.61)$$

O processo de minimização desse critério fornece os valores ótimos através das equações:

$$\underline{A}(n+1) = \underline{R}_{uu}^{-1}(n) \underline{r}_{aK}(n+1) \quad (3.62)$$

onde $\underline{r}_{aK}(n+1)$ é uma matriz de correlação entre os sinais $u_k(n+1)$ e seus valores passados para todos os índices k , isto é:

$$\underline{r}_{aK}(n+1) = \sum_{p=1}^{n+1} \underline{U}(p-1) \underline{Y}^T(p) \quad (3.63)$$

com a nova matriz de coeficientes obtida de forma recursiva, tal que:

$$\underline{A}(n+1) = \underline{A}(n) + \underline{G}(n) \underline{e}_a^T(n+1) \quad (3.64)$$

Os parâmetros da predição regressiva são obtidos de forma similar. Dispondo desses parâmetros, os coeficientes do vetor $\underline{H}_c(n+1)$ são computados pelo algoritmo FLS-MD, como mostrado na Tabela 3.1. Um típico valor inicial para os parâmetros é dado tomando a matriz de erro como uma matriz diagonal com termos diferentes, ou seja, toma-se como valor inicial da matriz de erro $\underline{E}_a(0)$ a matriz diagonal:

$$\underline{E}_a = \begin{bmatrix} E_1 & & & 0 \\ & E_2 & & \\ & & \ddots & \\ 0 & & & E_K \end{bmatrix} \quad (3.65)$$

Quantidades disponíveis no instante n:

$\underline{H}_c(n)$ - Vetor de coeficientes do filtro adaptativo com KM elementos (M - número de coeficientes por secção)

$\underline{A}(n)$ - Matriz de predição progressiva (KM linhas, K colunas)

$\underline{B}(n)$ - Matriz de predição regressiva (KM linhas, K colunas)

$\underline{U}(n)$ - Vetor de dados com KM elementos

$\underline{G}(n)$ - Vetor ganho de adaptação com KM elementos

$\underline{E}_a(n)$ - Matriz erro de predição quadrático (K, K)

Novos dados no instante (n+1):

$x(n+1)$ - Dado recebido

$y(n+1)$ - Sinal de referência

$\underline{Y}(n+1)$ - Entrada multidimensional

Atualização do vetor ganho de adaptação:

- (1) Vetor erro de predição progressiva à priori (K elementos)

$$\underline{e}_a(n+1) = \underline{Y}(n+1) - \underline{A}^T(n)\underline{U}(n)$$

- (2) Matriz de predição progressiva

$$\underline{A}(n+1) = \underline{A}(n) + \underline{G}(n)\underline{e}_a^T(n+1)$$

- (3) Vetor erro de predição progressiva à posteriori (K elementos)

$$\underline{\varepsilon}_a(n+1) = \underline{Y}(n+1) - \underline{A}^T(n+1)\underline{U}(n)$$

- (4) Matriz erro de predição quadrático

$$\underline{E}_a(n+1) = W \underline{E}_a(n) + \underline{e}_a(n+1)\underline{e}_a^T(n+1)$$

- (5) Vetor ganho de adaptação intermediário (KM+K elementos)

$$\underline{G}_1(n+1) = \begin{bmatrix} 0 \\ \underline{G}(n) \end{bmatrix} + \begin{bmatrix} \underline{I} \\ -\underline{A}(n+1) \end{bmatrix} \underline{E}_a^{-1}(n+1) \underline{\varepsilon}_a(n+1) = \begin{bmatrix} \underline{M}(n+1) \\ \underline{m}(n+1) \end{bmatrix}$$

- (6) Vetor erro de predição regressivo à priori

$$\underline{e}_b(n+1) = \underline{Y}(n+1-M) - \underline{B}^T(n)\underline{U}(n+1)$$

- (7) Vetor Ganho de adaptação

$$\underline{G}(n+1) = [1 - \underline{e}_b^T(n+1)\underline{m}(n+1)]^{-1} [\underline{M}(n+1) + \underline{B}(n)\underline{m}(n+1)]$$

- (8) Matriz de progressão regressiva

$$\underline{B}(n+1) = \underline{B}(n) + \underline{G}(n+1)\underline{e}_b^T(n+1)$$

atualização dos coeficientes do filtro adaptativo cascata:

- (9) Sinal de erro

$$e(n+1) = y(n+1) - (1/K)[\underline{H}_c^T(n)\underline{U}(n+1)]$$

- (10) Coeficientes

$$\underline{H}_c(n+1) = \underline{H}_c(n) + \underline{G}(n+1)e(n+1)$$

Tabela 3.1 - Algoritmo Multidimensional Proposto

Considera-se nulas todas as outras variáveis, exceto os primeiros coeficientes de cada seção, $h_{ok}(0)=1$. Essa imposição, diferente do algoritmo da seção 2.7, vem do fato de que, como no algoritmo do gradiente, a não diferenciação das seções a serem adaptadas pode levar a uma convergência a pontos de sela. Por outro lado, nenhuma outra condição seria necessária sobre os coeficientes, flexibilizando-se assim o funcionamento do algoritmo.

3.4.1. Aplicação à Predição Linear

A estrutura de um preditor em cascata é dada, aqui, pela função de transferência:

$$P(z) = \prod_{k=1}^K (1 - a_{1k} z^{-1} - a_{2k} z^{-2}) \quad (3.66)$$

A aplicação à predição linear na forma em cascata dos algoritmos LMS e RLS-multidimensional é feita da mesma maneira que na obtenção dos coeficientes do filtro, ou seja, a partir das equações 3.47 e 3.53, desenvolvidas anteriormente, usando o vetor de coeficientes do filtro preditor definido em 3.30 como solução da equação 3.41. Já utilizando o algoritmo rápido, FLS-multidimensional, não se tem mais o vetor de coeficientes da predição progressiva direto do algoritmo, como no caso monodimensional. Tem-se que, a estrutura para o cálculo do vetor ganho de adaptação utiliza matrizes de predição para se determinar um vetor de erro de predição progressiva. É conveniente entender o significado dos elementos dessa matriz.

O cálculo recursivo do vetor de coeficientes do filtro de predição, $\underline{A}_c(n)$, pode ser efetuado diretamente da matriz $\underline{A}(n)$, observando-se os elementos dos vetores e da matriz $\underline{A}(n)$ da equação 3.60. Reescrevendo, tem-se:

$$\begin{bmatrix} e_{a1}(n) \\ \vdots \\ e_{aK}(n) \end{bmatrix} = \begin{bmatrix} u_1(n) \\ \vdots \\ u_K(n) \end{bmatrix} - \begin{bmatrix} a_{-1}^T(n-1) \\ \vdots \\ a_{-K}^T(n-1) \end{bmatrix} \underline{U}(n-1) \quad (3.67)$$

onde $\underline{a}_{-k}(n)$ representa a k-ésima coluna dessa matriz. Esse vetor coluna é constituído de $2K$ elementos e fornece uma predição do sinal $u_k(n)$ a partir dos valores passados dos sinais $u_j(n)$, $j=1, \dots, K$. Seus valores ótimos são dados pela equação 3.62, que pode ser reescrita da forma expandida:

$$\left[\underline{a}_1(n), \dots, \underline{a}_K(n) \right] = \mathbf{R}_{uu}^{-1}(n-1) \left[\underline{r}_{a1}(n), \dots, \underline{r}_{aK}(n) \right] \quad (3.68)$$

onde $\underline{r}_{ak}(n)$ é a k-ésima coluna da matriz $\underline{r}_{aK}(n)$, escrita na equação 3.63 como:

$$\underline{r}_{aK}(n+1) = \sum_{p=1}^{n+1} \underline{U}(p-1) \left[\underline{u}_1(p) \dots \underline{u}_K(p) \right] \quad (3.69)$$

levando a se determinar:

$$\underline{r}_{ak}(n) = \sum_{p=1}^n \underline{U}(p-1) \underline{u}_k(p) \quad (3.70)$$

Considerando, agora, a definição dos coeficientes ótimos, dado pela equação 3.41, com $\underline{R}_{uu}(n)$ definido na equação 3.40, ou seja:

$$\underline{A}_c(n) = \mathbf{R}_{uu}^{-1}(n-1) \cdot \underline{R}_{uu}(n) \quad (3.71)$$

e tomando a relação entre a matriz $\underline{r}_{aK}(n)$ e o vetor \underline{R}_{uu} :

$$\underline{R}_{uu}(n) = \sum_{k=1}^K \underline{r}_{ak}(n) \quad (3.72)$$

pode-se chegar nos coeficientes ótimos de predição substituindo (3.72) em (3.71) e comparando com a equação (3.68). Tem-se então que os coeficientes do preditor em cascata podem ser obtidos a partir da matriz de predição da estrutura multi-dimensional, pela relação:

$$\underline{A}_c(n) = \sum_{k=1}^K \underline{a}_k(n) \quad (3.73)$$

Assim, para o caso da predição linear, um conjunto de equações do tipo das equações 3.60, 3.64 e 3.73 substituem as equações de atualização do filtro, na Tabela 3.1. Deve-se observar, entretanto, o número K de secções na equação 3.34 do erro de predição. Dessa forma, o processo de predição (filtragem) deve ser realizado em paralelo ao processo de adaptação, trabalhando-se com uma matriz $\mathbf{A}'(n)$ e um vetor $\underline{e}'_a(n)$, tal que:

$$\underline{e}'_a(n+1) = (\underline{Y}(n+1) - \mathbf{A}'^T(n) \underline{U}(n))/K \quad (3.74)$$

$$\mathbf{A}'(n+1) = \mathbf{A}'(n) + \underline{G}(n) \underline{e}'_a^T(n+1) \quad (3.75)$$

onde os vetores $\underline{a}_k(n)$ da equação 3.73 são as colunas da matriz $\mathbf{A}'(n)$.

3.5. SIMULAÇÕES E DISCUSSÕES

No que se refere à proposta apresentada neste capítulo, é fundamental confirmar o desenvolvimento teórico através de resultados de simulação, que serão agora analisados. Essas simulações foram feitas tomando-se o problema de predição de senóides na presença de ruído branco gaussiano aditivo.

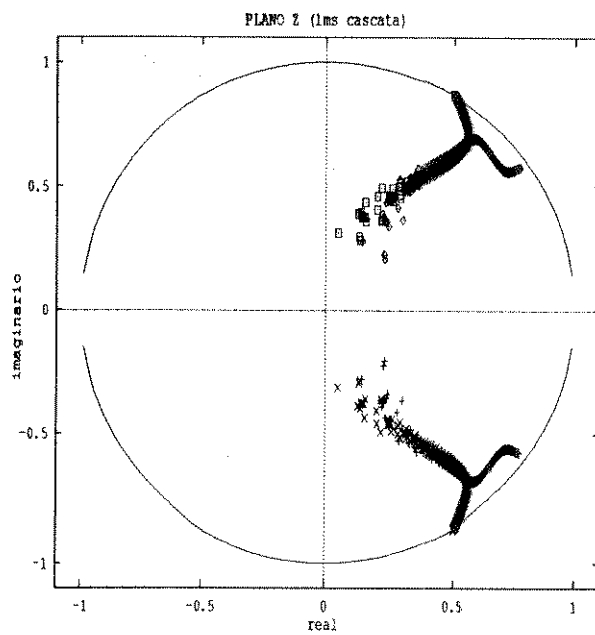
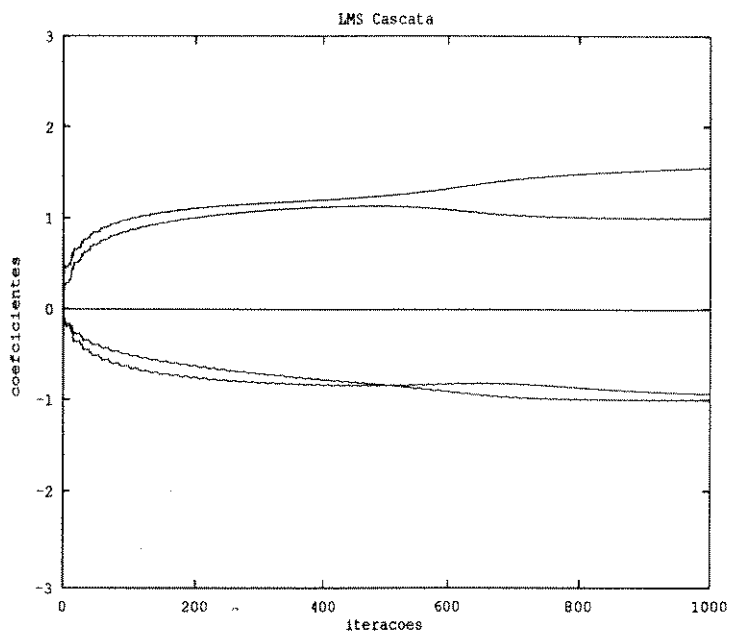
Primeiramente, toma-se a predição de duas senóides com frequências próximas: $f_1 = f_e/10$ e $f_2 = f_e/6$, onde f_e é a frequência de amostragem.

Na Figura 2.10.a, do capítulo anterior, foi mostrada a evolução dos coeficientes dos filtros preditores na estrutura direta, utilizando-se quatro coeficientes de tal forma a predizer as senóides, sendo mostrada na Figura 2.10.b a evolução dos zeros. Esses resultados já são bastante conhecidos, mostrando a melhoria da rapidez de convergência quando se usa algoritmos dos mínimos quadrados ao invés do gradiente, com uma complexidade computacional bastante aceitável, pelo fato de se usar o algoritmo rápido (FLS).

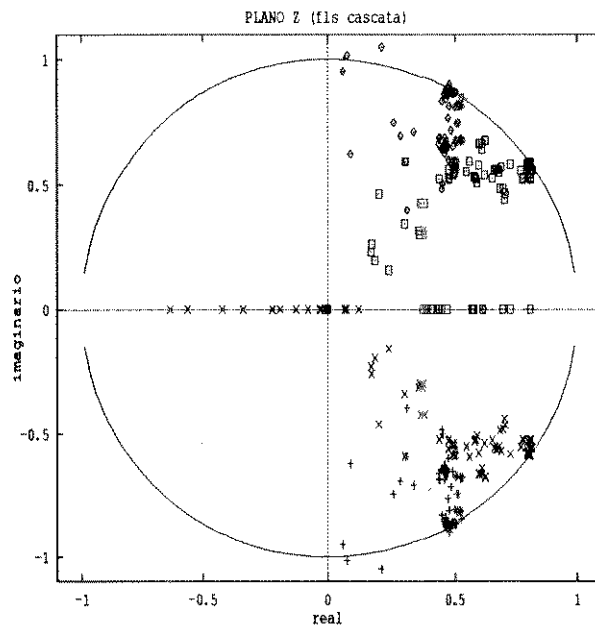
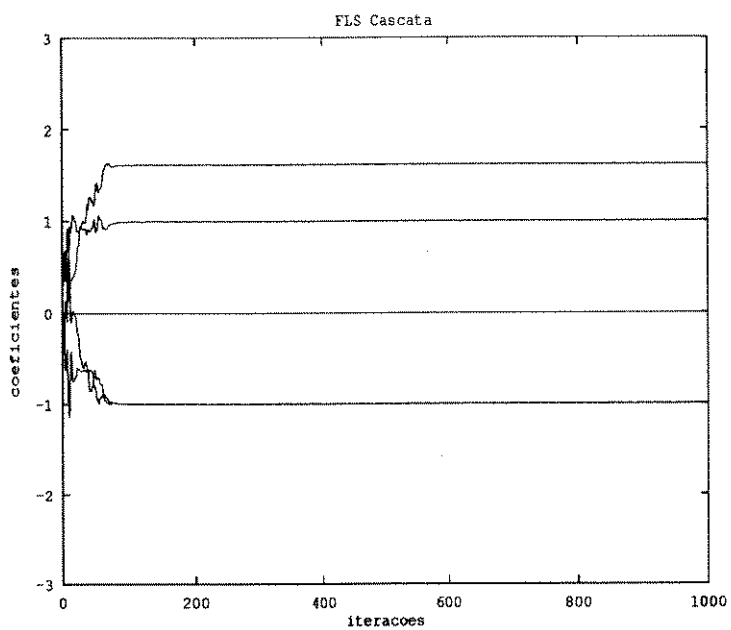
A Figura 3.8 mostra a evolução dos coeficientes e zeros, para as mesmas condições da Figura 2.10, utilizando-se a estrutura cascata proposta, onde cada secção contém dois coeficientes. No caso da Figura 3.8.a foi usado o algoritmo LMS e pode-se observar que a estrutura cascata conduz a uma velocidade de convergência bem maior que a da transversal. A Figura 3.8.b ilustra o desempenho do algoritmo FLS multidimensional, que mantém as boas propriedades de convergência do método de mínimos quadrados. Evidentemente, o ganho maior está em se obter diretamente os zeros do filtro e, a partir deles, as frequências das senóides de entrada. No caso transversal, seriam necessários complicados cálculos polinomiais a cada iteração.

A melhoria na rapidez de convergência quando se utiliza o algoritmo LMS na estrutura cascata com relação à estrutura convencional FIR se deve ao fato da alta sensibilidade ao espalhamento dos autovalores da matriz de correlação do sinal de entrada [1]. Se o espalhamento é muito grande, a convergência do algoritmo LMS é muito lenta. Na estrutura em cascata, a divisão do filtro em estágios de segunda ordem reduz o espalhamento na matriz de auto correlação dos sinais à entrada de cada estágio, melhorando assim a convergência.

Na Figura 3.9, toma-se duas senóides ainda mais próximas: $f_1 = f_e/10$ e $f_2 = f_e/8$. Para condições equivalentes, com $\mu = 0.05$ e $W = 0.95$ [1], a evolução dos coeficientes e dos zeros, evidenciam a diferença de desempenho entre os algoritmos LMS e FLS Multidimensional para a estrutura em cascata.



(a)



(b)

Fig.3.8 - Predição de Duas Senóides com Estrutura Cascata

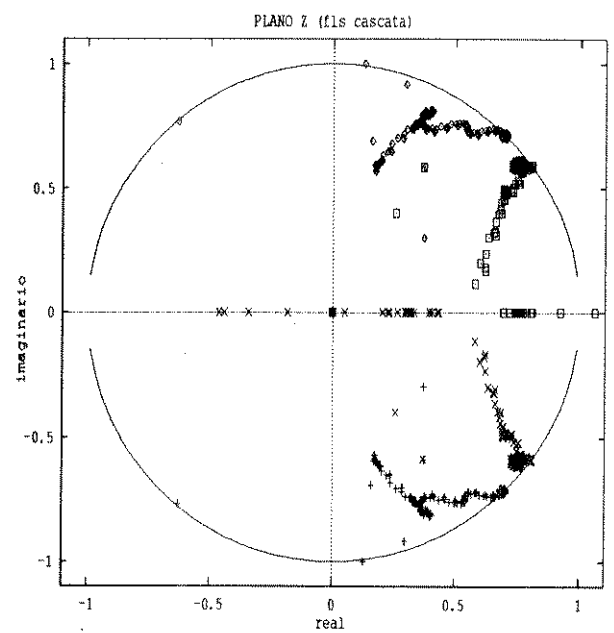
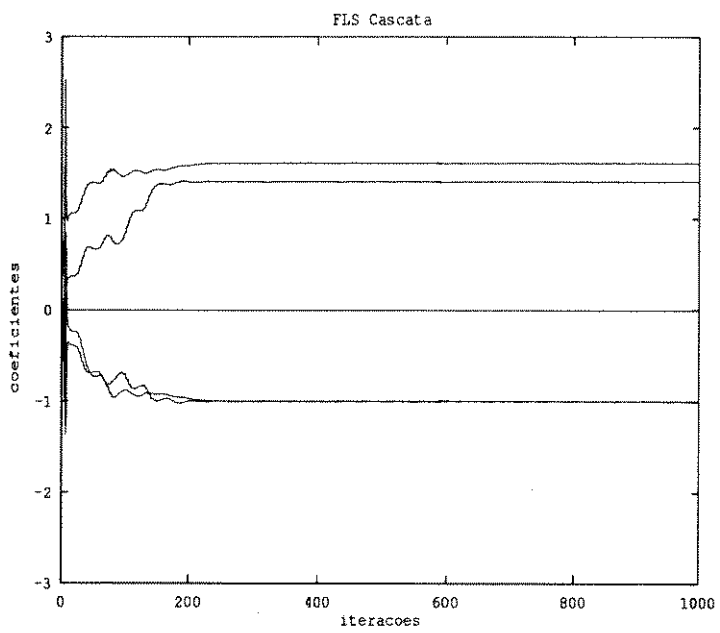
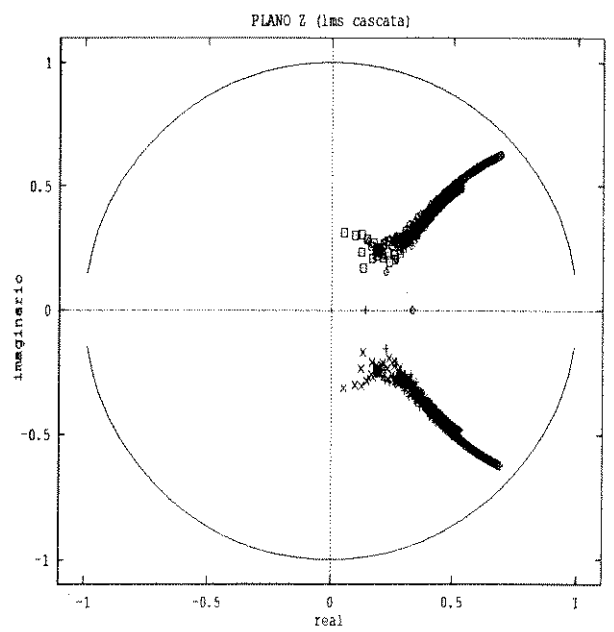
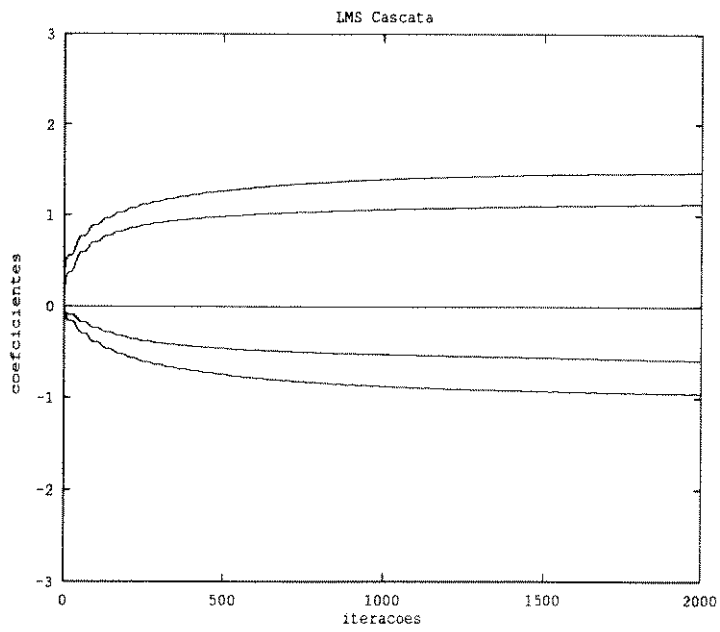


Fig.3.9 - Predição de Duas Senóides com Estrutura Cascata

As Figuras 3.10, 3.11, 3.12, 3.13 mostram o desempenho do preditor adaptativo com estrutura em cascata utilizando o algoritmo FLS Multidimensional para três, quatro, cinco e seis senóides, respectivamente. Novamente, vê-se o bom desempenho do filtro preditor proposto.

Vale ressaltar que o algoritmo LMS, embora com uma convergência mais lenta que o FLS-MD, se mostra mais rápido que na forma direta. Além disso, é extremamente estável, apesar das aproximações discutidas anteriormente, sendo que seu desempenho depende apenas da escolha do μ apropriado.

Já a proposta dos mínimos quadrados multidimensional apresenta uma certa instabilidade no início do processo até que o algoritmo se ajuste, convergindo, a partir daí, rapidamente. Isso se deve ao fato de que, no início da adaptação, a variação dos coeficientes é significativa, contrariando a própria hipótese de base, feita para linearizar o problema. Essa instabilidade se torna sensível em casos críticos, de excesso de senóides ou de senóides extremamente próximas. Para sanar esses problemas, um passo γ é colocado na adaptação dos coeficientes $\underline{H}_c(n)$, de forma a se controlar o transiente. Obtém-se, assim, um algoritmo híbrido que utiliza o FLS-MD para a atualização do ganho de adaptação combinado com uma técnica do tipo Gauss-Newton para a atualização dos coeficientes.

3.6. ALGORITMO RÁPIDO MODIFICADO

A proposta apresentada neste capítulo para o filtro na estrutura cascata mostra uma matriz de intercorrelação $\mathbf{R}_{uu}(n)$ da forma da equação 3.21, onde o vetor $\underline{U}(n)$ é descrito pela equação 3.15. Propõe-se, então, que essa matriz seja uma aproximação da hessiana do algoritmo Gauss-Newton, sendo que a atualização de sua inversa, $\mathbf{R}_{uu}^{-1}(n)$, é feita através dos algoritmos RLS e FLS multidimensionais.

A atualização do vetor de coeficientes do filtro é dada, então, pela equação 3.47 ou pela equação (10) da Tabela 3.1 para o FLS multidimensional, na qual se insere o passo γ , como referido anteriormente, de acordo com a equação 3.54.

No caso da predição linear, deve-se inserir o passo γ na atualização da matriz $\mathbf{A}'(n)$ definida e bem discutida na seção 3.4.1.

Como o passo está inserido, apenas, na atualização do vetor de coeficientes, ele resolve o problema de instabilidade inicial dos algoritmos dos mínimos quadrados multidimensionais, sem interferir na rapidez de convergência dos mesmos.

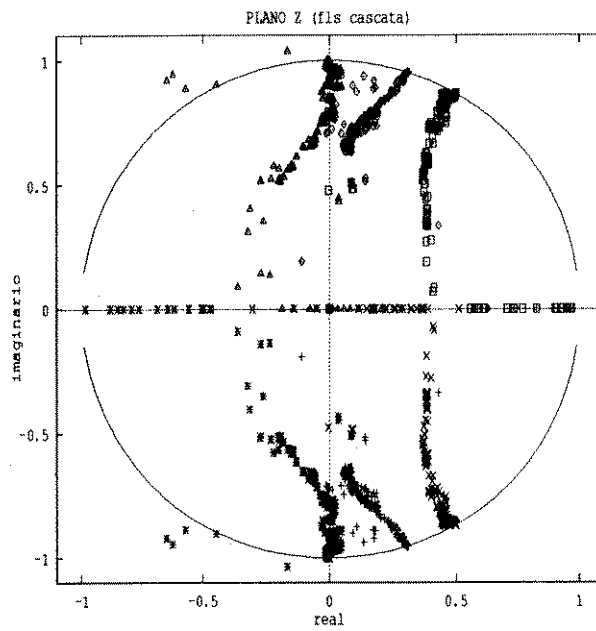
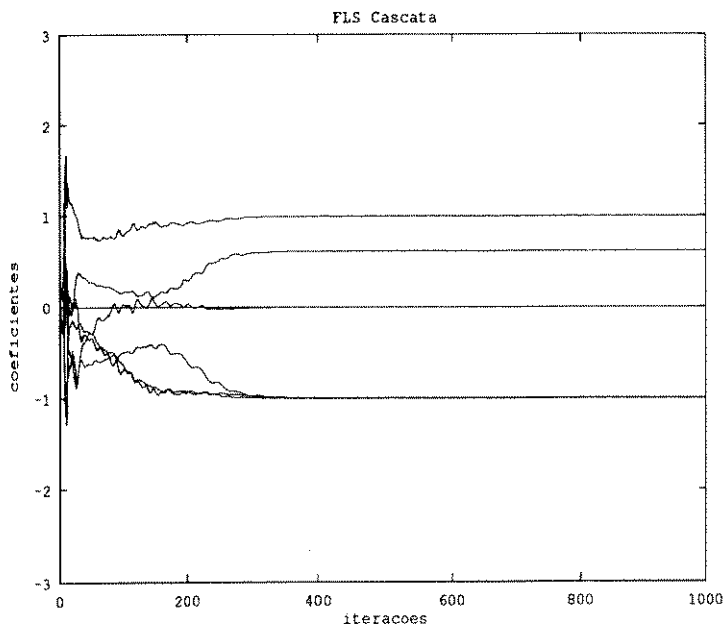


Fig.3.10 - Predição de Três Senóides com Estrutura Cascata

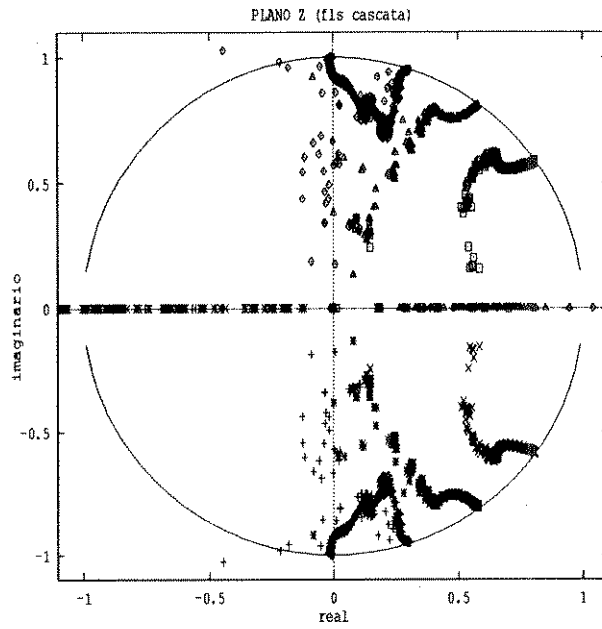
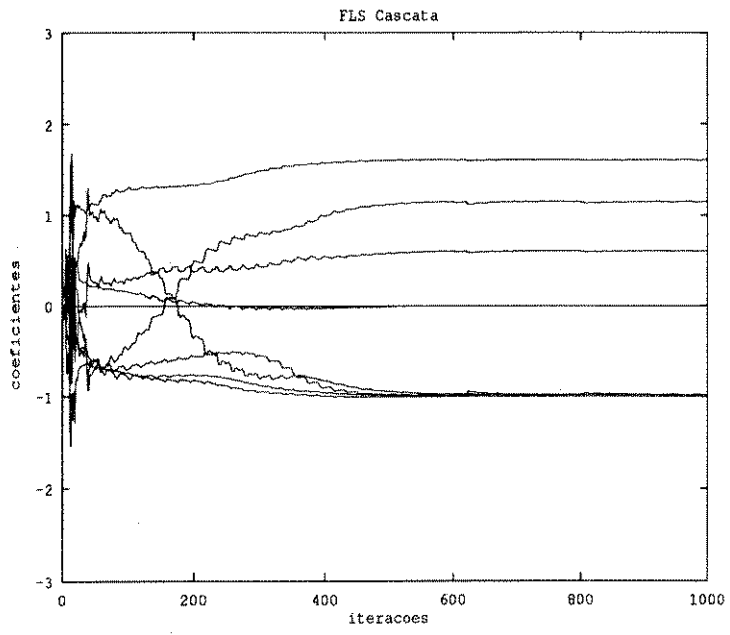


Fig.3.11 - Predição de Quatro Senóides com Estrutura Cascata

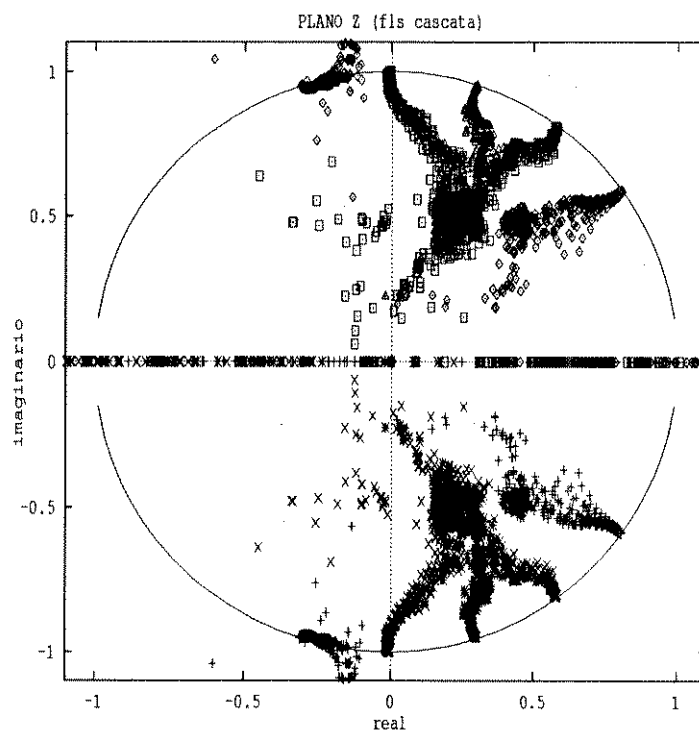
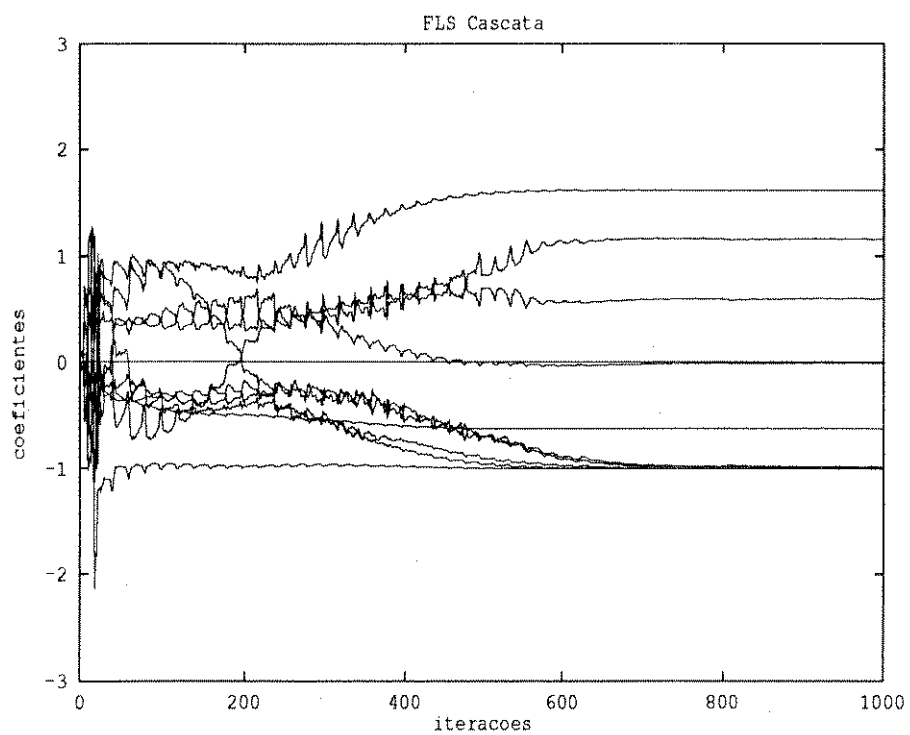


Fig.3.12 - Predição de Cinco Senóides com Estrutura Cascata

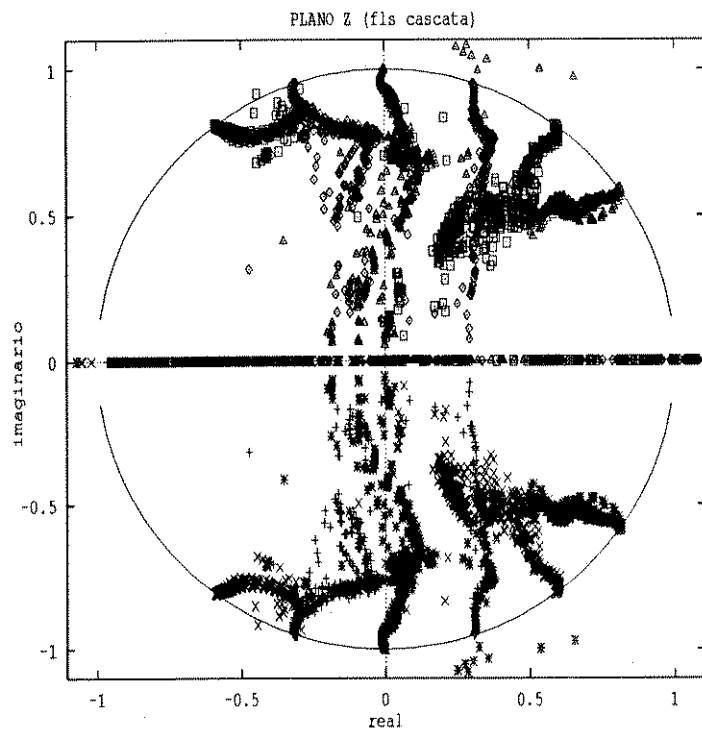
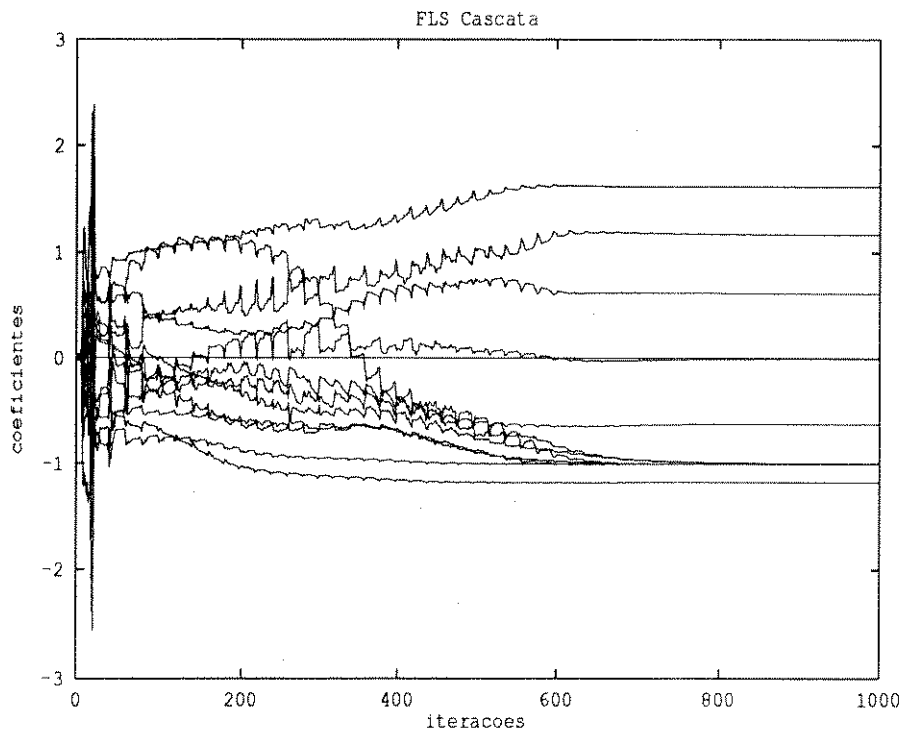


Fig.3.13 - Predição de Seis Senóides com Estrutura Cascata

A Figura 3.14 mostra a predição de sete senóides com frequências próximas, feita pelo algoritmo FLS-Multidimensional, como proposto anteriormente. A nova alternativa, com o algoritmo modificado, é mostrada na Figura 3.15, onde se utiliza o passo $\gamma=0.5$. Nota-se claramente que, embora tenha uma boa convergência (Figura 3.14.a), o algoritmo FLS-MD apresenta grande instabilidade no início do processo com muitos zeros fora do círculo unitário (Figura 3.14.b). Em contrapartida, a Figura 3.15 mostra o bom comportamento inicial quando se usa o algoritmo FLS-MD modificado. A inclusão do passo faz com que o algoritmo se mantenha estável, no início do processo, convergindo mais rapidamente para seus valores ótimos.

3.7. CONCLUSÕES

Neste capítulo, foi feita uma análise do comportamento de filtros adaptativos utilizando estruturas na forma em cascata de secções de segunda ordem, tendo como aplicação a predição de senóides na presença de ruído.

Os resultados se mostram bem superiores aos similares quando se usa os filtros convencionais FIR, com a vantagem de se obter os zeros diretamente dos coeficientes das secções de segunda ordem e, conseqüentemente, as frequências das senóides. Uma configuração multidimensional FIR é proposta ao invés da original, em cascata, a partir do desenvolvimento das equações do filtro ótimo.

A filtragem adaptativa LMS em cascata apresenta uma melhora considerável quando se compara com a estrutura direta convencional, sob as mesmas condições, tendo um comportamento bastante estável. Necessita-se, porém de condições iniciais diferentes para o vetor de coeficientes para não se cair em pontos de sela.

É proposta a abordagem do algoritmo dos mínimos quadrados rápidos multidimensional, FLS-MD, com uma rapidez de convergência muito superior ao LMS. Uma certa instabilidade dos coeficientes no início do processo, devido à aproximação feita para superar o problema da não linearidade, pode ser sanada usando-se um algoritmo modificado, a partir da junção do Gauss-Newton com o FLS-Multidimensional. Essa última proposta mostrou-se bastante eficiente e robusta, inclusive para os casos críticos de estimação das frequências de muitas senóides em meio a ruído.

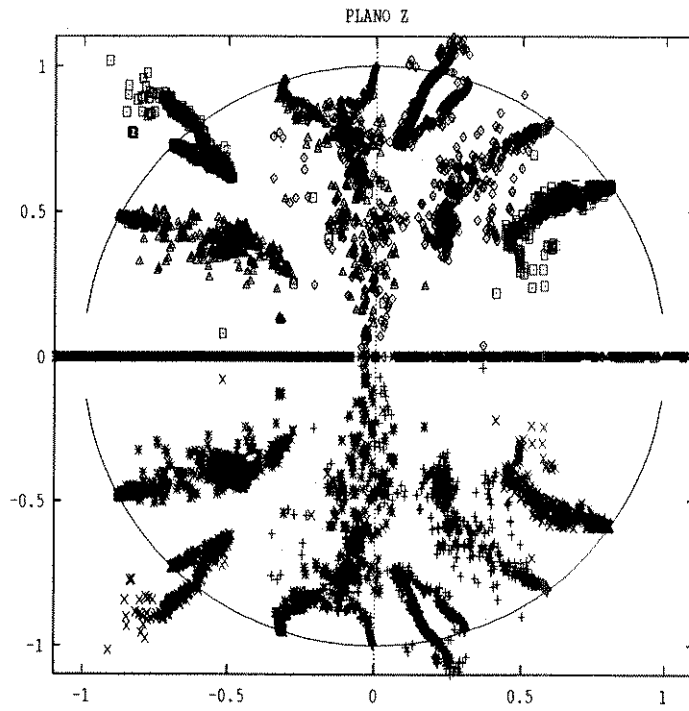
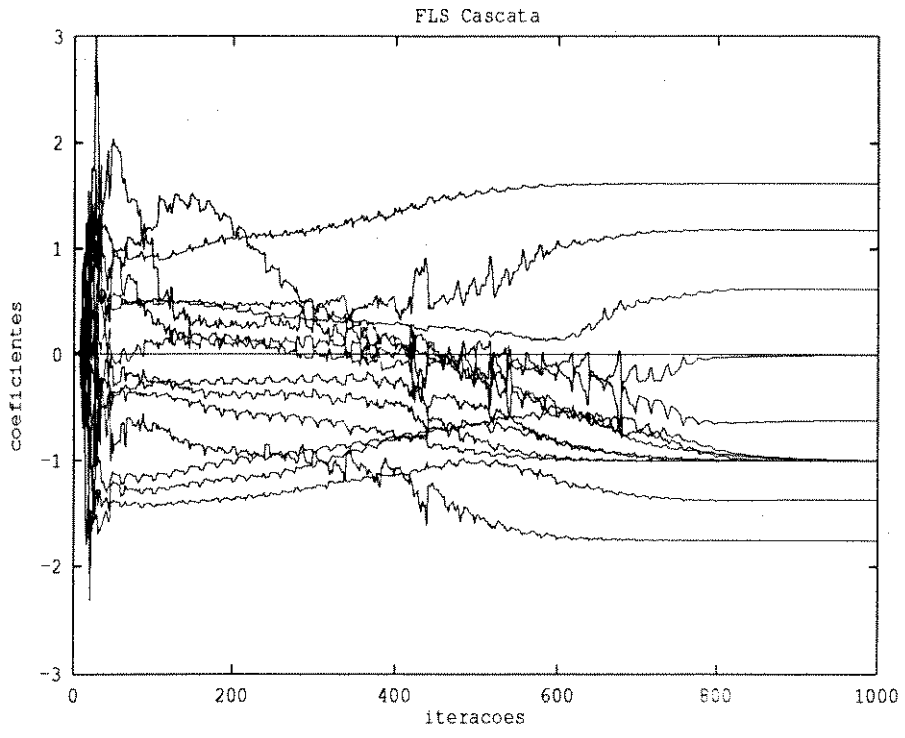


Fig.3.14 - Predição de Sete Senóides com Estrutura Cascata

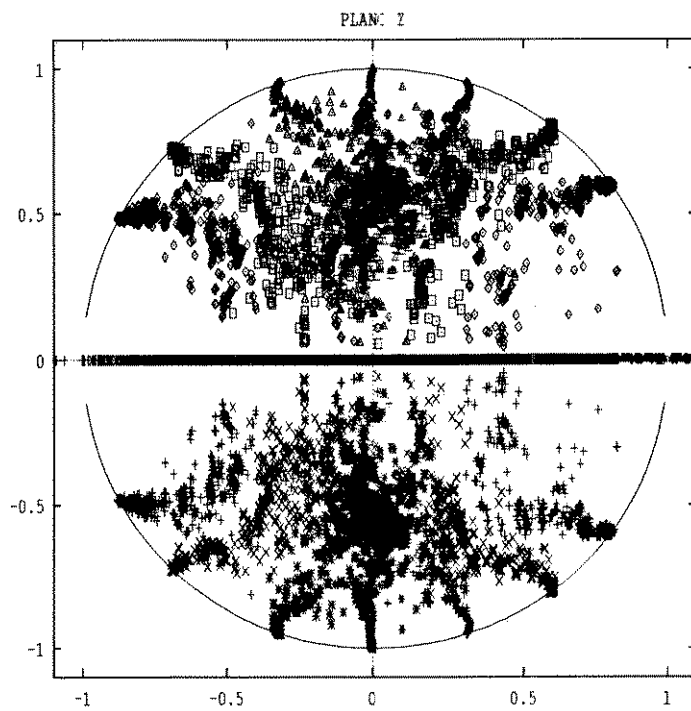
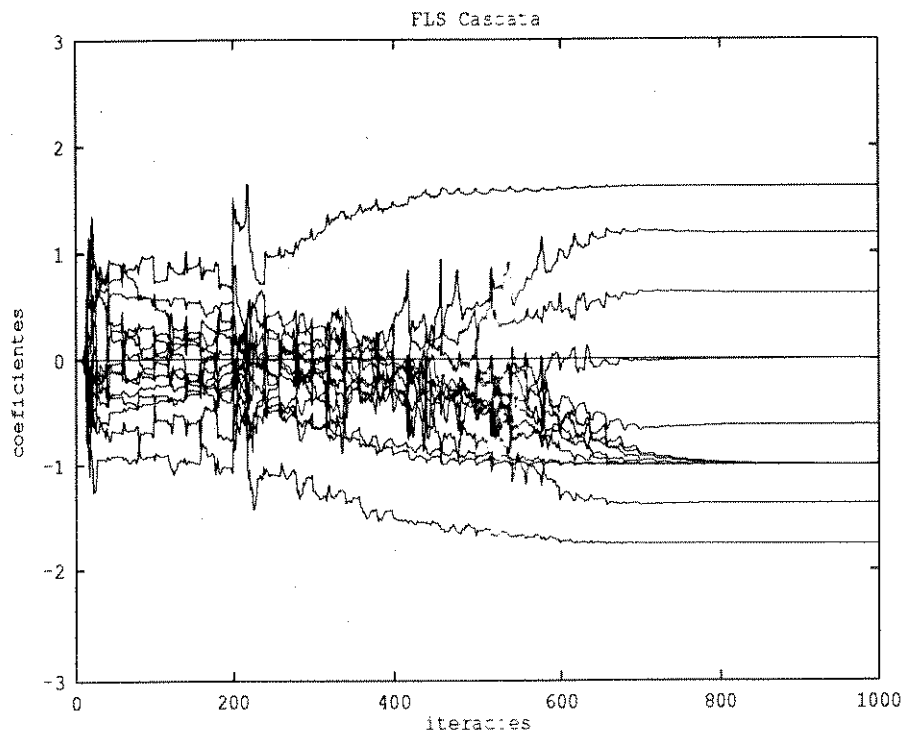


Fig.3.15 - Predição de Sete Senóides com Estrutura Cascata com o algoritmo modificado

CAPÍTULO 4

ANÁLISE DE CONVERGÊNCIA

A apresentação da estrutura em cascata como uma alternativa em filtragem adaptativa, leva à discussão da análise de convergência do processo de adaptação e do comportamento da curva de erro em função dos coeficientes dessa nova configuração.

No Capítulo 2, foram discutidos alguns problemas inerentes à filtragem adaptativa com filtros FIR, além de suas vantagens com relação a estruturas que não apresentam a curva de erro convexa. Os filtros FIR, na forma direta, são bem comportados, garantem estabilidade, porém exigem em muitos casos uma grande complexidade computacional. A necessidade de reduzir essa complexidade tem gerado muitos trabalhos nesta última década [6, 35, 36]. Esses trabalhos visam, em geral, estruturas IIR, apresentando a necessidade de controle de estabilidade dos filtros, bem como a dificuldade de convergência do processo de adaptação, além de não se garantir um mínimo global. Estruturas alternativas são apresentadas na literatura, com o objetivo de reduzir esses problemas [18, 37].

A proposta de filtragem na forma em cascata, apresentada neste trabalho, fornece um meio termo nos compromissos assumidos entre as estruturas FIR e IIR, como discutido anteriormente. Ou seja, a filtragem em cascata FIR garante algumas vantagens das estruturas não recursivas. Porém necessita de uma melhor avaliação da convergência do estimador e do comportamento da sua curva de erro, devido às aproximações feitas, da mesma forma que as estruturas IIR [5, 6].

Dessa forma, a análise de convergência do algoritmo proposto no capítulo anterior se faz necessária. Em tese, deve-se mostrar que o algoritmo proposto converge para um ponto de ótimo que corresponde ao mínimo global da estrutura FIR direta. Para tal, deve-se inserir a análise do filtro em cascata dentro de uma análise mais ampla de um algoritmo geral para estruturas arbitrárias. Essa discussão é, preliminarmente, proposta por Forssén [19], baseado em toda a formulação desenvolvida por Ljung [5], no que tange a convergência, além da análise da curva de erro, a partir de trabalhos desenvolvidos por Stearn, Soderstrom e Nayeri *et al.* [38, 39, 40].

Assim, a proposta do capítulo anterior, de filtragem em cascata, será

agora enquadrada no contexto das estruturas alternativas propostas por Forssén [19]. Mostra-se, nessa visão, que um algoritmo arbitrário converge para os pontos de mínimos locais da função custo [41], segundo a técnica da equação diferencial associada, ODE [5].

A seguir é mostrado que, para os filtros FIR em cascata, existe uma relação direta entre os seus pontos de mínimo com o ponto de mínimo global da estrutura direta FIR correspondente, de modo que o algoritmo proposto no capítulo anterior converge para mínimos globais, que correspondem ao mínimo global da estrutura direta FIR. Deve-se, entretanto, tomar precauções para não se cair em pontos de sela.

4.1. PROPOSTA DE ALGORITMO PARA ESTRUTURAS ARBITRÁRIAS

Trabalhos desenvolvidos por Ljung *et al.* [5, 27], levam a se definir um algoritmo geral, o qual tenta abordar as diversas formas de se tratar um problema de filtragem adaptativa (veja apêndice 4.A). Assim pode-se definir um algoritmo geral de adaptação, incluindo o processo de filtragem, levando-se em conta apenas as restrições do critério quadrático e da direção Gauss-Newton. A Tabela 4.1 mostra o algoritmo geral, conforme o desenvolvimento do apêndice 4.A.

$e(n) = y(n) - \hat{y}(n)$	(4.1.a)
$\mathbf{R}(n) = \mathbf{R}(n-1) + \gamma(n) [\underline{\eta}(n) \underline{\eta}(n)^T - \mathbf{R}(n-1)]$	(4.1.b)
$\underline{\mathbf{H}}_g(n) = \underline{\mathbf{H}}_g(n-1) + \gamma(n) \mathbf{R}^{-1}(n) \underline{\eta}(n) e(n)$	(4.1.c)
$\underline{\mathbf{V}}(n+1) = \mathbf{A}(\underline{\mathbf{H}}_g(n)) \underline{\mathbf{V}}(n) + \underline{\mathbf{b}}(\underline{\mathbf{H}}_g(n)) x(n)$	(4.1.d)
$\begin{bmatrix} \hat{y}(n+1) \\ \underline{\eta}(n+1) \end{bmatrix} = \mathbf{C}(\underline{\mathbf{H}}_g(n)) \underline{\mathbf{V}}(n+1)$	(4.1.e)

TABELA 4.1 - Algoritmo Geral

Vale ressaltar a caracterização de cada processo de adaptação, feita pela escolha do modelo, do sinal de entrada, do critério, da seqüência de ganho, além da direção de busca do algoritmo.

A direção de descida do algoritmo geral baseia-se na alteração da direção oposta à estimativa do gradiente pela multiplicação dessa pelo inverso da Hessiana. Qualquer matriz $R(n)$ pode ser usada, desde que positiva definida, de forma que o produto $R^{-1}(n)\underline{\eta}(n)$ forme um ângulo com $\underline{\eta}(n)$ menor que 90° . Assim, a definição do vetor gradiente $\underline{\eta}(n)$ é de fundamental importância para se determinar o algoritmo geral. A proposta de estruturas arbitrárias vem no sentido de se generalizar a obtenção do gradiente, para um determinado grupo de estruturas, para o qual se faz o mesmo tipo de aproximação.

4.1.1. A Escolha do Gradiente

A escolha do gradiente, no algoritmo geral da Tabela 4.1, depende de cada estrutura analisada. Forssén [19] propôs uma aproximação na obtenção do gradiente para o uso em estruturas arbitrárias. A chave do método é observar que se pode obter o gradiente de uma função de transferência de uma dada rede com relação aos seus coeficientes usando a formulação de sensibilidade de rede [8]. Essa visão é semelhante à proposta por Jackson [7], no caso em cascata, com a mesma base deste trabalho, ou seja, utiliza-se filtragens parciais do sinal de entrada para a obtenção dos parâmetros do gradiente. Assim, usa-se o fato de que, obtido o gradiente do erro com relação aos parâmetros do filtro, pode-se, diretamente, atualizar os parâmetros utilizando o algoritmo Gauss-Newton. Chega-se a um método razoavelmente simples, mesmo para algumas estruturas mais complexas.

4.1.1.1.- Apresentação do Método

A expressão para o gradiente $\underline{\eta}(n)$ será derivada observando-se que, no caso geral, é uma função dos coeficientes \underline{H}_g . No caso dos filtros FIR, não se tem esse problema, pois $\hat{y}(n)$ é linear com \underline{H}_g .

Primeiro, toma-se a definição de sensibilidade de rede, que é expressa utilizando o conceito de transposição [8], dada por:

$$\frac{\partial H_{a,b}}{\partial c_{i,j}} = H_{a,i} \cdot H_{j,b} \quad (4.2)$$

onde $H_{a,b}$ é a função de transferência entre os dois nós a e b , e $c_{i,j}$ o multiplicador entre os nós i e j . Tomando o filtro adaptativo como uma função de transferência variante no tempo, observa-se que seus coeficientes podem ser vis-

tos como os multiplicadores variantes no tempo entre dois nós diferentes.

O vetor gradiente consiste de um conjunto de derivadas parciais sob a forma:

$$\frac{\partial \hat{y}(n)}{\partial c_{i,j}} = H_{a,i} H_{j,b} x(n) \quad (4.3)$$

Considerando o sinal filtrado até o nó i como sendo $w_i(n)$, pode-se escrever

$$\frac{\partial \hat{y}(n)}{\partial c_{i,j}} = H_{j,b} w_i(n/\underline{H}_g) \quad (4.4)$$

onde \underline{H}_g corresponde ao respectivo conjunto de coeficientes, suposto conhecido. Tem-se ainda:

$$\frac{\partial \hat{y}(n)}{\partial c_{i,j}} \approx H_{j,b} w_i(n, \underline{H}_g) \quad (4.5)$$

onde essa igualdade aparece quando se supõe $H_{a,i}$ aproximadamente constante no tempo, ou seja, que as modificações sobre o vetor \underline{H}_g são suficientemente lentas. Essa é uma aproximação tipicamente usada, inclusive por Ljung *et al.* [5, pp. 29-30], e já comentada neste trabalho.

Como $w_i(n, \underline{H}_g)$ é um estado do filtro adaptativo, o único problema é encontrar $H_{j,b}$, a função de transferência do nó j para o nó b , a qual é difícil explicitar. Porém, a equação 4.5 pode ser visualizada graficamente. A Figura 4.1 mostra como obter a derivada do sinal estimado, e conseqüentemente do erro de estimação, em relação a um coeficiente arbitrário c .

Usando essa abordagem, não é necessário derivar expressões explícitas para as equações do gradiente. Pode-se ver cada função de transferência $H_{j,b}$ como um filtro gradiente. Basicamente, a equação 4.5 expressa que, para se obter a derivada com relação a um dos coeficientes, toma-se o sinal de entrada do nó do respectivo coeficiente e injeta-o como única entrada para um outro filtro com a mesma estrutura e os mesmos coeficientes do filtro original adaptativo, de forma que se tenha um filtro gradiente. O ponto de injeção desse sinal no filtro gradiente é o nó de saída do respectivo coeficiente, obtendo-se, assim, na saída desse filtro, a derivada da saída do filtro adaptativo com relação ao coeficiente especificado.

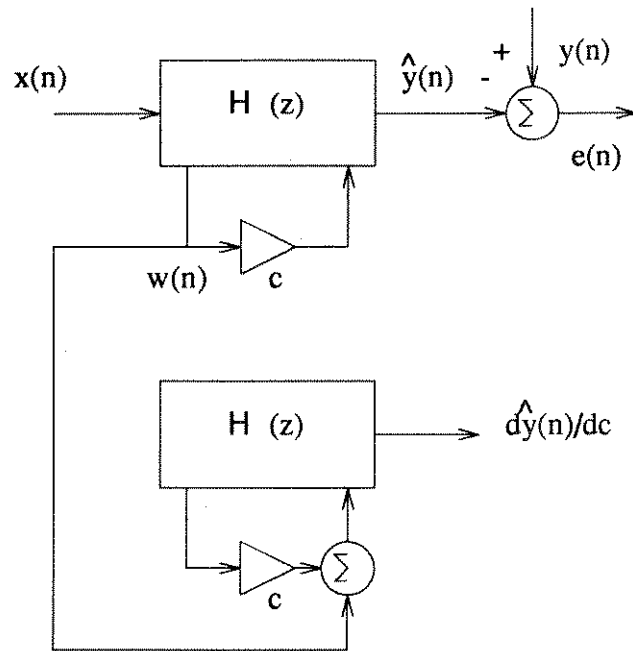


Fig. 4.1. - Cálculo da derivada do erro.

De uma maneira geral, é requerido um filtro gradiente para cada elemento do vetor $\underline{\eta}(n)$. Entretanto, isso somente é verdade para estruturas complexas. Para a maioria das redes, é possível simplificar esse trabalho removendo-se partes dos filtros gradientes que nunca serão excitadas e aproximando-se outras partes.

4.1.2. O algoritmo para Estrutura Arbitrária

Denotando-se cada filtro gradiente como sendo $G_1(z^{-1}), \dots, G_M(z^{-1})$ onde M é o número de coeficientes, tem-se que o vetor gradiente, como filtragem do sinal de entrada pode ser determinado por:

$$\underline{\psi}(n, \underline{H}_g) = \begin{bmatrix} G_1(z^{-1}) \\ \vdots \\ G_M(z^{-1}) \end{bmatrix} x(n) \quad (4.6)$$

Assim, o algoritmo Gauss-Newton para esse caso, *método do erro de estimação recursivo* (RPEM) [5] com estruturas arbitrárias é, agora, descrito pela Tabela

4.2, onde é suprimida a dependência com relação a \underline{H}_g da equação 4.6, de acordo com o desenvolvimento feito no apêndice 4.A.

$$\begin{aligned}
 e(n) &= y(n) - \hat{y}(n) & (4.7.a) \\
 \underline{H}_g(n) &= \underline{H}_g(n-1) + \gamma(n) \mathbf{R}^{-1}(n) \underline{\psi}(n) e(n) & (4.7.b) \\
 \mathbf{R}(n) &= \mathbf{R}(n) + \gamma(n) [\underline{\psi}(n) \underline{\psi}^T(n) - \mathbf{R}(n-1)] & (4.7.c) \\
 \begin{bmatrix} \hat{y}(n) \\ \underline{\psi}(n) \end{bmatrix} &= \begin{bmatrix} \mathbf{H}(z^{-1}) \\ \mathbf{G}_1(z^{-1}) \\ \vdots \\ \mathbf{G}_M(z^{-1}) \end{bmatrix} \cdot \mathbf{x}(n) & (4.7.d)
 \end{aligned}$$

TABELA 4.2 - Algoritmo Gauss-Newton para RPEM

4.2. O FILTRO EM CASCATA COMO UMA ESTRUTURA ARBITRÁRIA

Dentro da discussão feita no item anterior, deve-se verificar que o filtro em cascata, desenvolvido no capítulo anterior, se encaixa perfeitamente na proposta de um algoritmo geral para estruturas arbitrárias [19]. Tomando-se a formulação da estrutura multidimensional, vê-se que essa corresponde a uma estrutura de filtros gradientes, desenvolvida na seção 4.1.

Para se deixar mais claro, numa análise simples, pode-se relacionar, diretamente, as equações 3.5, 3.54 e 3.55 do Capítulo 3 com as equações 4.7 a, b, c da Tabela 4.2 do item anterior, que caracterizam o algoritmo do erro de predição. Além disso, tem-se a relação direta das equações 3.12 e 3.13 com a equação 4.7d da Tabela 4.2, caracterizando a obtenção do vetor gradiente, onde toma-se os filtros $G_i(z^{-1})$ ($i=1, \dots, M$), do item anterior, como sendo a função de transferência de cada conjunto de seções, menos a i -ésima seção, como definido na formação do vetor gradiente $\underline{U}(n)$, feita no capítulo anterior, na proposta da estrutura multidimensional.

Dessa forma, vê-se que a abordagem de uma estrutura arbitrária, feita por Forssén [19], é uma alternativa à proposta deste trabalho, desenvolvida no

Capítulo 3, da mesma forma que para outros casos de estruturas específicas apontadas na literatura [6, 18, 37].

Com isso, pode-se fazer a análise de convergência dos filtros em cascata, como consequência direta da análise assintótica proposta por Forssén em [41].

Nesse caso, deve-se mostrar que as estruturas alternativas formadas a partir do filtro gradiente mantém o comportamento assintótico do algoritmo geral proposto por Ljung [Apêndices 4.A, B, C].

4.3. ANÁLISE ASSINTÓTICA DE CONVERGÊNCIA PARA FILTROS COM ESTRUTURAS ARBITRÁRIAS

Deve-se enumerar algumas pré-condições de forma a se poder enunciar a proposição de convergência para estruturas arbitrárias, nos moldes dos apêndices 4.B para o caso geral, e 4.C para os algoritmos de erro de predição.

Seja um filtro adaptativo definido no espaço de estados (apêndice 4.A):

$$\underline{w}(n+1) = \underline{A}(\underline{H}_g) \underline{w}(n) + \underline{b}(\underline{H}_g) x(n) \quad (4.8.a)$$

$$\hat{y}(n) = \underline{c}^T(\underline{H}_g) \underline{w}(n) + d(\underline{H}_g) x(n) \quad (4.8.b)$$

sendo a função de transferência da realização entrada/saída da forma:

$$H(z^{-1}) = z^{-1} \underline{c}^T [\underline{I} - z^{-1} \underline{A}]^{-1} \underline{b} + d \quad (4.9)$$

onde \underline{A} , \underline{b} , \underline{c} e d não necessariamente são os mesmos do algoritmo geral.

Assim, relaciona-se as condições:

- *Sobre a estrutura:*

cond 1 - A matriz $\underline{A}(\underline{H}_g)$ tem todos os autovalores dentro do círculo unitário, ou seja, o filtro é estável.

cond 2 - Os elementos de \underline{A} , \underline{b} , \underline{c} e d são diferenciáveis à segunda, com relação a \underline{H}_g .

- Sobre a atualização:

cond 3 - O ganho de atualização tende a zero na forma $1/n$, ou seja:

$$\lim_{n \rightarrow \infty} n \cdot \gamma(n) = \text{cte}$$

cond 4 - $\mathbf{R}(n)$ é positiva definida usando-se, se necessário, métodos que garantam essa condição [1].

- Sobre os sinais de entrada e desejado

hip. 1 - Os seguintes limites existem $\forall \underline{H}_g \in \text{DM}$, definido no apêndice 4.B.

$$\bullet \bar{\mathbf{E}} \left\{ \frac{1}{2} e^2(n/\underline{H}_g) \right\} \triangleq \bar{\mathbf{J}}(\underline{H}_g)$$

$$\bullet \bar{\mathbf{E}} \left\{ \underline{\psi}(n/\underline{H}_g) e(n/\underline{H}_g) \right\} \triangleq \bar{\mathbf{f}}(\underline{H}_g)$$

$$\bullet \bar{\mathbf{E}} \left\{ \underline{\psi}(n/\underline{H}_g) \underline{\psi}^T(n/\underline{H}_g) \right\} \triangleq \bar{\mathbf{G}}(\underline{H}_g)$$

com $\bar{\mathbf{E}}$ definido em 4.C.3,

hip. 2 - É possível se escrever cada uma das seqüências $\{x(n)\}$ e $\{y(n)\}$ como soma de ruídos brancos filtrados através de filtros lineares estáveis e um número de senóides determinísticas.

hip. 3 - A parte de ruído da seqüência de entrada $\{u(n)\}$ deve ter energia em todas as frequências, ou seja, $\Phi_x(w) > 0 \quad \forall w$, o que força a existência de pelo menos um mínimo local.

A partir das condições expostas, é proposto o teorema sobre a convergência assintótica do algoritmo geral para estruturas arbitrárias.

Teorema 4.1 - Se o algoritmo satisfaz as condições de 1 a 4 e os dados às hipóteses de 1 a 3, então a seqüência $\{\underline{H}_g(n)\}$ convergirá com probabilidade 1 para o ponto onde $\nabla \underline{H}_g(\bar{\mathbf{J}}(\underline{H}_g)) = 0$ quando $n \rightarrow \infty$. Além disso, se as seqüências $\{x(n)\}$ e $\{y(n)\}$ são conjuntamente estacionárias no sentido amplo, $\{\underline{H}_g(n)\}$ convergirá para um mínimo local da função custo, para quase todas as condições iniciais.

A prova desse teorema consiste em três partes. Primeiro, aplicar ao algoritmo o teorema 4.B.1, que prova que os parâmetros $\underline{H}_g(n)$ convergem para um ponto onde $\nabla \underline{H}_g(\bar{J}(\underline{H}_g)) = 0$, ou para a fronteira do domínio permitido dos \underline{H}_g 's, ou seja δD_M . Segundo, provar que \underline{H}_g não converge para essa fronteira δD_M quando $\bar{J}(\underline{H}_g)$ tende para o infinito. Por último, que $\underline{H}_g(n)$ não se instalará em pontos estacionários instáveis (pontos de sela ou pontos de máximo). A prova completa se encontra em [41].

A restrição do teorema às condições iniciais leva em conta o fato que certas estruturas podem convergir para pontos de sela se as condições iniciais foram desfavoráveis. Esse problema foi apontado no caso do filtro em cascata com algoritmos baseados na aproximação estocástica quando se inicializa o vetor de coeficientes igualmente nulo pois, dessa forma, o algoritmo tenderá a buscar os mesmos coeficientes para seções diferentes. Deve-se, assim, escolher valores iniciais aleatórios para \underline{H}_g , de modo a anular a probabilidade de convergência para pontos de sela.

4.4. ANÁLISE DA CURVA DE ERRO

Como visto na seção anterior deste capítulo, a análise do comportamento da função custo dos filtros adaptativos é de grande importância, à medida que os algoritmos de adaptação só têm garantida sua convergência para os mínimos locais dessa função. Como a convergência para os mínimos locais, em geral, não garante a convergência para um mínimo global, seria altamente desejável que a função custo de um determinado filtro adaptativo tivesse somente mínimo global, o que nem sempre é possível.

A unicidade dos pontos de equilíbrio de um algoritmo adaptativo é necessária para um comportamento de convergência global. Desde que o objetivo da maior parte dos algoritmos é minimizar uma certa função custo, os pontos de equilíbrio do algoritmo coincidem com os pontos de mínimo da função custo.

A princípio, a característica de solução única é observada pelos métodos de aproximação estocástica apenas quando filtros FIR, transversais ou em treliça, são utilizados. Nesses casos, a curva de erro é convexa. Contudo, a unicidade pode não ser mantida para filtros adaptativos IIR quando se minimiza o erro médio quadrático de saída, ou seja, a superfície do erro pode não ser unimodal. A busca de condições que garantam certos comportamentos da curva de erro dos filtros adaptativos IIR na forma direta tem sido objeto de estudos nos últimos anos. Essa discussão é feita nas referências [38, 39, 40].

Para filtros com estruturas arbitrárias, o comportamento da curva de erro é, a princípio, desconhecido. Uma forma de análise dessas estruturas é feita através da relação entre suas realizações e a realização correspondente na forma direta. Ou seja, busca-se, aqui, relacionar curvas de erro de estruturas alternativas com as das estruturas na forma direta.

4.4.1. A Função Custo para Filtros com Estruturas Arbitrárias

Nesta seção, serão mostrados os pontos básicos da relação entre a forma direta e a correspondente forma alternativa de filtros adaptativos, no que diz respeito à curva de erro. Os recentes trabalhos desenvolvidos por Nayeri e Jenkins [18] e Forssén [19,41] inserem, basicamente, as seguintes afirmativas:

- (i) A função que relaciona os parâmetros de uma estrutura na forma direta àqueles que os correspondem na estrutura alternativa, também relaciona suas curvas de erro, tanto para os mínimos globais e locais, quanto para os pontos de máximo;
- (ii) Os pontos de sela da função custo das estruturas arbitrárias podem não ter correspondentes nas estruturas diretas, ou vice-versa;
- (iii) Nos casos das estruturas em cascata e paralela, múltiplos mínimos podem ocorrer devido à permutação entre as secções, o que implicará também no aparecimento de pelo menos um ponto de sela comum às regiões formadas pelas múltiplas secções.

A análise é formulada a partir da equivalência entre as realizações dos filtros adaptativos. Seja, então, a equação de atualização dos coeficientes a partir do algoritmo geral, dada por:

$$\underline{H}_g(n) = \underline{H}_g(n-1) + \gamma \underline{F}(n) \quad (4.10)$$

onde $\underline{H}_g(n)$ é o vetor de parâmetros ajustáveis, $\underline{F}(n)$ é o vetor que determina a direção de atualização de $\underline{H}_g(n)$ e γ é o passo de adaptação. Se a equação 4.10 converge para um ponto \underline{H}_g^* , então o vetor $\underline{F}(n)$ é identicamente nulo para \underline{H}_g^* . Assim, os pontos estacionários \underline{H}_g^* podem ser encontrados pela solução da equação:

$$\underline{F}(n) \Big|_{\underline{H}_g} = 0 \quad (4.11)$$

embora não sejam necessariamente únicos. Dessa forma, a igualdade em (4.11) determina os pontos estacionários da superfície de erro relacionada a uma estrutura geral. Deve-se analisar i) o que acontece com a superfície de erro MSE quando outras realizações são usadas, ii) se os novos pontos estacionários são criados com a transformação, e iii) se os novos pontos estacionários são estáveis (mínimos) ou instáveis (pontos de sela, máximo). O desempenho do algoritmo depende dessas questões.

Deve-se tomar os parâmetros ajustáveis da forma direta como sendo o vetor $\underline{H}_d(n) = [h_{d_1}, \dots, h_{d_M}]$. A realização de uma estrutura alternativa equivalente ao mesmo sistema tem $\underline{H}_a(n) = [h_{a_1}, \dots, h_{a_M}]$ como vetor parâmetro. Com isso, a relação entre as estruturas é feita pelas seguintes definições:

DEF.1) - Cada realização pode ser considerada como um mapeamento contínuo entre dois espaços \mathcal{D} e \mathcal{A} :

$$g: \mathcal{D} \rightarrow \mathcal{A}$$

$$g(\underline{H}_d) \triangleq \underline{H}_a \quad (4.12)$$

onde $\mathcal{D} \subset S_d$ e $\mathcal{A} \subset S_a$, sendo que S_d e S_a são tais que garantem estabilidade para \underline{H}_d e \underline{H}_a ,

DEF.2) - Sejam $f: \mathcal{D} \rightarrow \mathbb{R}$ e $h: \mathcal{A} \rightarrow \mathbb{R}$ como funcionais do MSE definidas em \mathcal{D} e \mathcal{A} , respectivamente. Então, uma realização g é dita realização equivalente se $\forall \underline{H}_d \in \mathcal{D}$:

$$f(\underline{H}_d) = h(g(\underline{H}_d)) = h(\underline{H}_a) \quad (4.13)$$

É importante conhecer como se altera a natureza das superfícies de erro na forma direta quando se faz a transformação g . Se g é diferenciável, pode-se usar a regra da cadeia:

$$\frac{\partial f(\underline{H}_d)}{\partial \underline{H}_d} = \frac{\partial h(\underline{H}_a)}{\partial \underline{H}_d} = \frac{\partial h(\underline{H}_a)}{\partial \underline{H}_a} \cdot \frac{\partial \underline{H}_a}{\partial \underline{H}_d} \quad (4.14)$$

onde $\partial f / \partial \underline{H}_d$ e $\partial h / \partial \underline{H}_a$ são vetores ($M \times 1$) e $\partial \underline{H}_a / \partial \underline{H}_d$ é uma matriz jacobiana ($M \times M$). Dessa forma, se \underline{H}_a^* é ponto estacionário de $h(\underline{H}_a)$, \underline{H}_d^* é um ponto estacionário

de $f(\underline{H}_d)$, tal que $g(\underline{H}_d^*) = \underline{H}_a^*$. Isto é:

$$\left. \frac{\partial h}{\partial \underline{H}_a} \right|_{\underline{H}_a^*} = 0 \quad \Rightarrow \quad \left. \frac{\partial f}{\partial \underline{H}_d} \right|_{\underline{H}_d^*} = 0 \quad (4.15)$$

É importante frisar que \underline{H}_d^* existe desde que g é um mapeamento sobre. Além disso, como:

$$\nabla_{\underline{H}_d}^2 f(\underline{H}_d) \Big|_{\underline{H}_d^*} = \left(\frac{\partial \underline{H}_a}{\partial \underline{H}_d} \right)^H \Big|_{\underline{H}_d^*} \cdot \nabla_{\underline{H}_a}^2 h(\underline{H}_a) \Big|_{\underline{H}_a^*} \cdot \left(\frac{\partial \underline{H}_a}{\partial \underline{H}_d} \right) \Big|_{\underline{H}_d^*}, \quad (4.16)$$

tem-se que a natureza dos pontos estacionários \underline{H}_d^* é similar à de \underline{H}_a^* .

Pode-se, alternativamente, analisar a suposição de que exista algum $\underline{H}_a^+ \in \mathcal{A}$ tal que $\left. \partial h / \partial \underline{H}_a \right|_{\underline{H}_a^+} = 0$ e $\left. \partial f / \partial \underline{H}_d \right|_{\underline{H}_d^+} \neq 0$ quando $\underline{H}_a^+ = g(\underline{H}_d^+)$, o que caracteriza a situação onde um ponto estacionário é formado para \underline{H}_a^+ mas não é formado para \underline{H}_d^+ . Assim, a equação 4.14 não é aplicável, isto é, g deve ser não diferenciável em \underline{H}_d^+ . Nesse sentido, Nayeri *et al.* [18] introduzem os seguintes lemas:

lema 1 - A função g não é diferenciável para alguns pontos estacionários recentemente formados sobre $h(\underline{H}_a)$.

lema 2 - Todos esses pontos estacionários recentemente formados são, então, pontos de sela.

Prova do lema 2:

Seja $B(\underline{H}_d^+, r)$ uma bola aberta, centrada em \underline{H}_d^+ e de raio $r > 0$, na qual se aplica a transformação g . Como g é sobre, a imagem de B contém uma bola aberta $B'(\underline{H}_a^+, r')$ com $r' > 0$.

Como \underline{H}_d^+ não é mínimo (nem máximo), existe uma direção \underline{d} onde:

$$f(\underline{H}_d^+ + \underline{d}) < f(\underline{H}_d^+) \quad (4.17)$$

Desde que

$$(\underline{H}_d^+ + \underline{d}) \in B \Rightarrow g(\underline{H}_d^+ + \underline{d}) \in B' \quad (4.18)$$

e, pela propriedade de equivalência:

$$h(\underline{g}(\underline{H}_d^+ + \underline{d})) < h(\underline{H}_a^+) \quad (4.19)$$

Logo, \underline{H}_a^+ não é mínimo (nem máximo) em h . Porém, como se parte da hipótese que

$$\left. \frac{\partial h}{\partial \underline{H}_a} \right|_{\underline{H}_a^+} = \underline{0} \quad (4.20)$$

conclui-se que \underline{H}_a^+ é ponto de sela. □

4.5. APLICAÇÃO À FORMA CASCATA

Considerando a relação entre a forma direta dos filtros FIR e a forma em cascata, tem-se uma realização de equivalência forte, já que as funções de transferência definidas na equação 3.1 são iguais pontualmente, ou seja:

$$H_D(\underline{H}_d) = H_C(\underline{H}_c) \quad (4.21)$$

Dessa forma a transformação (Mx1), sobre os espaços de parâmetros, definida pela função:

$$\underline{F}: \text{Casc} \rightarrow \text{Diret} \quad (4.22)$$

é um mapeamento contínuo e sobre, já que, pela propriedade de fatoração, todos os pontos da forma direta possuem correspondentes na forma cascata. Ainda, tem-se a relação de equivalência entre as suas funções custo, ou seja:

$$J_C(\underline{H}_c) = J_D(\underline{F}(\underline{H}_c)) \quad (4.23)$$

Assim, o seguinte teorema pode ser enunciado:

TEOREMA 4.2 - Para todos os mínimos da função custo de uma estrutura alternativa em cascata FIR, existirá um mínimo com mesmo valor de função custo para um correspondente filtro na forma direta FIR.

Prova do Teorema 2:

Sejam:

$H_D \rightarrow$ filtro transversal FIR com parâmetros \underline{H}_d

$H_C \rightarrow$ filtro em cascata FIR com parâmetros \underline{H}_c

e as relações de equivalência:

$$(1) \quad H_C(\underline{H}_c) = H_D(\underline{H}_d) = H_D(F(\underline{H}_c)) \quad (4.24)$$

$$(2) \quad J_C(\underline{H}_c) = J_D(\underline{H}_d) = J_D(F(\underline{H}_c)) \quad (4.25)$$

onde a transformação $F : \underline{H}_c \rightarrow \underline{H}_d$ é representada por:

$$\underline{H}_d = F(\underline{H}_c) \quad (4.26)$$

Sabe-se que:

- (i) $J_D(\underline{H}_d)$ é convexa
- (ii) F é sobre, pois existe sempre \underline{H}_d tal que: $F(\underline{H}_c) = \underline{H}_d$,

pois todo filtro direto pode ser associado à uma fatoração em cascata, com $(M/2)!$ associações diferentes, conforme se ordena as secções de segunda ordem.

Seja agora \underline{H}_c° um ponto não-extremo de J_C , ou seja, $\exists \underline{H}_c^-$ e \underline{H}_c^+ tal que

$$J_C(\underline{H}_c^-) < J_C(\underline{H}_c^\circ) < J_C(\underline{H}_c^+) \quad (4.27)$$

assim, pela transformação F (contínua e sobre),

$$J_D(\underline{H}_d^-) < J_D(\underline{H}_d^\circ) < J_D(\underline{H}_d^+) \quad (4.28)$$

ou seja, $\underline{H}_d^\circ = F(\underline{H}_c^\circ)$ não é ponto extremo da curva J_D . Seja também

$$\left. \frac{\partial J_C(\underline{H}_c)}{\partial \underline{H}_c} \right|_{\underline{H}_c^\circ} = \underline{0} \quad (4.29)$$

o que significa que \underline{H}_c° é ponto estacionário, porém não é ponto extremo. Como

$$\left. \frac{\partial J_D(\underline{H}_d)}{\partial \underline{H}_d} \right|_{\underline{H}_d^\circ} \neq 0 \quad (4.30)$$

pois J_D é convexa, logo \underline{H}_c° é ponto de sela. Ainda, se existe \underline{H}_d^* tal que:

$$J_D(\underline{H}_d^*) < J_D(\underline{H}_d) \quad \forall \underline{H}_d \quad (4.31)$$

então \underline{H}_d^* é o ponto de mínimo em J_D . Como \underline{F} é sobre, existirá $\underline{H}_c^{(i)}$ (i - índice genérico limitado) tal que:

$$J_D(\underline{H}_d^*) = J_C(\underline{H}_c^{(i)}) \quad (4.32)$$

Além disso, tem-se de (4.31) que

$$J_D(\underline{F}(\underline{H}_c^{(i)})) < J_D(\underline{F}(\underline{H}_c)) \quad \forall \underline{H}_c \quad (4.33)$$

onde

$$\underline{H}_d^* = \underline{F}(\underline{H}_c^{(i)}) \quad (4.34)$$

Das duas últimas relações, conclui-se que $\underline{H}_c^{(i)}$ constituem pontos de mínimo da cascata, que esses mínimos são equivalentes entre si e, por sua vez, equivalentes ao mínimo da forma direta. \square

Como as curvas de erro médio quadrático das estruturas diretas FIR são convexas, ou seja, só têm um mínimo global, a correspondente estrutura em cascata apresentará, apenas, mínimos globais, segundo o Teorema 4.2. Logo, os demais pontos estacionários de $J_C(\underline{H}_c)$ são necessariamente pontos de sela, o que está de acordo com o lema 2 do item anterior.

Os pontos de sela podem ter muita influência na convergência. É mostrado que, se a matriz de covariância for singular, não se tem condição suficiente para a convergência para um mínimo [41]. A localização dos pontos de sela se dá de acordo com a transformação \underline{F} . Assim, tem-se o seguinte teorema sobre a localização dos pontos de sela na curva de erro dos filtros em cascata.

TEOREMA 4.3 - Os pontos de sela que são formados com o mapeamento, estão nos correspondentes múltiplos do subespaço onde dois ou mais zeros do filtro são iguais.

Prova: Os pontos estacionários formados são \underline{H}_c° tais que

$$\left. \frac{\partial J_C(\underline{H}_c)}{\partial \underline{H}_c} \right|_{\underline{H}_c^\circ} = \underline{0} \quad (4.35)$$

quando seu correspondente \underline{H}_d° tem

$$\left. \frac{\partial J_D(\underline{H}_d)}{\partial \underline{H}_d} \right|_{\underline{H}_c^\circ} \neq \underline{0} \quad (4.36)$$

com $\underline{H}_d^\circ = \underline{F}(\underline{H}_c^\circ)$.

Logo, a regra da cadeia,

$$\frac{\partial J_C}{\partial \underline{H}_c} = \frac{\partial J_D}{\partial \underline{H}_d} \cdot \frac{\partial \underline{H}_d}{\partial \underline{H}_c} \quad (4.37)$$

só será mantida nos casos de singularidade da matriz Jacobiana $\partial \underline{H}_d / \partial \underline{H}_c$. O Apêndice 4D (similar à referência [18]), mostra que

$$\det \left[\frac{\partial \underline{H}_d}{\partial \underline{H}_c} \right] = \prod_{\substack{i, j=1 \\ i < j}}^M (c_i - c_j) \quad i, j = 1, \dots, M \quad (4.38)$$

logo, o determinante será nulo quando dois ou mais zeros forem iguais, sendo c_k os coeficientes da função de transferência do filtro em cascata. \square

4.6. CONCLUSÕES

A análise efetuada neste capítulo teve como ponto de partida o fato de que a proposta do capítulo anterior, para a adaptação de estruturas do tipo FIR em cascata, se inclui na formulação de algoritmo geral introduzida por Ljung, em particular na versão dita RPEM (erro de estimação recursivo). Por sua vez, a estrutura multidimensional proposta pode ser vista como um conjunto de filtros gradientes em paralelo.

A partir dessa idéia, pode-se aplicar a análise assintótica, desenvolvida para o algoritmo geral, nesse caso de interesse. Essa análise, comumente utilizada para estudos de convergência em processamento adaptativo, tem como princípio associar uma equação diferencial ordinária (ODE) ao algoritmo em questão.

Entretanto, muitas referências não levam em conta que a técnica da ODE está fundamentalmente apoiada na hipótese de que os parâmetros variam lentamente, o que a torna apropriada para situações próximas do regime estacionário ou para passos de adaptação muito pequenos. Além disso, garante-se apenas a convergência para um ponto de equilíbrio estável que pode ser um mínimo local. A convergência para o ótimo dependerá, assim, das condições iniciais.

Torna-se então necessário complementar a análise da ODE pelo estudo das superfícies de erro quadrático médio. Nesse sentido, procurou-se neste capítulo particularizar os resultados de Forssén e Nayeri ao caso em cascata FIR. Com isto chegou-se no Teorema 4.2 ao importante resultado de que *a estrutura em cascata FIR possui, como pontos estáveis, apenas mínimos globais correspondentes ao ótimo único da versão transversal. Por sua vez a análise assintótica garante que o algoritmo proposto alcançará um desses pontos de mínimo.*

Finalmente, os pontos de sela representam a situação onde duas ou mais secções de 2ª ordem convergem para os mesmos valores. Isso se verifica em algumas simulações com algoritmo do tipo LMS, onde as condições iniciais não foram adequadas, mas não ocorre com a proposta do algoritmo rápido multidimensional, dada a diferenciação naturalmente introduzida nas condições iniciais.

Os resultados obtidos no capítulo abrem assim interessantes perspectivas para a aplicação da estrutura e do algoritmo em estudo nesta tese. Uma dessas aplicações é desenvolvida no capítulo seguinte. A análise da superfície de erro pode também ser aprofundada no sentido de se buscar, de forma mais segura, métodos viáveis para se modelar um sinal através dos coeficientes do filtro cascata (ou zeros do preditor), como opção à modelização auto-regressiva clássica.

O desenvolvimento aqui apresentado baseia-se num tratamento matemático que, embora não elementar, é conhecido na literatura. Os aspectos fundamentais desse tratamento estão dispostos nos apêndices a seguir.

APÊNDICE 4.A

O ALGORITMO GERAL DE ADAPTAÇÃO

Segundo Ljung [5], o processo de adaptação (identificação recursiva) pode ser tratado sob vários aspectos, porém os algoritmos resultantes exibem características muito similares. Das possíveis formas de abordar o problema de adaptação, a mais comum, em processamento de sinais digitais, é a aproximação estocástica. Essa abordagem é a que tem característica mais genérica, pois a sua derivação provém de aproximações a partir da curva de erro, independente da estrutura, via as técnicas de busca, levando-se em conta o desconhecimento das estatísticas dos sinais envolvidos.

Da mesma forma que nas seções 2.4 e 3.3, tomando-se $\underline{H}_g(n)$ como o vetor de todos os coeficientes de um filtro genérico, chega-se a sua atualização via o algoritmo do Gauss-Newton dentro das aproximações consideradas em [5], que serão revistas aqui.

4.A.1. Dedução do Algoritmo

Será considerado um sistema com uma entrada e uma saída (caso particular das equações gerais de Ljung) e também o critério quadrático. A partir do método de otimização Quase-Newton, deriva-se o algoritmo geral. Esse método de otimização recursivo de funções não lineares realiza a expansão quadrática em torno da estimativa corrente da solução ótima do problema, alterando a estimativa conforme o valor máximo ou mínimo da expansão.

Seja \underline{H}_{g_k} o vetor da solução ótima estimada do problema até a iteração k , $J(\underline{H}_g)$ a função critério, $\nabla[J(\underline{H}_g)]$ o gradiente em relação a \underline{H}_g e $\mathbf{H}(\underline{H}_g)$ o Hessiano de $J(\underline{H}_g)$ em relação a \underline{H}_g . A nova estimativa da solução ótima do problema na iteração $(k+1)$, utilizando o método Quase-Newton, como visto em 2.4, é dada por:

$$\underline{H}_{g_{k+1}} = \underline{H}_{g_k} - \gamma(n) \cdot \mathbf{H}(\underline{H}_g)^{-1} \cdot \nabla[J(\underline{H}_{g_k})]^T \quad (4.A.1)$$

onde $\gamma(n)$ é um passo qualquer de adaptação (ganho, geralmente menor que 1).

Para se chegar ao algoritmo geral, utiliza-se a eq.(4.A.1) com estimativas do Hessiano e do gradiente do critério obtidos a partir de dados observáveis do problema, fazendo-se a recursão no tempo. Este procedimento constitui o algoritmo do Gauss-Newton.

Como dito, anteriormente, o critério utilizado envolve o erro de predição da adaptação, dado por:

$$e(n, \underline{H}_g) = y(n) - \hat{y}(n/\underline{H}_g) \quad (4.A.2)$$

sendo $y(n)$ a referência e $\hat{y}(n/\underline{H}_g)$ a estimativa obtida na saída do filtro adaptativo. De uma maneira geral, o filtro pode ser descrito pela realização no espaço de estados, como função do vetor de observações $\underline{w}(n)$, que são resultados de certos tratamentos das medidas anteriores dos sinais disponíveis, dependendo do modelo. Assim, a seguinte estrutura para a geração do vetor $\underline{w}(n)$ e da estimativa $\hat{y}(n/\underline{H}_g)$ é definida:

$$\underline{w}(n+1, \underline{H}_g) = \underline{F}(\underline{H}_g)\underline{w}(n, \underline{H}_g) + \underline{f}(\underline{H}_g)x(n) \quad (4.A.3.a)$$

$$\hat{y}(n+1/\underline{H}_g) = \underline{h}^T(\underline{H}_g)\underline{w}(n+1, \underline{H}_g) \quad (4.A.3.b)$$

onde as matrizes \underline{F} , \underline{f} e \underline{h} são escolhidas apropriadamente. Por exemplo, na filtragem FIR, a equação 4.A.3 é definida diretamente da equação 2.17, que é a expressão da regressão linear. Também, para o caso proposto da filtragem em cascata, foi desenvolvida a expressão para a estimativa, como na equação 3.16.

Define-se o critério quadrático (função custo) da forma:

$$J(\underline{H}_g) = \frac{1}{2} E \left[e^2(n, \underline{H}_g) \right] \quad (4.A.4)$$

onde a busca da minimização desse critério é feita através do algoritmo recursivo Gauss-Newton.

Deve-se lembrar que a função custo não é disponível, de uma forma geral, necessitando-se trabalhar com estimativas, daí a aproximação estocástica. O método descrito é o do erro de estimação, para estruturas gerais, ao invés dos mínimos quadrados. Tomando-se o gradiente da função custo, em relação ao vetor de coeficientes \underline{H}_g , tem-se:

$$\underline{\nabla}_{\underline{H}_g} [J(\underline{H}_g)] = E \left[e(n, \underline{H}_g) \frac{\partial}{\partial \underline{H}_g} e(n, \underline{H}_g) \right] \quad (4.A.5)$$

mas

$$\frac{\partial}{\partial \underline{H}_g} e(n, \underline{H}_g) = - \frac{\partial}{\partial \underline{H}_g} \hat{y}(n/\underline{H}_g) \quad (4.A.6)$$

supondo-se que o sinal de referência, $y(n)$, não depende dos parâmetros do filtro. Definindo-se o vetor:

$$\underline{\eta}(n, \underline{H}_g)^T \triangleq \frac{\partial}{\partial \underline{H}_g} \hat{y}(n/\underline{H}_g) \quad (4.A.7)$$

escreve-se o gradiente da função custo como:

$$\underline{\nabla} [J(\underline{H}_g)] = -E \left[e(n, \underline{H}_g) \cdot \underline{\eta}(n, \underline{H}_g)^T \right] \quad (4.A.8)$$

Novamente, faz-se uso da aproximação estocástica, na qual se toma $-e(n, \theta) \cdot \underline{\eta}(n, \underline{H}_g)^T$ como estimativa de equação 4.A.8, de sorte que:

$$\underline{\nabla} [J(\underline{H}_g)] \cong -e(n, \underline{H}_g) \cdot \underline{\eta}(n, \underline{H}_g)^T \quad (4.A.9)$$

O Hessiano é expresso, como na equação 3.45, por:

$$\mathbf{H}(\underline{H}_g) = E \left[\underline{\eta}(n, \underline{H}_g) \cdot \underline{\eta}(n, \underline{H}_g)^T \right] + E \left[e(n, \underline{H}_g) \cdot \frac{\partial^2}{\partial \underline{H}_g^2} e(n, \underline{H}_g) \right] \quad (4.A.10)$$

É essencial, no problema de minimização, que a matriz Hessiana seja positiva-definida. Essa exigência vem do fato que, como a superfície da função custo, no caso geral, não é convexa, deve-se evitar uma tentativa equivocada de maximização, impondo-se garantias sobre a Hessiana. Além disso, mantém-se a não singularidade obrigatória para se trabalhar com sua inversa. Na equação 4.A.10, essa exigência é sempre atendida pelo primeiro termo da equação. Logo, é razoável a consideração feita nos capítulos anteriores de se desprezar o segundo termo da Hessiana. Aliado a isso, tem-se que, perto do ótimo, é suposto que $\{e(n, \underline{H}_g)\}$ seja uma seqüência de variáveis aleatórias independentes, com média nula. Ou seja, $e(n, \underline{H}_g)$ é independente inclusive de sua derivada segunda, levando-se a anular o segundo termo da equação 4.A.10.

A aproximação estocástica fornece a estimativa recursiva do Hessiano, segundo Robbin-Monro, da forma:

$$\mathbf{R}(n, \underline{H}_g) = \mathbf{R}(n-1, \underline{H}_g) + \gamma(n) \left[\underline{\eta}(n, \underline{H}_g) \underline{\eta}(n, \underline{H}_g)^T - \mathbf{R}(n-1, \underline{H}_g) \right] \quad (4.A.11)$$

com condição inicial \mathbf{R}_0 positiva-definida, e $\gamma(n)$ como passo, sem perdas de generalidade.

Analisando os procedimentos anteriores, nas equações 4.A.1, 4.A.2, 4.A.8, 4.A.9 e 4.A.10, vê-se que, para se obter o algoritmo geral, deve-se ter uma forma de determinação do vetor gradiente, já que, para o caso geral, sua expressão não é definida. Nos capítulos anteriores, foram vistos que: (i) no caso FIR transversal, devido à regressão linear, a obtenção do gradiente é direta; (ii) na proposta do filtro em cascata, a expressão do gradiente foi derivada a partir da filtragem por todas as secções exceto pela k -ésima secção. Assim, a forma específica do cálculo desse termo depende, exclusivamente, do modelo do filtro adaptativo utilizado. Assim sendo, na intenção de se chegar a um algoritmo geral, considera-se no desenvolvimento o vetor genérico $\underline{\eta}(n, \underline{H}_g)$. Para cada caso, esse vetor é calculado apropriadamente através da formulação por variáveis de estado, pela qual se obtém ainda o cálculo da estimativa do sinal, $\hat{y}(n)$. Inclui-se também no algoritmo, a recursão em termos de variáveis de estado dos elementos internos do filtro, como definido na equação 4.A.3. A estabilidade do filtro requer \underline{H}_g tal que $\mathbf{F}(\underline{H}_g)$ tenha todos os autovalores dentro do círculo unitário. Como o modelo é diferenciável com relação a \underline{H}_g , introduz-se o vetor:

$$\underline{V}(n, \underline{H}_g) = \left[\underline{w}(n, \underline{H}_g), \frac{d\underline{w}(n, \underline{H}_g)}{d(\underline{H}_g)}, \dots, \frac{d^d \underline{w}(n, \underline{H}_g)}{d\underline{H}_g^d} \right] \quad (4.A.12)$$

tal que o sistema ampliado com entrada $x(n)$ e saídas $\hat{y}(n/\underline{H}_g)$ e $\underline{\eta}(n, \underline{H}_g)$ seja dado pela equação de estado:

$$\underline{V}(n, \underline{H}_g) = \mathbf{A}(\underline{H}_g) \underline{V}(n-1, \underline{H}_g) + \underline{b}(\underline{H}_g) x(n-1) \quad (4.A.13.a)$$

$$\underline{V}(0, \underline{H}_g) = \underline{V}_0 \quad (4.A.13.b)$$

$$\begin{bmatrix} \hat{y}(n/\underline{H}_g) \\ \underline{\eta}(n, \underline{H}_g) \end{bmatrix} = \mathbf{C}(\underline{H}_g) \underline{V}(n, \underline{H}_g) \quad (4.A.13.c)$$

sendo $\mathbf{A}(\underline{H}_g)$ e $\mathbf{C}(\underline{H}_g)$ matrizes quaisquer e $\underline{b}(\underline{H}_g)$ um vetor qualquer, sempre dependentes do modelo adotado. Como o algoritmo geral fornece um novo vetor \underline{H}_g a cada iteração, a equação 4.A.13 é melhor escrita na forma:

$$\underline{V}(k+1, \underline{H}_g(n)) = \mathbf{A}(\underline{H}_g(n)) \underline{V}(k, \underline{H}_g(n)) + \underline{b}(\underline{H}_g(n)) x(n) \quad k=1, \dots, n \quad (4.A.14.a)$$

$$\underline{V}(0, \underline{H}_g(n)) = \underline{V}_o \quad (4.A.14.b)$$

$$\begin{bmatrix} \hat{y}(n+1/\underline{H}_g(n)) \\ \underline{\eta}(n+1, \underline{H}_g(n)) \end{bmatrix} = \underline{C}(\underline{H}_g(n)) \underline{V}(n+1, \underline{H}_g(n)) \quad (4.A.14.c)$$

Para tornar a equação prática, na recursividade, tem-se que:

$$\underline{V}(n, \underline{H}_g(n-1)) = [\underline{A}(\underline{H}_g(n-1))]^M \underline{V}_o + \sum_{k=0}^{n-1} [\underline{A}(k, \underline{H}_g(n-1))]^{n-k-1} \underline{b}(\underline{H}_g(n-1)) x(k) \quad (4.A.15)$$

Com $[\underline{A}(\underline{H}_g(n-1))]^M$ tendendo a zero na região de estabilidade de $\underline{H}_g(n-1)$, e fazendo-se outras aproximações [5], lembrando que $\gamma(n)$ é muito pequeno, chega-se a:

$$\underline{V}(n, \underline{H}_g(n-1)) \approx \underline{V}(n) \triangleq \sum_{k=0}^{n-1} \left[\prod_{m=k+1}^{n-1} \underline{A}(\underline{H}_g(m)) \right] \cdot \underline{b}(\underline{H}_g(k)) x(k) \quad (4.A.16)$$

Dessa forma, a equação 4.A.12 pode ser expressa por:

$$\underline{V}(n+1) = \underline{A}(\underline{H}_g(n)) \underline{V}(n) + \underline{b}(\underline{H}_g(n)) x(n) \quad (4.A.17)$$

e extendendo a outras variáveis a dependência apenas de n , a forma final do algoritmo, então, é apresentada pela Tabela 4.A.1, ressaltando-se que deve-se impor que $\underline{H}_g(n)$ esteja na região de estabilidade do filtro, além do fato que o algoritmo geral executa apenas buscas locais.

$e(n) = y(n) - \hat{y}(n)$	(4.A.18.a)
$\underline{R}(n) = \underline{R}(n-1) + \gamma(n) [\underline{\eta}(n) \underline{\eta}(n)^T - \underline{R}(n-1)]$	(4.A.18.b)
$\underline{H}_g(n) = \underline{H}_g(n-1) + \gamma(n) \underline{R}^{-1}(n) \underline{\eta}(n) e(n)$	(4.A.18.c)
$\underline{V}(n+1) = \underline{A}(\underline{H}_g(n)) \underline{V}(n) + \underline{b}(\underline{H}_g(n)) x(n)$	(4.A.18.d)
$\begin{bmatrix} \hat{y}(n+1) \\ \underline{\eta}(n+1) \end{bmatrix} = \underline{C}(\underline{H}_g(n)) \underline{V}(n+1)$	(4.A.18.e)

TABELA 4.A.1 - Algoritmo Geral

APÊNDICE 4.B

ANÁLISE ASSINTÓTICA DA CONVERGÊNCIA

Dentro da proposta de se fazer uma análise assintótica da convergência do estimador para os filtros adaptativos com estruturas alternativas, toma-se os trabalhos desenvolvidos por Ljung *et al.* [5, 27], fazendo-se uma revisão das discussões sob convergência a partir do algoritmo geral de adaptação. As propriedades assintóticas do estimador $\underline{H}_g(n)$ são colocadas quando n tende ao infinito e o ganho de adaptação $\gamma(n)$ tende a zero.

4.B.1.- O Método da Equação Diferencial Ordinária (ODE) Associada

O método de análise de convergência proposto por Ljung [27], o da equação diferencial ordinária associada (ODE), de uma maneira geral, busca amarrar as propriedades de convergência do algoritmo às propriedades de estabilidade de uma equação diferencial construída a partir da expressão de atualização do estimador e das suas estatísticas.

O método consiste em definir pré-condições ao algoritmo, de maneira que se possa equacionar a expressão de atualização dos elementos do algoritmo na forma de uma equação diferencial ordinária. A partir daí, são estudadas as propriedades de estabilidade dessa equação diferencial, definindo às condições de convergência do estimador.

A análise é feita tomando-se o algoritmo geral Gauss-Newton com uma função custo quadrática, definido pela equação 4.A.18.

Escrevendo a equação 4.A.18.d, de outra maneira, obtém-se:

$$\underline{V}(n) = \left[\prod_{k=0}^{n-1} A(\underline{H}_g(k)) \right] \underline{V}(0) + \sum_{j=0}^{n-1} \left[\prod_{k=j+1}^{n-1} A(\underline{H}_g(k)) \right] \underline{b}(\underline{H}_g(j)) x(j) \quad (4.B.1)$$

onde, sem perda de generalidade, pode-se considerar $\underline{V}(0) \equiv \underline{0}$, reduzindo a equação 4.B.1, tal que:

$$\underline{V}(n) = \sum_{j=0}^{n-1} \left[\prod_{k=j+1}^{n-1} A(\underline{H}_g(k)) \right] \underline{b}(\underline{H}_g(j)) x(j) \quad (4.B.2)$$

Supondo que uma estimativa do vetor de parâmetros $\underline{H}_g(k)$ pertença à vizinhança de um certo valor \overline{H}_g , para $n-m \leq k \leq n-1$, onde \overline{H}_g é tal que $A(\overline{H}_g)$ tenha todos seus autovalores dentro do círculo unitário. Então, para uma vizinhança suficientemente pequena, escreve-se:

$$\prod_{k=n-K}^{n-1} A(\underline{H}_g(k)) \approx A(\overline{H}_g)^K \quad (4.B.3)$$

Para valores grandes de K , considerando que os autovalores de $A(\overline{H}_g)$ têm módulo menor que 1, pode-se aproximar a equação 4.B.2, usando a equação 4.B.3, para:

$$\underline{V}(n) = \sum_{j=n-K}^{n-1} A(\overline{H}_g)^{n-j-1} \underline{b}(\overline{H}_g) x(j) \quad (4.B.4)$$

Observando que as contribuições ao somatório de 4.B.4 são desprezíveis para $j < n-K$, pode-se incluí-las nesse somatório. Dessa forma:

$$\underline{V}(n) \approx \sum_{j=0}^{n-1} A(\overline{H}_g)^{n-j-1} \underline{b}(\overline{H}_g) x(j) \triangleq \underline{V}(n, \overline{H}_g) \quad (4.B.5)$$

desde que $A(\overline{H}_g)$ seja tal que garanta estabilidade. A equação 4.B.5 pode ser escrita recursivamente como:

$$\begin{aligned} \underline{V}(n+1, \overline{H}_g) &= A(\overline{H}_g) \underline{V}(n, \overline{H}_g) + \underline{b}(\overline{H}_g) x(n) \\ \underline{V}(0, \overline{H}_g) &= 0 \end{aligned} \quad (4.B.6)$$

Em conseqüência, por analogia com os resultados acima, pode-se escrever para os demais parâmetros do algoritmo:

$$\hat{y}(n) \approx y(n, \overline{H}_g) \quad (4.B.7.a)$$

$$\underline{\eta}(n) \approx \underline{\eta}(n, \overline{H}_g) \quad (4.B.7.b)$$

$$e(n) \approx e(n, \overline{H}_g) \quad (4.B.7.c)$$

onde, se determina:

$$\begin{bmatrix} \hat{y}(n+1, \underline{H}_g) \\ \underline{\eta}(n+1, \underline{H}_g) \end{bmatrix} = C(\underline{H}_g) \underline{V}(n+1, \underline{H}_g) \quad (4.B.8.a)$$

$$e(n, \underline{H}_g) = y(n) - \hat{y}(n/\underline{H}_g) \quad (4.B.8.b)$$

Quando $H_g(n)$ está na proximidade de \bar{H}_g e supondo também a matriz $R(n)$ suficientemente próxima de um determinado valor \bar{R} pode-se, com n muito grande, usar a aproximação 4.B.7 no algoritmo geral para concluir que:

$$\underline{H}_g(n) \approx \underline{H}_g(n-1) + \gamma(n) \bar{R}^{-1} \underline{\eta}(n, \underline{H}_g) e(n, \underline{H}_g) \quad (4.B.9.a)$$

$$R(n) \approx R(n-1) + \gamma(n) [\underline{\eta}(n, \underline{H}_g) \underline{\eta}^T(n, \underline{H}_g) - R] \quad (4.B.9.b)$$

Introduz-se, agora, os seguintes valores esperados:

$$\underline{f}(\underline{H}_g) \triangleq E [\underline{\eta}(n, \underline{H}_g) e(n, \underline{H}_g)] \quad (4.B.10)$$

e

$$\underline{G}(\underline{H}_g) \triangleq E [\underline{\eta}(n, \underline{H}_g) \underline{\eta}^T(n, \underline{H}_g)] \quad (4.B.11)$$

com relação a seqüência de dados disponíveis.

Na equação 4.A.18.d, toda vez que a estimativa dos parâmetros \underline{H}_g muda de valor, um transiente indesejável aparece em $\underline{V}(n+1)$. Necessita-se, assim, da condição que $\gamma(n)$ seja pequeno, tornando a adaptação lenta. Já na equação 4.B.8, não ocorre esse problema, desde que n seja grande. Assim, supondo as equações 4.B.10 e 4.B.11 invariantes no tempo, tem-se:

$$\underline{H}_g(n) \approx \underline{H}_g(n-1) + \gamma(n) \bar{R}^{-1} \underline{f}(\underline{H}_g) + \gamma(n) \underline{N}(n) \quad (4.B.12.a)$$

e

$$R(n) \approx R(n-1) + \gamma(n) [\underline{G}(\underline{H}_g) - \bar{R}] + \gamma(n) N(n) \quad (4.B.12.b)$$

onde $\underline{N}(n)$ e $N(n)$ são variáveis aleatórias de média nula. Seja $\Delta\tau$ um número pequeno definido por:

$$\Delta\tau \triangleq \sum_{k=n}^{n'} \gamma(k) \quad (4.B.13)$$

Considerando que, na interação anterior, já se atinja o valor fixo \underline{H}_g e \bar{R} , ou seja, $\underline{H}_g(n) = \underline{H}_g$ e $R(n) = \bar{R}$, reescreve-se as equações 4.B.12 com $n' > n$, ou seja:

$$\underline{H}_g(n') \approx \underline{H}_g + \Delta\tau \bar{R}^{-1} \underline{f}(\underline{H}_g) + \sum_{k=n}^{n'} \gamma(k) \underline{N}(k) \quad (4.B.14.a)$$

$$R(n') \approx \bar{R} + \Delta\tau [G(\underline{H}_g) - \bar{R}] + \sum_{k=n}^{n'} \gamma(k) N(k) \quad (4.B.14.b)$$

Desde que $\underline{N}(k)$ e $N(k)$ têm média nula, o terceiro termo da equação 4.B.14 tende a contribuir com uma magnitude muito pequena. Portanto:

$$\underline{H}_g(n') \approx \underline{H}_g + \Delta\tau \bar{R}^{-1} \underline{f}(\underline{H}_g) \quad (4.B.15.a)$$

$$R(n') \approx \bar{R} + \Delta\tau [G(\underline{H}_g) - \bar{R}] \quad (4.B.15.b)$$

Com uma troca de escala, de acordo com a equação 4.B.13, tal que $n \leftrightarrow \tau$ e $n' \leftrightarrow \tau + \Delta\tau$, pode-se reescrever:

$$\frac{\underline{H}_g(\tau + \Delta\tau) - \underline{H}_g(\tau)}{\Delta\tau} \approx \bar{R}^{-1}(\tau) \underline{f}(\underline{H}_g(\tau)) \quad (4.B.16.a)$$

$$\frac{R(\tau + \Delta\tau) - R(\tau)}{\Delta\tau} \approx G(\underline{H}_g(\tau)) - R(\tau) \quad (4.B.16.b)$$

e no limite, para $\Delta\tau \rightarrow 0$, aproxima-se da equação diferencial

$$\frac{\partial}{\partial\tau} \underline{H}_{g_D}(\tau) = \bar{R}_D^{-1}(\tau) \underline{f}(\underline{H}_{g_D}(\tau)) \quad (4.B.17.a)$$

$$\frac{\partial}{\partial\tau} R_D(\tau) = G(\underline{H}_{g_D}(\tau)) - R_D(\tau) \quad (4.B.17.b)$$

onde o índice D é colocado para distinguir a solução da equação diferencial

4.B.17 das variáveis do algoritmo. Nota-se que a equação 4.B.15 é o método de Euler para a solução numérica da Equação Diferencial Ordinária (ODE) 4.B.17. A cadeia de argumentos adotada até aqui sugere que, se para algum n_0 muito grande:

$$\underline{H}_g(n_0) = \underline{H}_{g_D}(\tau_0), \quad \mathbf{R}(n_0) = \mathbf{R}_D(\tau_0) \quad \text{e} \quad \sum_{k=1}^{n_0} \gamma(k) = \tau_0 \quad (4.B.18)$$

então para $n > n_0$:

$$\underline{H}_g(n) \approx \underline{H}_{g_D}(\tau), \quad \mathbf{R}(n) = \mathbf{R}_D(\tau) \quad \text{e} \quad \sum_{k=1}^n \gamma(k) = \tau \quad (4.B.19)$$

Essa última relação é que estabelece a ligação entre o algoritmo da equação 4.A.18 com a equação diferencial 4.B.17 de maneira heurística, ou seja, que a trajetória de 4.B.17 descreve o comportamento assintótico do algoritmo, quando $\gamma(n)$ é muito pequeno. Assim, as estimativas dos parâmetros, $\underline{H}_g(n)$, devem seguir assintoticamente as trajetórias da equação diferencial.

Geralmente, o comportamento de uma equação diferencial é analisado em termos de suas propriedades de estabilidade, as quais devem, então ser verificadas. Dessa forma, alguns pontos da teoria de estabilidade devem ser lembrados.

● *Segundo Método de Estabilidade de Lyapunov*

Seja a equação diferencial (estado) $[\dot{\theta} = \partial\theta(\tau)/\partial\tau]$:

$$\dot{\theta} = f(\theta) \quad (4.B.20)$$

Um conjunto D_c é invariante para a equação diferencial se:

$$\theta(0) \in D_c \implies \theta(\tau) \in D_c \quad \forall \tau \quad (4.B.21)$$

Um ponto estacionário $\bar{\theta}$ é tal que:

$$f(\bar{\theta}) = 0 \quad (4.B.22)$$

Os pontos estacionários constituem, então, um conjunto invariante, uma vez que:

$$\theta(0) = \bar{\theta} \implies \dot{\theta} = 0 \implies \theta(\tau) = \bar{\theta} \quad \forall \tau \quad (4.B.23)$$

Um domínio de atração D_A , do conjunto D_c , é tal que:

$$\theta(0) \in D_A \implies \theta(\tau) \text{ tende a } D_c \text{ quando } \tau \rightarrow \infty \quad (4.B.24)$$

Logo $D_c \subset D_A$. Se D_A contém a vizinhança de D_c , D_c é estável. Se D_A é o conjunto no qual a equação diferencial é definida, pode-se falar em estabilidade assintótica global de D_c .

Assim, diz-se que uma função $V(\theta)$ positiva, ou seja,

$$V(\theta) \geq 0 \quad \forall \theta(\tau) \quad (4.B.25)$$

é Lyapunov (estável segundo Lyapunov), quando:

$$(i) \quad \frac{\partial}{\partial \tau} V(\theta(\tau)) = V'(\theta(\tau)) f(\theta(\tau)) \leq 0 \quad \forall \theta(\tau) \quad (4.B.26)$$

$$(ii) \quad \frac{\partial}{\partial \tau} V(\theta(\tau)) = 0 \quad \implies \quad \theta(\tau) \in D_c \quad (4.B.27)$$

ou seja, fora de D_c , $V(\theta)$ é estritamente decrescente, como função de τ , mas como $V(\theta)$ é limitada inferiormente, $\theta(\tau)$ tende para D_c . Isso mostra que D_c é um conjunto invariante assintoticamente estável.

A conexão entre o algoritmo e a equação diferencial pode ser feita, a partir das seguintes conjecturas:

- a) Supor que D_c é um conjunto invariante para a equação diferencial (4.B.17) e D_A , seu domínio de atração. Então, desde que $\underline{H}_g(n) \in D_A$, $\underline{H}_g(n)$ tenderá para D_c com probabilidade um (w.p.1) quando n tender para infinito. Isso é uma característica de uma convergência local.
- b) Somente pontos estacionários estáveis na equação 4.B.17 são pontos possíveis de convergência do algoritmo da equação 4.A.18.
- c) As trajetórias $\underline{H}_{g_D}(\tau)$ da equação 4.B.17 são os caminhos assintóticos das estimativas $\underline{H}_g(n)$, gerados pelo algoritmo 4.18.

Dessa forma, pode-se resumir a seguinte seqüência para a análise de convergência do algoritmo 4.A.18:

- 1) Computar o erro de estimação $e(n, \underline{H}_g)$ e a aproximação gradiente $\underline{\eta}(n, \underline{H}_g)$, para um fixo e constante modelo \underline{H}_g .
- 2) Calcular a direção média de adaptação para o algoritmo, baseado nessas variáveis, ou seja, $\underline{f}(\underline{H}_g)$ e $\underline{G}(\underline{H}_g)$.

3) Definir a equação diferencial a partir dessas direções, isto é:

$$\dot{\underline{H}}_{g_D} = \underline{R}_D^{-1} f(\underline{H}_{g_D}) \quad (4.B.28.a)$$

$$\dot{\underline{R}}_D = \underline{G}(\underline{H}_{g_D}) - \underline{R}_D \quad (4.B.28.b)$$

4) Estudar as propriedades de estabilidade dessas equações diferenciais.

As proposições (a), (b) e (c) são formalmente demonstradas por Ljung nos teoremas 1, 2 e 3 da referência [29].

Na dedução formal do método da equação diferencial é imperativo que $\gamma(n) \rightarrow 0$ quando $n \rightarrow \infty$ [5]. Essa exigência ocorre por vários motivos técnicos da demonstração. Duas conseqüências interessantes advém desta exigência. A primeira é que quando $\gamma(n) \rightarrow 0$, a superfície da energia do erro de adaptação (quantidade que se deseja minimizar) tende a variar cada vez menos com o passar do tempo. Isto implica que o ponto de mínimo para onde a estimativa de $\underline{H}_g(n)$ está convergindo tenderá a ficar cada vez mais fixo no espaço dos parâmetros. Tal fato favorece a que as estimativas de $\underline{H}_g(n)$ fiquem cada vez mais próximas desse ponto de mínimo da superfície. Como conseqüência, a variância das variáveis aleatórias $\underline{N}(n)$ e $N(n)$ da equação 4.B.12 tendem a zero a medida em que o tempo passa. Tal fato garante a aproximação feita nas passagens da equação 4.B.14 para a equação 4.B.15.

A segunda conseqüência interessante do fato de $\gamma(n) \rightarrow 0$ quando $n \rightarrow \infty$ surge ao se considerar a equação 4.B.15 como uma solução numérica da equação 4.B.17. Ora, quanto menor for $\gamma(n)$, menor será $\Delta\tau$ e, conseqüentemente, melhor será a aproximação da equação diferencial.

4.B.2.- Condições de Regularidade do Algoritmo Geral

Para o estabelecimento formal da associação entre a equação diferencial, dada na equação 4.B.17, e o algoritmo geral, da equação 4.A.18, é necessária a imposição de certas condições sobre os dados do algoritmo. Duas destas condições merecem destaque especial. A primeira diz respeito à positividade da matriz $\underline{R}(n)$, e a segunda à estabilidade do filtro adaptativo.

A matriz $\underline{R}(n)$ é, normalmente, obrigada a ser positiva definida. Tal exigência pode ser intuitivamente compreendida pelo fato de $\underline{R}(n)$ desempenhar o

papel de Hessiano para a função objetivo, a cada iteração do algoritmo. Caso a matriz $R(n)$ seja singular, o algoritmo Gauss-Newton terá problemas numéricos devido à equação 4.A.18.c. Se alguma implementação especial deste algoritmo é utilizada, como por exemplo, usando o lema de inversão de matrizes, ou usando algoritmos rápidos [1], cuidados especiais devem ser tomados também com a possibilidade de $R(n)$ deixar de ser positiva semi-definida. Algumas técnicas mais usadas para a garantia da positividade da matriz $R(n)$ são: reinicialização da matriz por δI , fatorização U-D e método de Levenberg-Marquardt [1].

Quanto aos parâmetros do filtro adaptativo, é necessária a utilização de métodos que monitorem e corrijam eventuais instabilidades. Isto ocorre pois constata-se na prática que, quando o filtro fica com pólos instantâneos fora da região de estabilidade, geralmente o algoritmo diverge. Tal fato é especialmente verdadeiro para algoritmos do tipo erro de estimação recursivo.

Assim, a partir do algoritmo geral da equação 4.A.18, Ljung [5] propõe condições para se discutir a convergência. Definindo-se DM como um conjunto de modelos possíveis para um determinado sistema, tem-se as seguintes condições:

- Condições do Conjunto de Modelos

M1 - O conjunto de modelo DM é compacto, ou seja, ele é limitado e fechado tal que $\underline{H}_g \in DM$. Isso significa que, $A(\underline{H}_g)$ tem todos os autovalores estritamente dentro do círculo unitário, ou seja, é estável.

M2 - As matrizes $A(\underline{H}_g)$, $C(\underline{H}_g)$ e $\underline{b}(\underline{H}_g)$ são continuamente diferenciáveis com relação a \underline{H}_g , $\forall \underline{H}_g \in DM$, ou seja, o modelo é diferenciável.

- Condições de Amortecimento

Cr1- A função que incrementa a atualização do vetor de parâmetros $\underline{H}_g(n)$ tem um comportamento suave, $\forall \underline{H}_g \in DM$, sendo diferenciável com relação a \underline{H}_g , e e $\underline{\eta}$.

Cr2- A função que incrementa a atualização da matriz $R(n)$ tem um comportamento suave, $\forall \underline{H}_g \in DM$, sendo diferenciável com relação a R , \underline{H}_g , e e $\underline{\eta}$.

Cr3- e Cr4 - A função critério tem comportamento suave, $\forall \underline{H}_g \in DM$, sendo diferenciável à terceira com relação a e e $\underline{\eta}$.

- *Condições da Direção de Busca*

R1 - A recursividade de $R(n)$ é tal que $R(n)$ seja simétrica e $R(n) \geq \delta I$ $\forall n$ e algum $\delta > 0$, ou seja, $R(n) - \delta I$ é semi-positiva definida.

G1 - A seqüência de ganho $\{\gamma(n)\}$ tem a restrição $\lim_{n \rightarrow \infty} n \cdot \gamma(n) = \mu > 0$.

- *Condições dos Dados*

A1 - A geração dos dados é assintoticamente estacionária. Assim, para uma seqüência de dados disponíveis, existe os seguintes limites, com $\underline{H}_g \in DM$:

$$\begin{aligned} & \cdot \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \underline{\eta}(n, \underline{H}_g) \cdot e(n, \underline{H}_g) \triangleq \underline{f}(\underline{H}_g) \\ & \cdot \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \underline{\eta}(n, \underline{H}_g) \cdot \underline{\eta}^T(n, \underline{H}_g) \triangleq \underline{G}(\underline{H}_g) \end{aligned}$$

A condição A1 deve ser satisfeita com probabilidade 1. Para isso:

A2 - Os limites são definidos em termos dos valores esperados com relação aos dados, ou seja,

$$\begin{aligned} & \cdot \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E \left\{ \underline{\eta}(n, \underline{H}_g) \cdot e(n, \underline{H}_g) \right\} = \underline{f}(\underline{H}_g) \\ & \cdot \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E \left\{ \underline{\eta}(n, \underline{H}_g) \cdot \underline{\eta}^T(n, \underline{H}_g) \right\} = \underline{G}(\underline{H}_g) \end{aligned}$$

S2 - A geração dos dados é exponencialmente estável, ou seja, para cada n, m com $n \geq m$, existe uma variável aleatória $x_m(n)$ pertencente a seqüência $\{x(n), \dots, x(0)\}$ e independente de $\{x(m), \dots, x(0)\}$, tal que $E\{|x(n) - x_m(n)|^4\} < C\lambda^{m-n}$, $C < \infty$ e $\lambda < 1$ (ver Ljung [5]).

Dessa forma, sob as condições de regularidade expostas, Ljung e Soderstrom demonstram as conjecturas (a), (b) e (c) citadas, que definem a convergência do algoritmo geral da equação 4.A.18, através do Teorema 4.2, para a proposição (a), resultado 4.3, para a proposição (b) e resultado 4.2, para a proposição (c) na referência [5].

APÊNDICE 4.C

ANÁLISE DE CONVERGÊNCIA PARA O MÉTODO DO ERRO DE ESTIMAÇÃO RECURSIVO

O método do erro de estimação recursivo (RPEM, segundo Ljung e Soderstron [5]), é uma das mais genéricas extensões dos mínimos quadrados, quando não se tem conhecimento da estrutura do filtro utilizado, sendo desenvolvido a partir da aproximação estocástica. A adaptação é feita através do algoritmo geral da equação 4.A.18, sendo a geração do gradiente função do erro de estimação, ou seja:

$$\underline{\eta}(n) = \underline{\psi}(n) = - \frac{\partial e(n)}{\partial \underline{H}_g} \quad (4.C.1)$$

O algoritmo é descrito, então, pela Tabela 4.C.1:

$e(n) = y(n) - \hat{y}(n)$	(4.C.2.a)
$\underline{H}_g(n) = \underline{H}_g(n-1) + \gamma(n) \underline{R}^{-1}(n) \underline{\psi}(n) e(n)$	(4.C.2.b)
$\underline{R}(n) = \underline{R}(n-1) + \gamma(n) [\underline{\psi}(n) \underline{\psi}^T(n) - \underline{R}(n-1)]$	(4.C.2.c)
$\underline{V}(n+1) = \underline{A}(\underline{H}_g(n)) \underline{V}(n) + \underline{b}(\underline{H}_g(n)) x(n)$	(4.C.2.d)
$\begin{bmatrix} \hat{y}(n+1) \\ \underline{\psi}(n+1) \end{bmatrix} = \underline{C}(\underline{H}_g(n)) \cdot \underline{V}(n+1)$	(4.C.2.e)

TABELA 4.C.1 - Algoritmo RPEM

considerando o critério quadrático e a direção Gauss-Newton. Para a análise de convergência, primeiramente, define-se:

$$\bar{E}(\bullet) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N E(\bullet) \quad (4.C.3)$$

introduzindo a seguinte condição:

A3 - Os limites

$$\bullet \bar{E} \left[\frac{1}{2} e^2(n, \underline{H}_g) \right] \triangleq \bar{J}(\underline{H}_g) \quad (4.C.4.a)$$

$$\bullet \bar{E} \left[\underline{\psi}(n, \underline{H}_g) \cdot e(n, \underline{H}_g) \right] \triangleq \underline{f}(\underline{H}_g) \quad (4.C.4.b)$$

$$\bullet \bar{E} \left[\underline{\psi}(n, \underline{H}_g) \cdot \underline{\psi}^T(n, \underline{H}_g) \right] \triangleq \underline{G}(\underline{H}_g) \quad (4.C.4.d)$$

existem $\forall \underline{H}_g \in DM$.

Observa-se que:

$$\underline{f}(\underline{H}_g) = - \bar{E} \left[\frac{1}{2} \frac{\partial e^2}{\partial \underline{H}_g}(n, \underline{H}_g) \right]^T = - \left[\frac{\partial}{\partial \underline{H}_g} \bar{J}(\underline{H}_g) \right]^T \quad (4.C.5)$$

Considerando a trajetória da equação 4.B.17, pode-se escrever:

$$\frac{\partial}{\partial \tau} \bar{J}(\underline{H}_{g_D}(\tau)) = \frac{\partial}{\partial \underline{H}_g} \bar{J}(\underline{H}_g) \Big|_{\underline{H}_g = \underline{H}_g(\tau)} \cdot \underline{R}^{-1} \underline{f}(\underline{H}_{g_D}(\tau)) \quad (4.C.6)$$

e substituindo a equação 4.C.5, chega-se a:

$$\frac{\partial}{\partial \tau} \bar{J}(\underline{H}_{g_D}(\tau)) = - \underline{f}^T(\underline{H}_g(\tau)) \underline{R}^{-1} \underline{f}(\underline{H}_{g_D}(\tau)) \quad (4.C.7)$$

o que mostra que \bar{J} é decrescente fora do conjunto:

$$D_c = \{ \underline{H}_g / \underline{f}(\underline{H}_g) = 0 \} \quad (4.C.8)$$

Dessa forma, $\bar{J}(\underline{H}_g)$ é uma função Lyapunov de 4.B.17.a, de acordo com as equações 4.B.20-4.B.27. Assim, a trajetória da equação 4.B.17.a convergirá para D_c quando $\tau \rightarrow \infty$. De acordo com a conjectura (a), isso implica que

$$\underline{H}_g(n) \rightarrow D_c \quad \text{com prob. 1} \quad \text{para } n \rightarrow \infty \quad (4.C.9)$$

ou, ainda, que \underline{H}_g tende para a fronteira de DM , o que não pode ser excluído a menos que o particular caminho $\underline{H}_g(n)$ seja projetado em DM (ver Ljung *et al.* [5, pp. 162]). Também, da conjectura (b), pontos que não são mínimos locais de $\bar{J}(\underline{H}_g)$ podem ser excluídos de D_c . Encontra-se, assim, que o estimador RPEM

convergir  com prob. 1 para um m nimo local de $\bar{J}(\underline{H}_g)$ (ou para a fronteira de DM) quando $n \rightarrow \infty$. Esse resultado   o mesmo da contra-partida "off-line".

Resumindo, escreve-se o seguinte Teorema:

Teorema 4.C.1 - Seja o algoritmo da equa o 4.C.2. Supondo as condi es M1, M2, Cr2, Cr3, A3, S1, R1, G1, ent o $\underline{H}_g(n) \in DM$ convergir , com prob. 1, para o conjunto

$$D_c = \left\{ \underline{H}_g / \frac{\partial}{\partial \underline{H}_g} \bar{J}(\underline{H}_g) = \underline{0} \right\} \quad (4.C.10)$$

ou seja, para o m nimo local de $\bar{J}(\underline{H}_g)$, ou para a fronteira de DM quando $n \rightarrow \infty$, tomando $\gamma(n) \rightarrow 0$.

A prova deste teorema est  em [5, pag. 182].

APÊNDICE 4D

CÁLCULO DO DETERMINANTE DA MATRIZ JACOBIANA

Sejam as funções de transferência, do filtro FIR direto:

$$H_D = 1 + d_1 z^{-1} + \dots + d_M z^{-M} \quad (4.D.1)$$

e do filtro em cascata (considerando, aqui, secções de 1ª ordem):

$$H_C = \prod_{i=1}^M (1 + c_i z^{-1}) \quad (4.D.2)$$

Pode-se escrever a correspondência, tal que:

$$H_D(\underline{Hd}) = H_C(\underline{Hc}) \quad (4.D.3)$$

ou seja

$$1 + d_1 z^{-1} + \dots + d_M z^{-M} \equiv (1 + c_i z^{-1}) \prod_{\substack{j=1 \\ j \neq i}}^M (1 + c_j z^{-1}) \quad (4.D.4)$$

O mapeamento leva à matriz Jacobiana:

$$\frac{\partial \underline{Hd}}{\partial \underline{Hc}} = \Omega \quad (M \times M) \quad (4.D.5)$$

cujos elementos são tais que:

$$\frac{\partial d_k}{\partial c_i} = \text{coeficiente de } z^{-k} \text{ em } \left\{ z^{-1} \prod_{\substack{j=1 \\ j \neq i}}^M (1 + c_j z^{-1}) \right\} \quad i, k = 1, \dots, M \quad (4.D.6)$$

Seja

$$F_i(z^{-1}) \triangleq z^{-1} \prod_{\substack{j=1 \\ j \neq i}}^M (1 + c_j z^{-1}) \quad (4.D.7)$$

Constrói-se:

$$\gamma_{1i} \triangleq \text{coef. } z^{-1} \text{ em } F_i(z^{-1}) = 1 \quad (4.D.8)$$

$$\gamma_{2i} \triangleq \text{coef. } z^{-2} \text{ em } F_i(z^{-1}) = \sum_{\substack{m_1=1 \\ m_1 \neq i}}^M c_{m_1}$$

$$\gamma_{3i} \triangleq \text{coef. } z^{-3} \text{ em } F_i(z^{-1}) = \sum_{\substack{m_1, m_2=1 \\ m_1 \neq m_2 \neq i}}^M c_{m_1} c_{m_2}$$

⋮

$$\gamma_{ki} \triangleq \text{coef. } z^{-k} \text{ em } F_i(z^{-1}) = \sum_{\substack{m_1, m_2, \dots, m_{k-1}=1 \\ m_1 \neq m_2 \neq \dots \neq m_{k-1} \neq i}}^M c_{m_1} \dots c_{m_{k-1}}$$

com

$$\Omega_{M \times M} = \begin{bmatrix} 1 & 1 & \dots & 1 & \dots & 1 \\ \gamma_{21} & \gamma_{22} & \dots & \gamma_{2i} & \dots & \gamma_{2M} \\ \vdots & \vdots & & \vdots & & \vdots \\ \gamma_{k1} & \gamma_{k2} & \dots & \gamma_{ki} & \dots & \gamma_{kM} \\ \vdots & \vdots & & \vdots & & \vdots \\ \gamma_{M1} & \gamma_{M2} & \dots & \gamma_{Mi} & \dots & \gamma_{MM} \end{bmatrix} \quad (4.D.9)$$

Para se achar $\det[\Omega]$, faz-se redução das adjuntas. Primeiro, subtrai-se a primeira coluna das demais, sendo que, para cada coluna, tem-se

$$\gamma_{ki} - \gamma_{k1} = (c_1 - c_i) \left\{ \text{coef. de } z^{-1} \text{ no polinômio } z^{-1} \prod_{\substack{j=2 \\ j \neq i}}^M (1 + c_j z^{-1}) \right\} \quad (4.D.10)$$

onde a expressão nas chaves é igual a

$$(*) = \frac{F_i(z^{-1})}{(1 + c_1 z^{-1})} \quad (4.D.11)$$

donde

$$\gamma_{ki} - \gamma_{ki} \stackrel{\Delta}{=} (c_1 - c_i) \cdot \delta_{(k-1)(i-1)} \quad (4.D.12)$$

dessa forma,

$$\det[\Omega] = \begin{bmatrix} 1 & 0 & \dots & 0 & \dots & 0 \\ \gamma_{21} & (c_1 - c_2) & & (c_1 - c_i) & & (c_1 - c_M) \\ \vdots & \vdots & & \vdots & & \vdots \\ \gamma_{k1} & (c_1 - c_2)\delta_{(k-1).1} & \dots & (c_1 - c_i)\delta_{(k-1)(i-1)} & \dots & (c_1 - c_M)\delta_{(k-1)(M-1)} \\ \vdots & \vdots & & \vdots & & \vdots \\ \gamma_{M1} & (c_1 - c_2)\delta_{(M-1).1} & & (c_1 - c_i)\delta_{(M-1)(i-1)} & & (c_1 - c_M)\delta_{(M-1)(M-1)} \end{bmatrix} \quad (4.D.13)$$

$$\det[\Omega] = \left\{ \prod_{i=2}^M (c_1 - c_i) \right\} \cdot \det[\text{Adjunta}] \quad (4.D.14)$$

e assim, sucessivamente,

$$\det[\Omega] = \prod_{\substack{i,j=1 \\ i < j}}^M (c_i - c_j) \quad i, j = 1, \dots, M. \quad (4.D.15)$$

CAPÍTULO 5

UMA ESTRUTURA ADAPTATIVA E ALGORITMO RÁPIDO PARA MODELAGEM DE VOZ POR PARES DE RAIAS ESPECTRAIS (LSP)

Como citado no Capítulo 2, uma das mais eficazes técnicas de análise de voz é feita através da predição linear. Tem-se utilizado essa técnica para estimação de parâmetros básicos, tipo: pitch, formantes, espectros, funções do trato vocal, etc...; e para representação da voz em transmissão com baixas taxas de bits. A importância desse método está na precisão da estimação, bem como na relativa rapidez computacional [9, 10, 24].

A modelagem de sinais por "Pares de Raias Espectrais" (LSP - Line Spectrum Pairs) foi proposta por Itakura [15], como uma alternativa para a codificação preditiva de sinais de voz, tendo crescido o interesse em seu uso nos últimos anos. Algumas vantagens são comumente atribuídas aos parâmetros LSP, em relação aos originais coeficientes de predição linear transversal (LPC - "Linear Prediction Code") e aos coeficientes de reflexão (predição linear em estrutura em treliça - PARCOR). Em particular, os coeficientes LSP são tidos como mais robustos aos efeitos de quantização, o que conduz a uma codificação mais eficiente do sinal. Essa robustez deve-se à sensibilidade uniforme ao longo do espectro de frequência do sinal de voz a ser codificado [16].

Os parâmetros LSP podem ser calculados a partir dos elementos da autocorrelação do sinal de entrada, através de um processamento por bloco de amostras, como na análise LPC convencional. Entretanto, para diversas aplicações em tempo real é interessante o uso de técnicas adaptativas e, nesse sentido, algumas técnicas baseadas no algoritmo do gradiente (LMS) já foram apresentadas na literatura [16, 17, 42].

O objetivo deste capítulo é também propor uma técnica adaptativa de obtenção dos coeficientes LSP, porém baseada nos algoritmos recursivos de mínimos quadrados. Para se chegar nesse resultado, duas principais contribuições são fornecidas:

- (i) A partir da análise teórica dos parâmetros LSP, é mostrado que eles podem ser associados a um par de filtros de erro de predição com fase linear decompostos em secções de segunda ordem.
- (ii) Em conseqüência, constata-se que a configuração multidimensional e o algoritmo FLS-MD propostos no Capítulo 3 podem ser utilizados para adaptar esses filtros em cascata.

Dessa forma, uma interessante estrutura adaptativa é então proposta para a modelagem LSP; ou seja, é apresentada uma técnica direta e eficiente para se obter os parâmetros LSP, sem conversões intermediárias [14, 43, 44].

O desempenho do método proposto é analisado através de simulações onde se utiliza dois trechos diferentes de sinais de voz digitalizados. Antes porém de se apresentar o desenvolvimento da proposta e os resultados obtidos, uma revisão sobre a modelagem LSP será feita, definindo-se os parâmetros do preditor LSP e inserindo o problema da adaptação.

5.1. OS PARÂMETROS LSP

A definição dos parâmetros LSP se baseia no princípio da predição linear. Seja um sinal de entrada $x(n)$ para o qual se obtém o filtro ótimo de erro de predição, cuja função de transferência é dada por:

$$1 - A(z) = 1 - \sum_{i=1}^M a_i z^{-i} \quad (5.1)$$

onde o conjunto de coeficientes ótimos é obtido pela minimização do erro quadrático médio de predição $E[e^2(n)]$, sendo:

$$e(n) = x(n) - \sum_{i=1}^M a_i x(n-i) \quad (5.2)$$

Pode-se, a partir da equação 5.1, definir os seguintes polinômios de ordem $(M+1)$:

$$P(z) = 1 - A(z) - z^{-(M+1)}[1 - A(z^{-1})] \quad (5.3)$$

$$Q(z) = 1 - A(z) + z^{-(M+1)}[1 - A(z^{-1})] \quad (5.4)$$

Verifica-se facilmente que esses polinômios são, respectivamente, anti-simétrico

e simétrico; portanto, suas raízes estão dispostas sobre o círculo de ralo unitário, donde se determina as frequências espectrais. Outra maneira interessante de se chegar a esta conclusão é observar que as equações 5.3 e 5.4 podem ser obtidas através de recursão na ordem dos coeficientes de predição, no algoritmo de Levinson-Durbin, [1], expressa por:

$$1 - A_{(M)}(z) = 1 - A_{(M-1)}(z) - k_{(M)} z^{-M} [1 - A_{(M-1)} z^{-1}] \quad (5.5)$$

quando os coeficientes de reflexão de ordem (M+1) valem: $k_{(M+1)} = +1$ e $k_{(M+1)} = -1$, respectivamente, para $P(z)$ e $Q(z)$. Ora, se os coeficientes de reflexão tem módulo unitário, pelas considerações dos coeficientes da estrutura treliça, os sinais são considerados senóides sem ruído ou seja, os zeros desses preditores estão necessariamente sobre o círculo unitário. Assim, tomando por exemplo M par, e verificando que $P(1) \equiv Q(-1) \equiv 0$, as seguintes fatorações podem ser efetuadas:

$$P(z) = (1 - z^{-1}) \prod_{k=1}^{M/2} (1 - 2 \cos \theta_k z^{-1} + z^{-2}) \quad (5.6)$$

$$Q(z) = (1 + z^{-1}) \prod_{k=1}^{M/2} (1 - 2 \cos \omega_k z^{-1} + z^{-2}) \quad (5.7)$$

Os dois conjuntos de parâmetros θ_k e ω_k ($k = 1, \dots, M/2$) expressos em (5.6) e (5.7), são definidos como frequências ou coeficientes LSP, podendo ser usados, ao invés dos coeficientes a_i ($i = 1, \dots, M$), para caracterizar o filtro de erro de predição.

Se $z_0 = e^{j\omega_0}$ denota um zero do polinômio $[1 - A_{(M)}(z)]$ sobre o círculo unitário, será também um zero de $P_M(z)$ e $Q_M(z)$. Agora, se o zero se move para dentro do círculo unitário, os correspondentes zeros de $P_M(z)$ e $Q_M(z)$ se moverão sobre o círculo unitário em direções opostas.

A condição necessária e suficiente para que o polinômio $[1 - A_M(z)]$ seja de fase mínima, como é requerida para síntese de LPC, é que os zeros de $P_M(z)$ e $Q_M(z)$ sejam simples e alternados ao longo do círculo unitário [45]. Dessa forma, considerando que as raízes ocorrem em pares de complexo conjugado, tem-se, a menos de $z = +1$ e $z = -1$:

$$0 < \omega_1 < \theta_1 < \omega_2 < \dots < \omega_{M/2} < \theta_{M/2} < \pi \quad (5.8)$$

que estão dispostos no semicírculo superior.

Em adição, um procedimento que determine um conjunto ordenado de parâmetros LSP pode ser usado para construir um filtro de erro de predição linear de fase mínima. Somando-se as equações 5.3 e 5.4, obtém-se o filtro de erro de predição, como combinação desses polinômios, ou seja:

$$1 - A(z) = \frac{1}{2} [P(z) + Q(z)] \quad (5.9)$$

Dessa forma, uma realização correspondente do filtro de erro de predição pode ser representada pelo esquema da Figura 5.1.

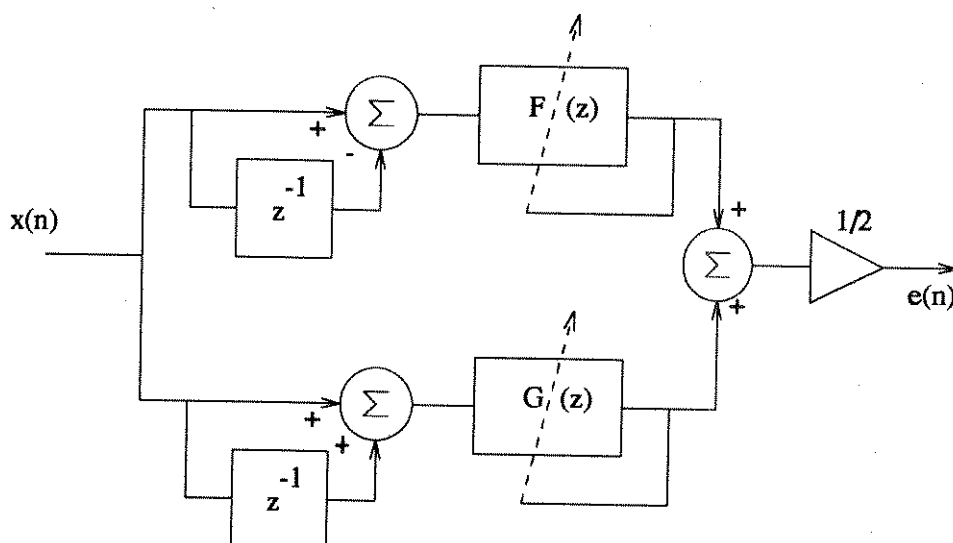


Fig. 5.1 - Preditor LSP

As funções de transferência $F(z)$ e $G(z)$, da Figura 5.1, correspondem aos termos contidos a direita nos produtórios das expressões 5.6 e 5.7, respectivamente. De modo a bem compreender o funcionamento desses filtros, deve-se reescrever suas funções de transferência, como:

$$F(z) = \prod_{k=1}^{M/2} (1 - 2 \cos \theta_k z^{-1} + z^{-2}) \quad (5.10)$$

$$G(z) = \prod_{k=1}^{M/2} (1 - 2 \cos \omega_k z^{-1} + z^{-2}) \quad (5.11)$$

Observa-se então que $F(z)$ e $G(z)$ são descritos por uma cascata de secções de 2ª ordem com fase linear, pois o segundo coeficiente de cada secção é igual a 1. Os seus zeros são todos complexos conjugados de módulo unitário e fase θ_k e ω_k .

Além disso, os parâmetros θ_k e ω_k são tais que otimizam o critério de mínimo erro quadrático médio aplicado ao sinal de saída $e(n)$, uma vez que o esquema geral da Figura 5.1 corresponde a um filtro de erro de predição.

Baseados nas considerações anteriores, constata-se que $F(z)$ e $G(z)$ comportam-se como filtros de erro de predição de seus respectivos sinais de entrada $[x(n) - x(n - 1)]$ e $[x(n) + x(n - 1)]$, com a particularidade de possuírem a restrição de fase linear. Isso, de fato, significa que $F(z)$ e $G(z)$ fornecem uma modelagem harmônica (CSM - "Composite Sinusoidal Modelling") dos respectivos sinais de entrada. As M frequências LSP (θ_k e ω_k) correspondentes ao sinal $x(n)$ aparecem, então, como sendo as mesmas frequências obtidas pela decomposição harmônica (CSM) dos sinais $[x(n) - x(n - 1)]$ e $[x(n) + x(n - 1)]$. Por sua vez, os coeficientes de predição (LPC) do sinal $x(n)$ podem ser obtidos a partir dos LSP pela equação 5.9. Esses mesmos resultados são obtidos por um procedimento distinto no elegante trabalho apresentado em [46], onde uma análise mais aprofundada das relações entre as modelagens LPC, CSM e LSP é apresentada.

A partir das interpretações acima e do esquema da Figura 5.1, o método desenvolvido no Capítulo 3 é sugerido para adaptar os filtros em cascata $F(z)$ e $G(z)$, obtendo assim, de forma adaptativa, os coeficientes LSP.

5.2. MODELAGEM ADAPTATIVA LSP

De acordo com as equações 5.10 e 5.11, na seção anterior, pode-se afirmar que os filtros $F(z)$ e $G(z)$ são formados por estruturas em cascata de secções de segunda ordem, cujas funções de transferência são, respectivamente

$$F_k(z) = 1 - f_k z^{-1} + z^{-2} \quad (5.12)$$

$$G_k(z) = 1 - g_k z^{-1} + z^{-2} \quad (5.13)$$

onde $f_k = 2 \cos \theta_k$ e $g_k = 2 \cos \omega_k$ para $k = 1, 2, \dots, K = M/2$. Dessa forma, os zeros desses filtros podem ser obtidos diretamente dos coeficientes ótimos de cada secção. O interesse, aqui, é obter esses coeficientes de maneira

adaptativa. Assim, de forma a utilizar o método descrito no Capítulo 3, a estrutura apresentada na Figura 3.7 deve ser aplicada em ambos os filtros da Figura 5.1. Entretanto, é necessário se levar em conta que, neste caso, as secções possuem a restrição de fase linear. Isto significa que apenas os coeficientes centrais de cada secção, f_k e g_k , devem ser atualizados, enquanto os outros permanecem iguais a 1.

De uma forma geral, o erro à saída de um preditor de 2ª ordem com fase linear é expresso por

$$e(n) = u(n) - h_1 u(n - 1) + u(n - 2) \quad (5.14)$$

onde $u(n)$ é a entrada e h_1 o coeficiente a ser otimizado.

A fim de aplicar diretamente a técnica de mínimos quadrados para atualizar o coeficiente h_1 , propõe-se a estrutura da Figura 5.2 onde se cria um sinal de referência que incorpora a restrição de fase linear.

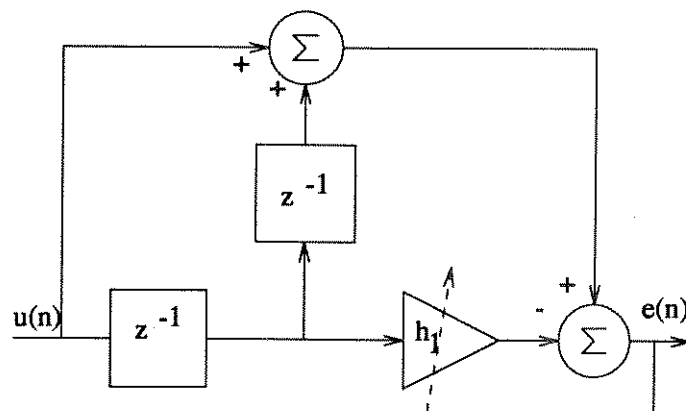


Fig. 5.2 - Preditor Adaptativo de Fase Linear

Assim o sinal de erro, nesse caso, é computado da forma:

$$e(n) = [u(n) + u(n - 2)] - h_1(n)u(n - 1) \quad (5.15)$$

Através dessa astúcia, substitui-se o preditor de 2ª ordem por um filtro adaptativo de 1ª ordem cujo sinal de referência é dado por

$$y(n) = [u(n) + u(n - 2)] \quad (5.16)$$

Aplicando o critério de mínimos quadrados, dado na equação 3.2 do Capítulo 3, sobre este filtro, pode-se mostrar que o coeficiente ótimo é dado por [23]:

$$h_1(n) = \frac{\sum_{p=1}^n W^{n-p} y(p) u(p-1)}{\sum_{p=1}^n W^{n-p} u^2(p-1)} = \frac{r_{yu}(n)}{r_{uu}(n)} \quad (5.17)$$

sendo obtido recursivamente por

$$h_1(n+1) = h_1(n) + r_{uu}^{-1}(n+1) u(n) e(n+1) \quad (5.18)$$

Esse resultado é então aplicável aos filtros das equações 5.12 e 5.13, para adaptar os parâmetros f_k e g_k .

Finalmente, a modelagem LSP adaptativa é obtida quando se associa o esquema da Figura 5.2 às secções que constituem o filtro multidimensional da Figura 3.5, combinando com o esquema obtido para adaptar $F(z)$ e $G(z)$ na Figura 5.1. A estrutura final proposta é apresentada na Figura 5.3.

A partir da filtragem adaptativa na forma em cascata, desenvolvida no Capítulo 3, aplicando-se a equação (3.35) à estrutura obtida na Figura 5.3, chega-se à seguinte atualização do vetor $\underline{F}(n)$, dos coeficientes f_k :

$$\underline{F}(n+1) = \underline{F}(n) + \frac{1}{K} \mathbf{R}_{uu}^{-1}(n) \underline{U}(n) e_1(n+1) \quad (5.19)$$

onde

$$\underline{F}^T(n) = [f_1(n), \dots, f_K(n)] \quad (5.20)$$

e

$$\underline{U}^T(n) = [u_1(n), \dots, u_K(n)] \quad (5.21)$$

sendo o erro expresso na forma:

$$e_1(n+1) = \sum_{k=1}^K [u_k(n+1) + u_k(n-1)] - \underline{F}^T(n) \underline{U}(n) \quad (5.22)$$

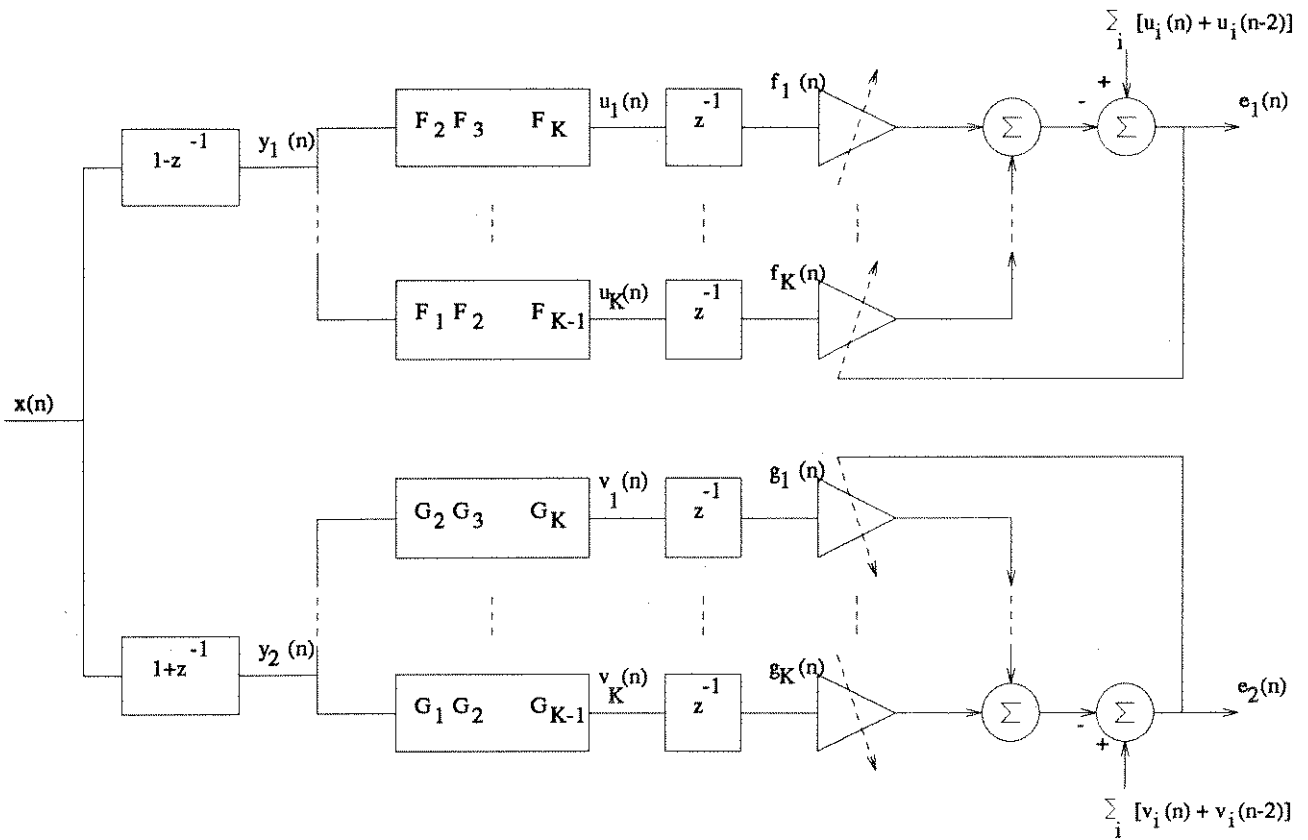


Fig. 5.3 - Estrutura Adaptativa para Estimação LSP

As mesmas expressões são usadas para adaptar o vetor de coeficientes $\underline{G}(n)$ empregando os sinais $v_k(n)$ e o erro $e_2(n)$, conforme indicado na Figura 5.3, ou seja:

$$\underline{G}(n+1) = \underline{G}(n) + (1/K) \underline{R}_{vv}^{-1}(n) \underline{V}(n) e_2(n+1) \quad (5.23)$$

com

$$\underline{G}^T(n) = [g_1(n), \dots, g_K(n)] \quad (5.24)$$

e

$$\underline{V}^T(n) = [v_1(n), \dots, v_K(n)] \quad (5.25)$$

e o erro

$$e_2(n+1) = \sum_{k=1}^k [u_k(n+1) + u_k(n-1)] - \underline{G}^T(n) \underline{U}(n) \quad (5.26)$$

Evidentemente, a versão rápida (algoritmo FLS-MD) pode ser implementada em lugar da expressão 5.18 conduzindo, matematicamente, à mesma recursão [47].

Enfim, as frequências LSP são diretamente obtidas a partir dos coeficientes $\underline{F}(n)$ e $\underline{G}(n)$, uma vez que

$$\theta_k(n) = \cos^{-1} \left(\frac{f_k(n)}{2} \right) \quad \text{e} \quad \omega_k(n) = \cos^{-1} \left(\frac{g_k(n)}{2} \right) \quad (5.27)$$

para $k = 1, \dots, M/2$.

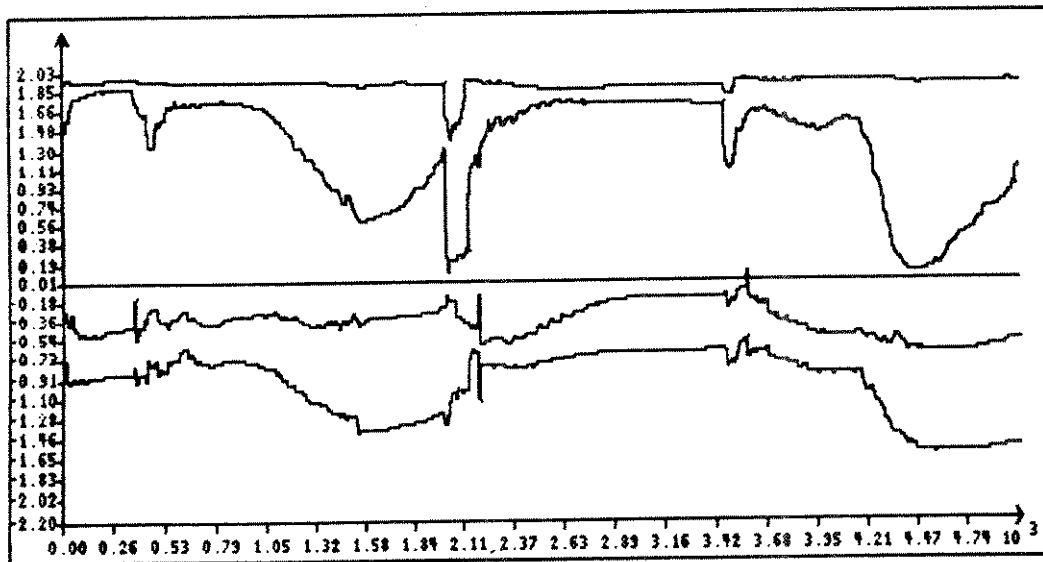
A complexidade computacional do algoritmo como um todo é proporcional ao quadrado do número de secções, $(M/2)^2$, devido às manipulações matriciais do algoritmo multidimensional, em cada parte.

5.3. RESULTADOS DAS SIMULAÇÕES

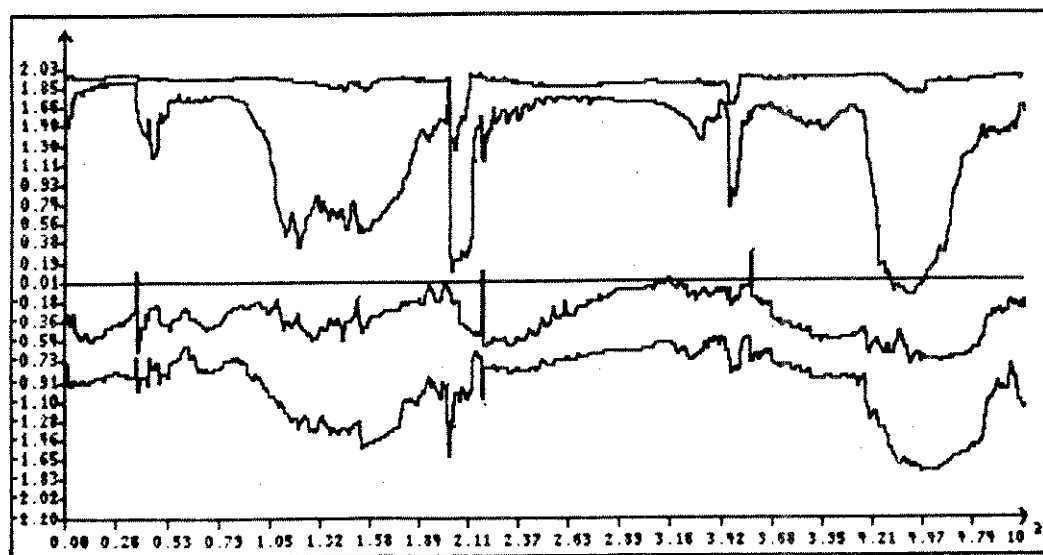
A técnica proposta neste capítulo foi aplicada na análise LSP de sinais de voz, utilizando-se filtros de quarta e sexta ordem, correspondente no caso em cascata a filtros com duas e três secções de segunda ordem. A evolução dos coeficientes $f_k(n)$ e $g_k(n)$ ($k = 1, 2$) é apresentada na Figura 5.4.a para o caso de duas secções empregando um fator de esquecimento $W = 0.99$. Foram consideradas 5000 amostras de um sinal de voz digitalizado, correspondentes a um trecho de 0,625 segundos (taxa de 96 kbits/s). Para avaliar o desempenho do algoritmo, é mostrado na Figura 5.4.c os valores teóricos desses parâmetros LSP, obtidos através do algoritmo de Levinson-Durbin, que fornece os coeficientes de predição, em conjunto com as equações 5.3 à 5.7. A análise de correlação foi feita para blocos de 256 amostras cada. A Figura 5.4.b mostra a capacidade de rastreamento quando se altera o fator de esquecimento, $W = 0.98$.

Cabe ressaltar que o sinal de voz utilizado foi extraído dos arquivos do Sistema de Análise e Processamento Digital de Voz (SAPDV-A, Laboratório de Processamento Digital de Voz da FEE-UNICAMP).

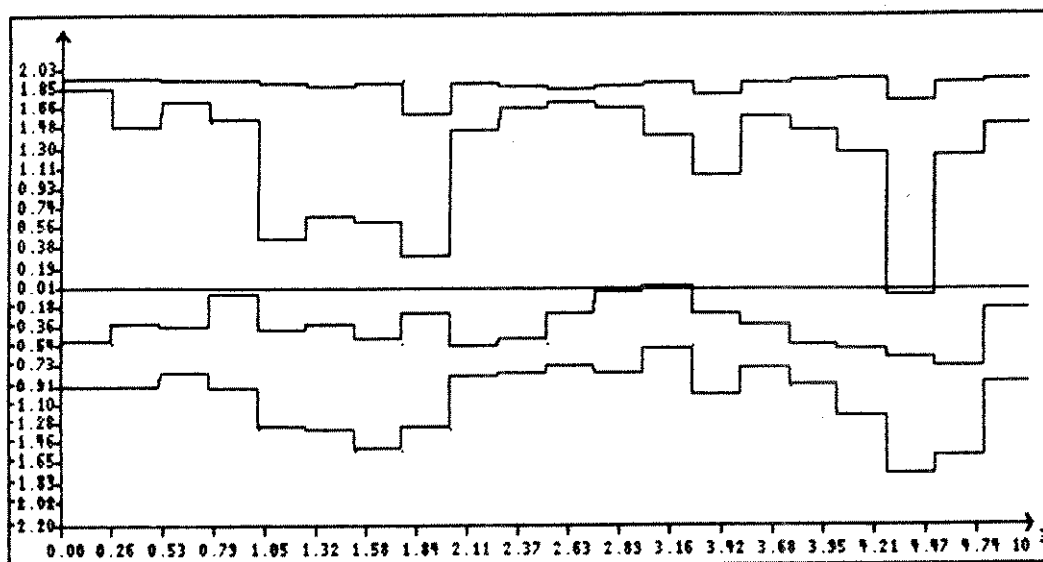
Um resultado semelhante é apresentado nas Figuras 5.5.a, b e c, porém utilizando-se três secções de 2ª ordem em cascata.



(a) $W = 0.99$

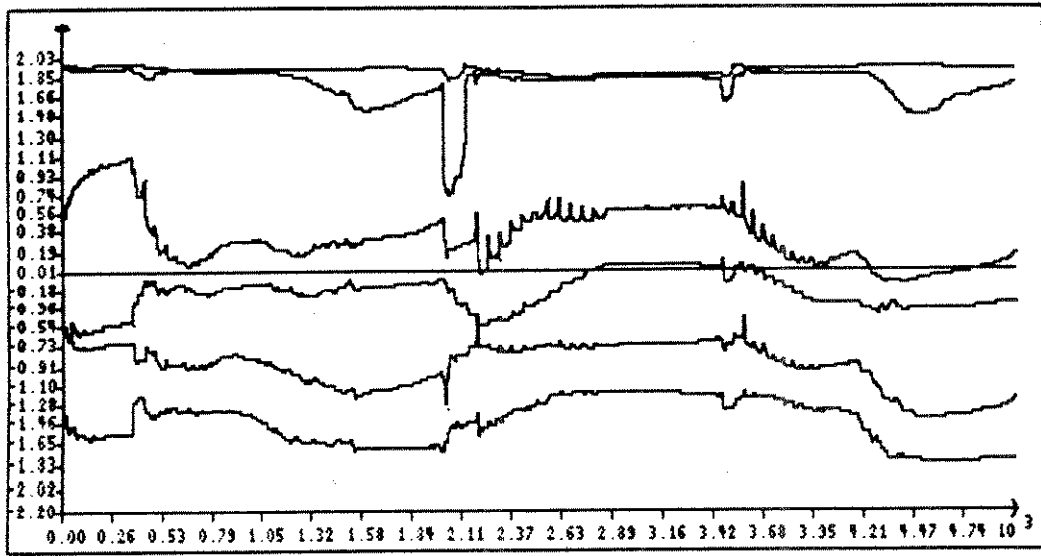


(b) $W = 0.98$

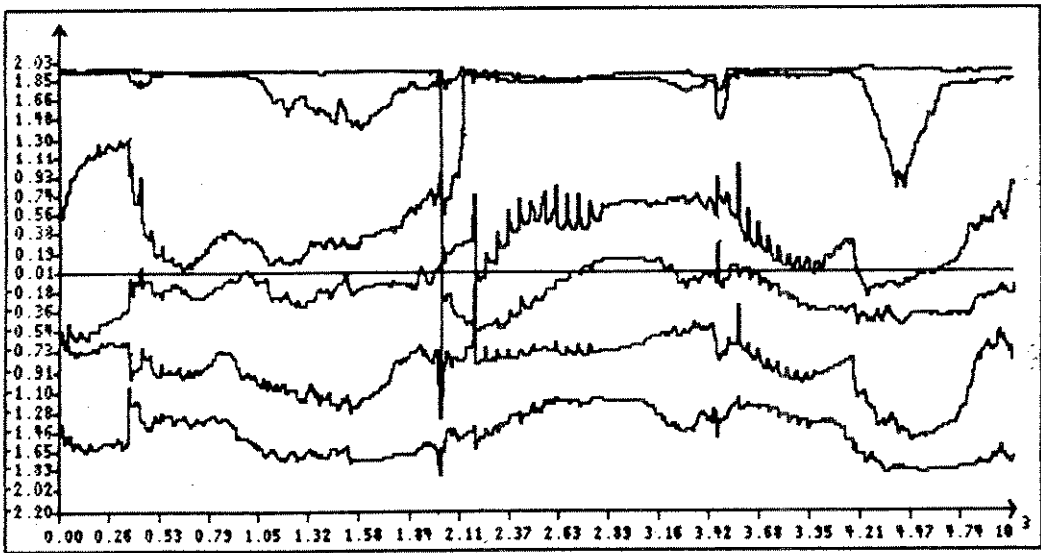


(c) Por Levinson-Durbin

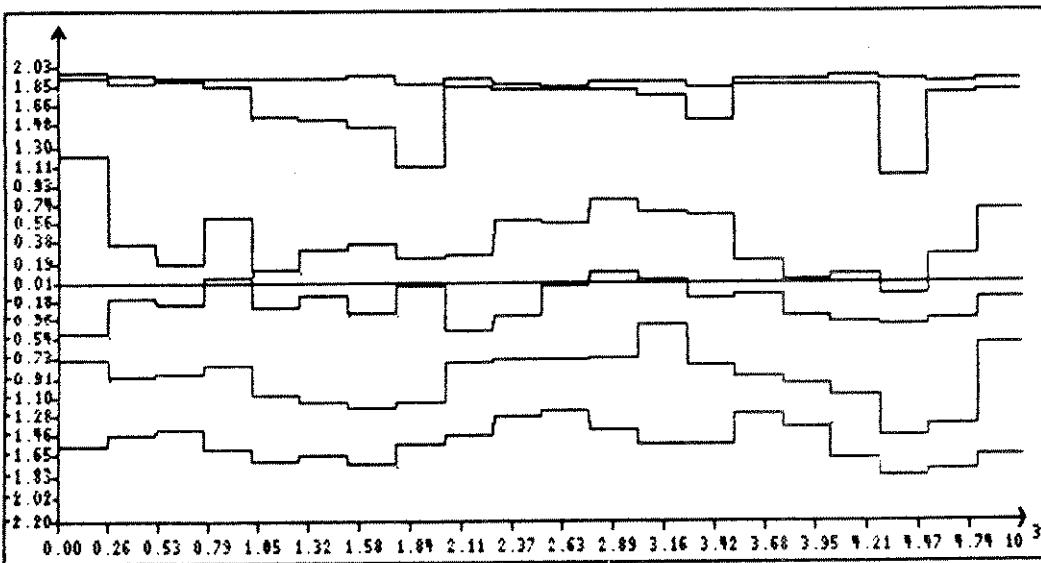
Fig. 5.4 - Evolução dos coeficientes LSP para 2 secções



(a) $W = 0.99$



(b) $W = 0.98$



(c) Por Levinson-Durbin

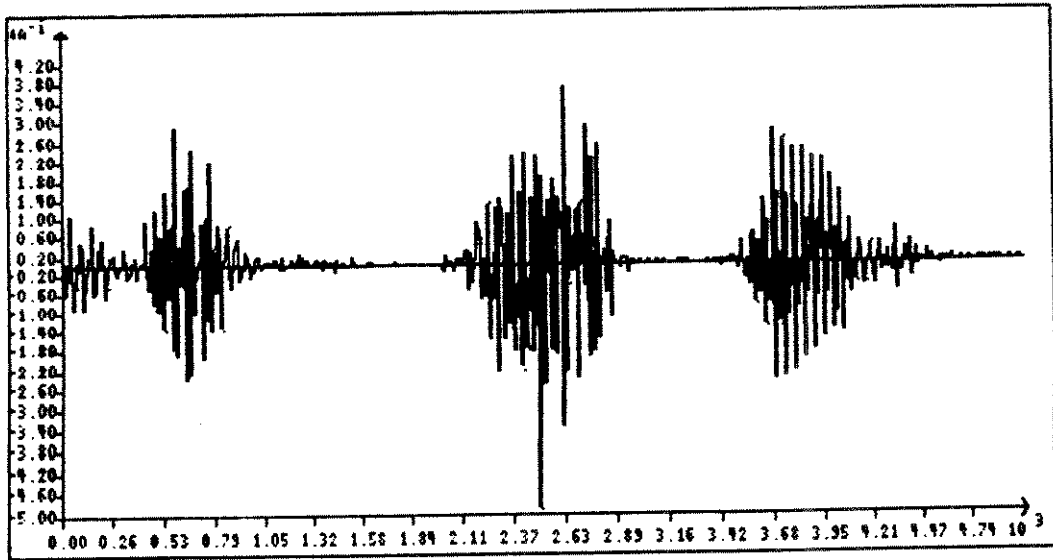
Fig. 5.5 - Evolução dos coeficientes LSP para 3 secções

Comparando-se as curvas, é possível constatar o potencial de rastreamento do algoritmo, diante do comportamento não-estacionário do sinal de voz, mostrando a boa dinâmica da estimação.

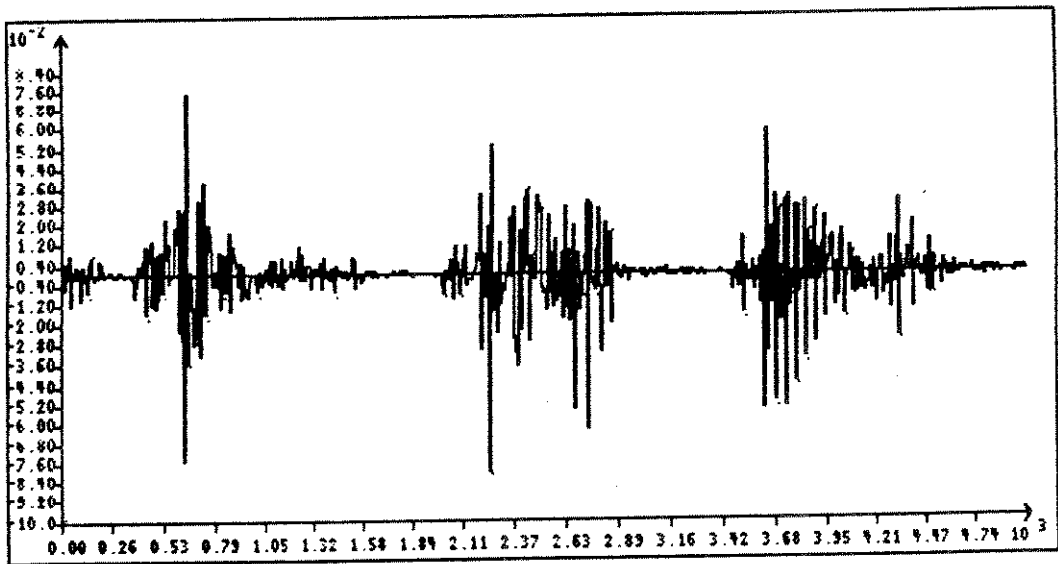
Uma outra característica importante diz respeito à estabilidade de sistema. Como visto, pode-se demonstrar que os parâmetros LSP correspondem a um modelo estável quando os zeros de $P(z)$ e $Q(z)$ em 5.3 e 5.4 estão dispostos alternadamente sobre o círculo unitário. Essa propriedade foi verificada durante toda a adaptação, sem que fosse necessário algum controle adicional. Assim, as condições iniciais foram escolhidas de forma a se ter os zeros de $P(z)$ e $Q(z)$ distribuídos uniformemente sobre o círculo unitário e respeitando a alternância.

As Figuras 5.6.a, b e c mostram respectivamente, o sinal de voz, o sinal de saída do erro e o erro quadrático para o caso citado da Figura 5.4.b (2 secções). É notória a capacidade de rastreamento evidenciada na comparação do erro com o sinal de entrada, além da busca da minimização do erro mostrada na Figura 5.6.c. Resumindo, a Figura 5.6 mostra, também, o bom desempenho do esquema proposto, em termos de ganho de predição. A Figura 5.7 mostra o desempenho para o caso de três secções (Figura 5.5.b). Pode-se observar que os valores absolutos do erro de predição diminuem sensivelmente com o aumento da ordem do preditor.

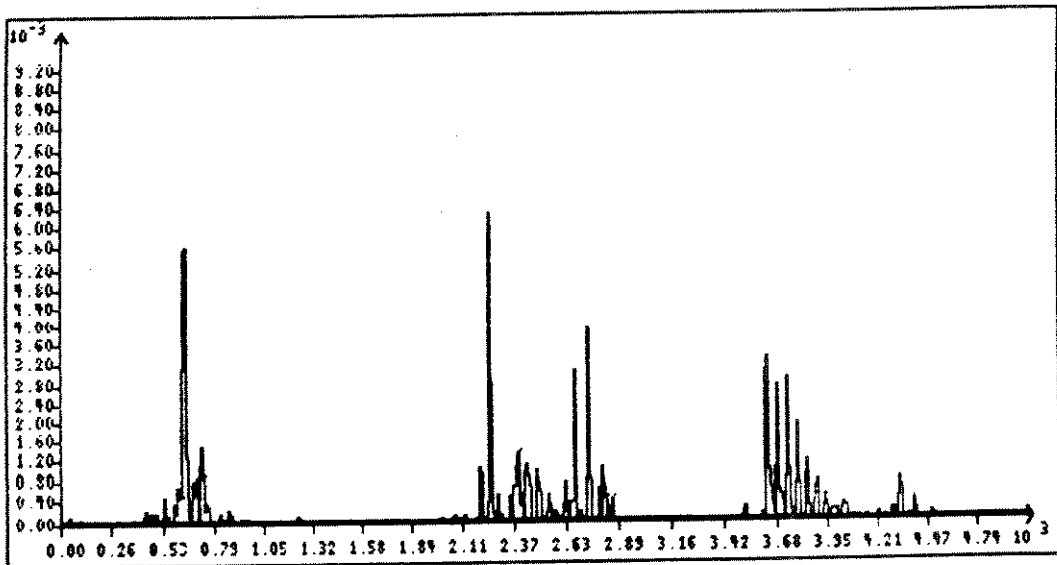
O desempenho do algoritmo foi confirmado em outros exemplos, onde utilizou-se diferentes arquivos de voz com resultados de simulação análogos. Um outro trecho de sinal de voz é mostrado na Figura 5.8 (sinal, erro, erro quadrático) para duas secções e, na Figura 5.9, para três secções. Os coeficientes desses casos são mostrados nas Figuras 5.10 e 5.11, para duas e três secções. Novamente, percebe-se uma diminuição nos valores do erro com o aumento do número de secções.



(a) Sinal de Voz

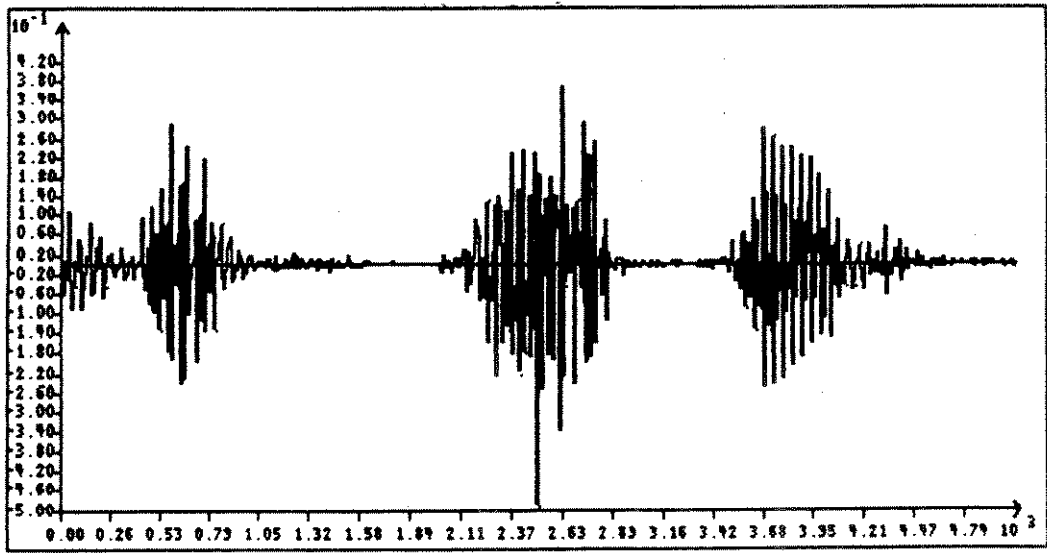


(b) Erro

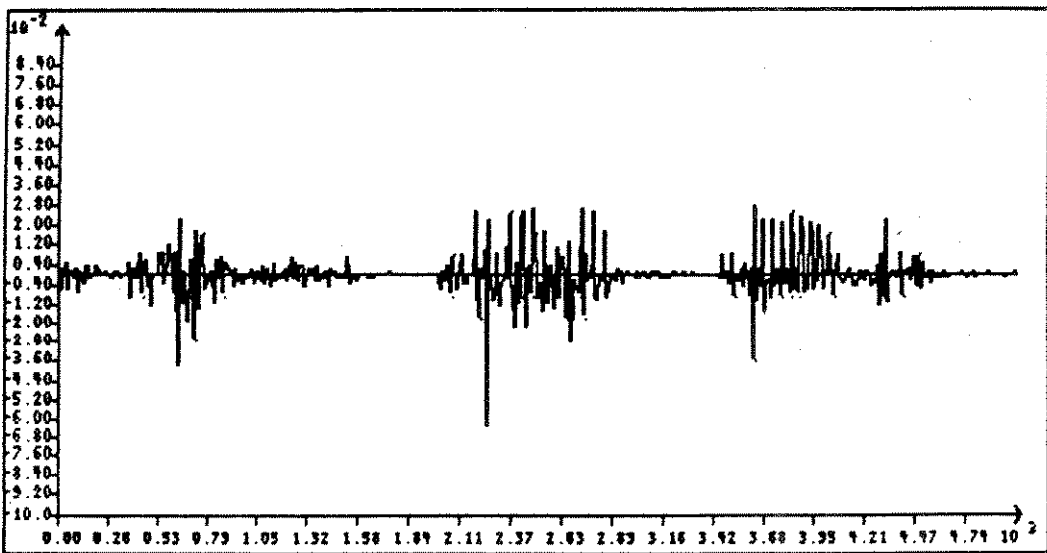


(c) Erro Quadrático

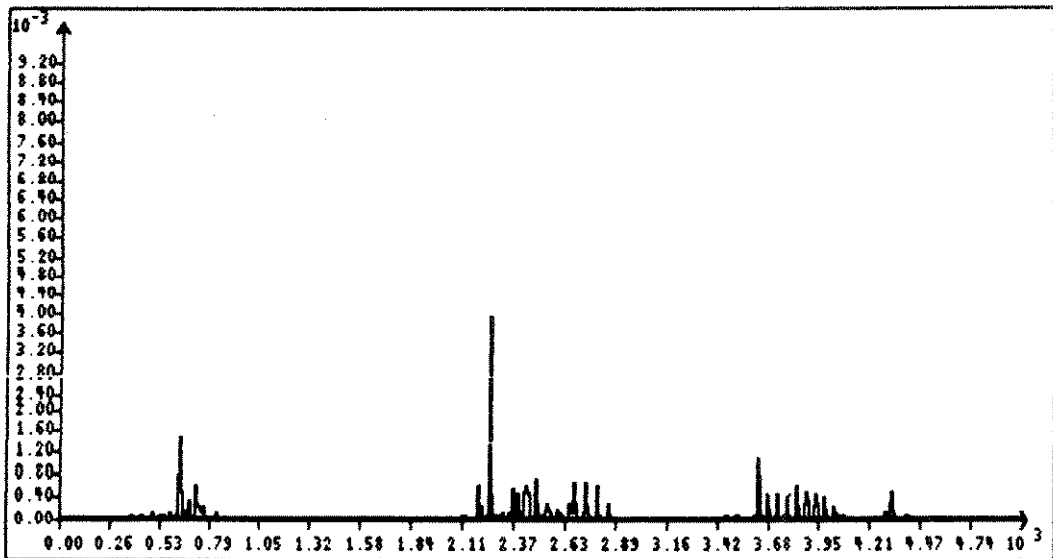
Fig. 5.6. - Trecho do Sinal de Voz Utilizado (2 secções)



(a) Sinal de Voz

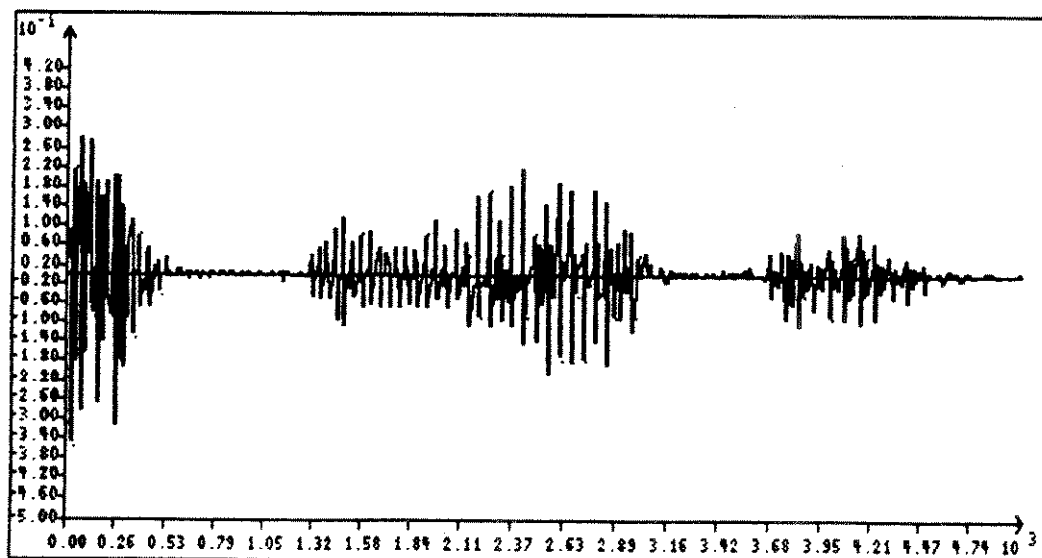


(b) Erro

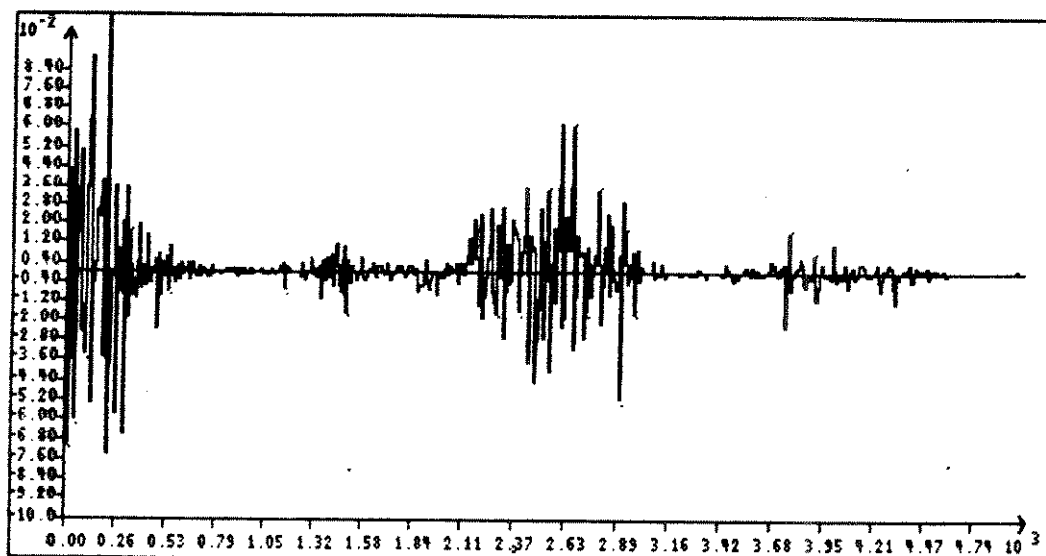


(c) Erro Quadrático

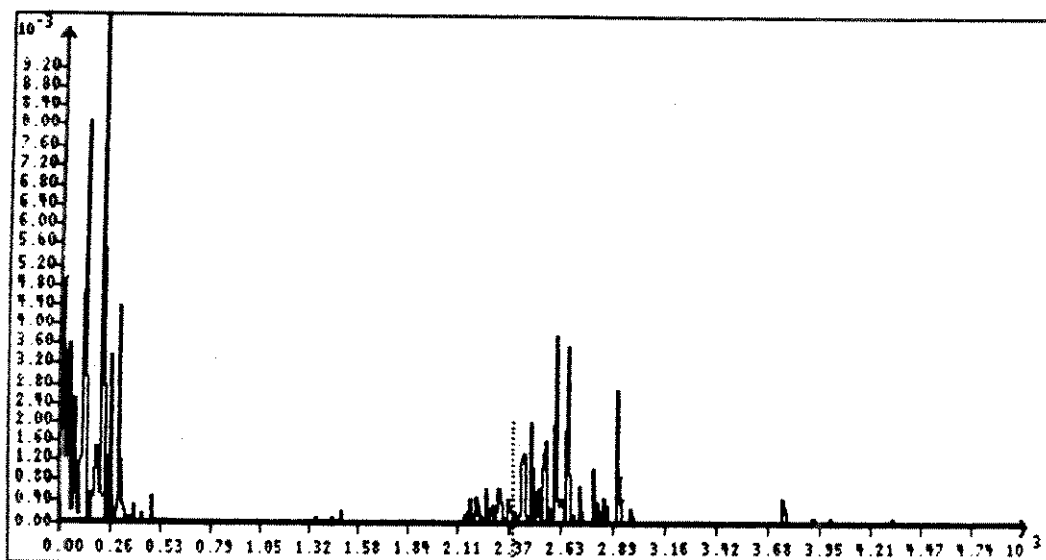
Fig. 5.7 - Trecho do Sinal de Voz Utilizado (3 secções)



(a) Sinal de Voz

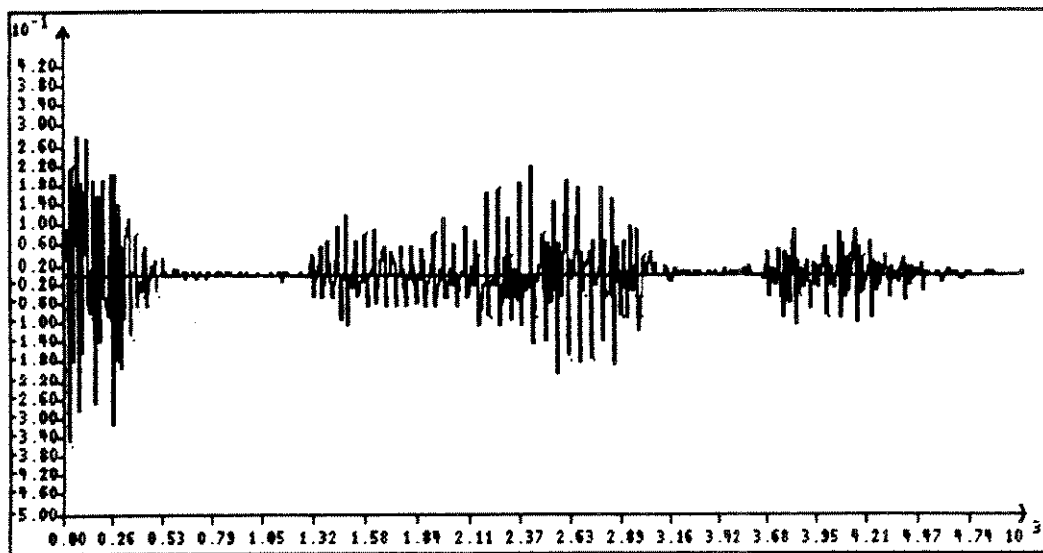


(b) Erro

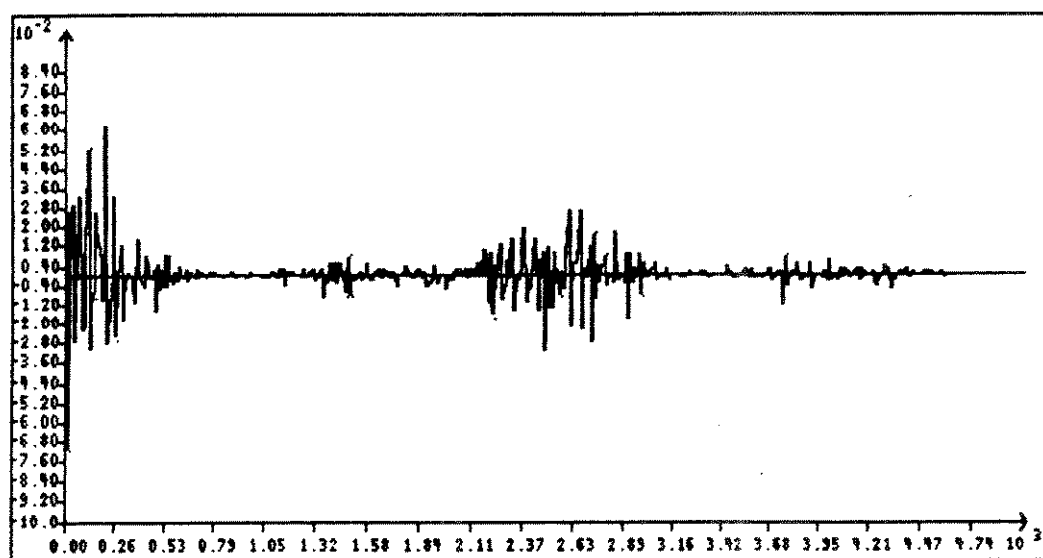


(c) Erro Quadrático

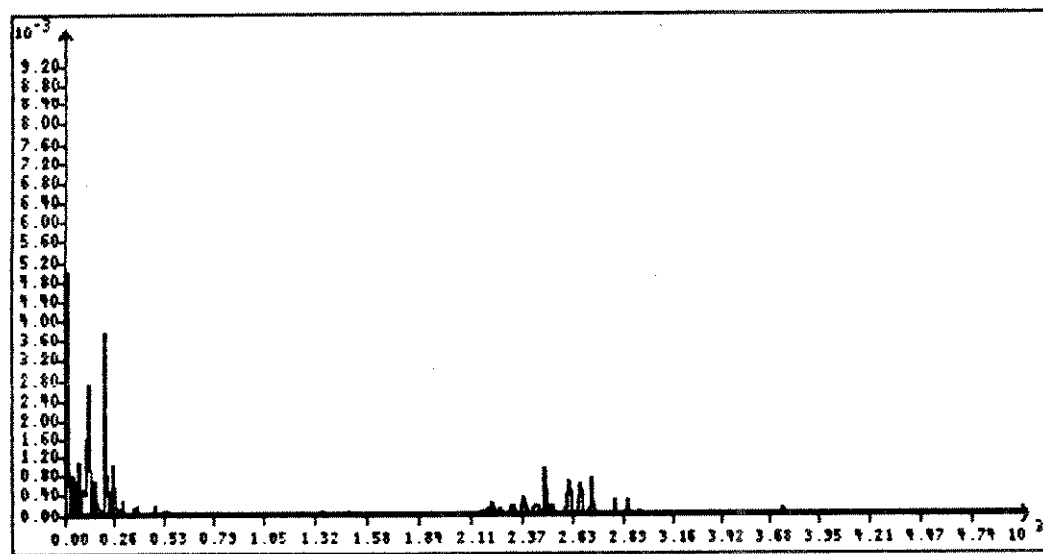
Fig. 5.8 - Trecho de um 2º Arquivo de Voz (2 secções)



(a) Sinal de Voz

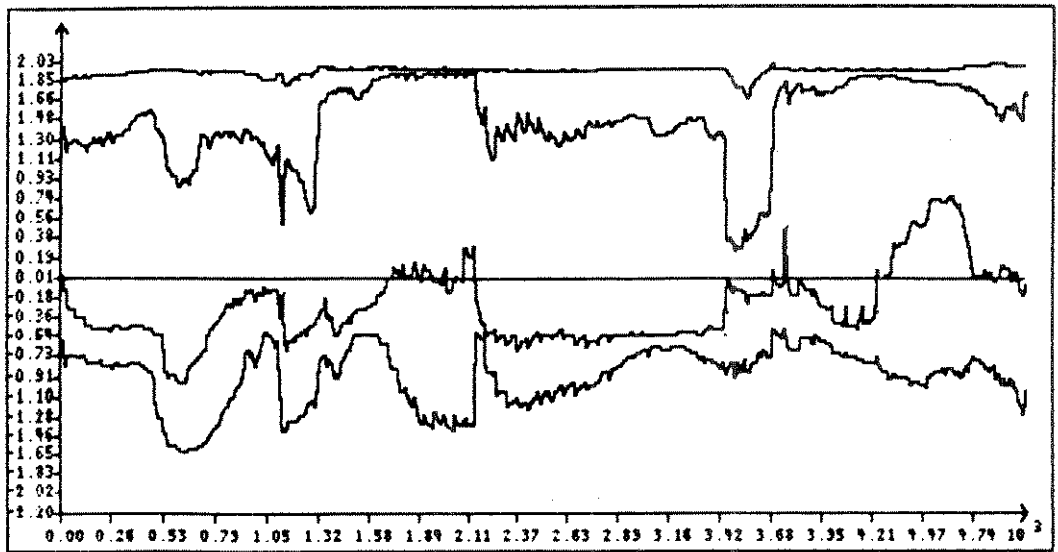


(b) Erro

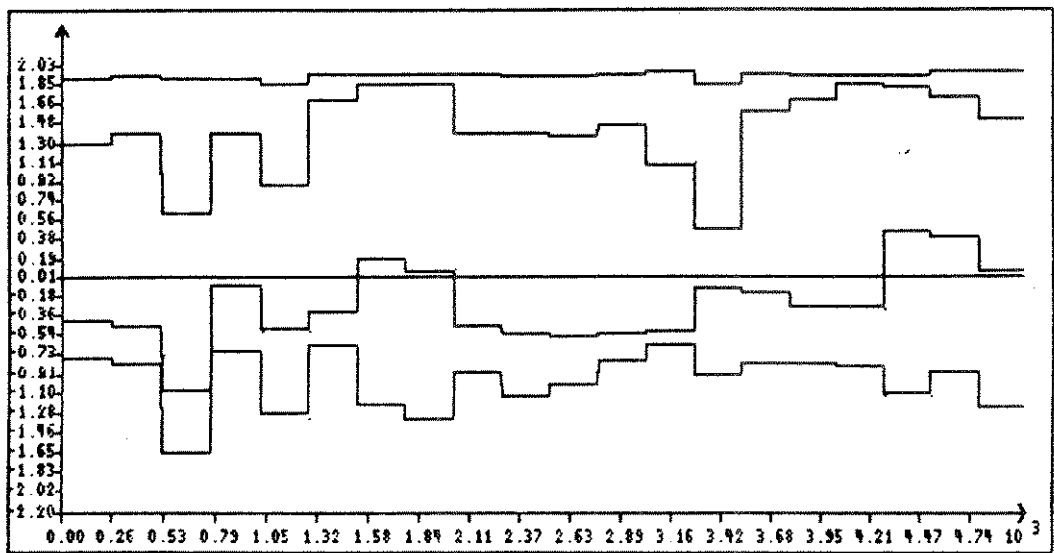


(c) Erro Quadrático

Fig. 5.9 - Trecho de um 2º Arquivo de Voz (3 secções)

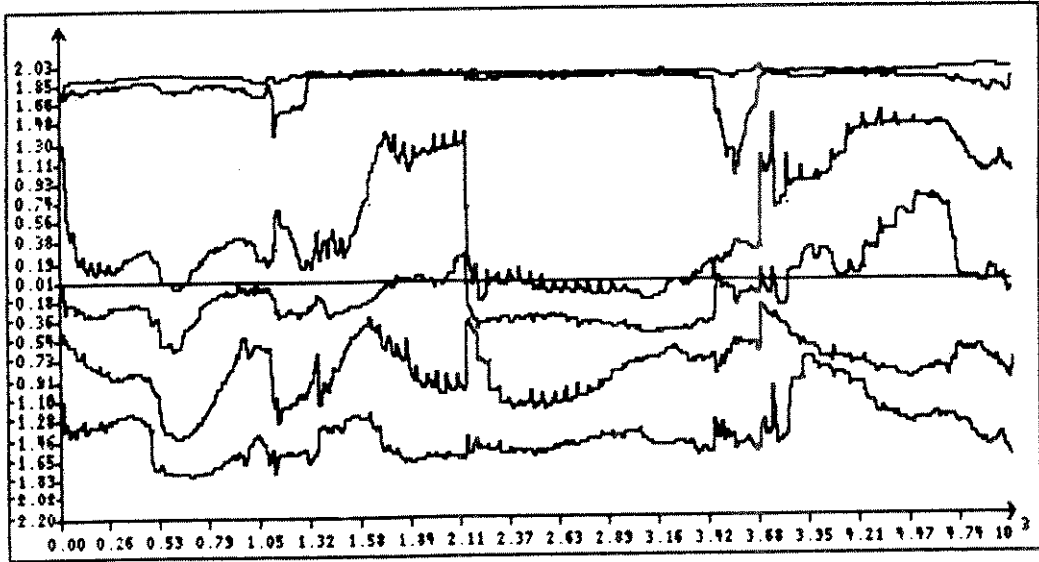


(a) $W = 0.98$

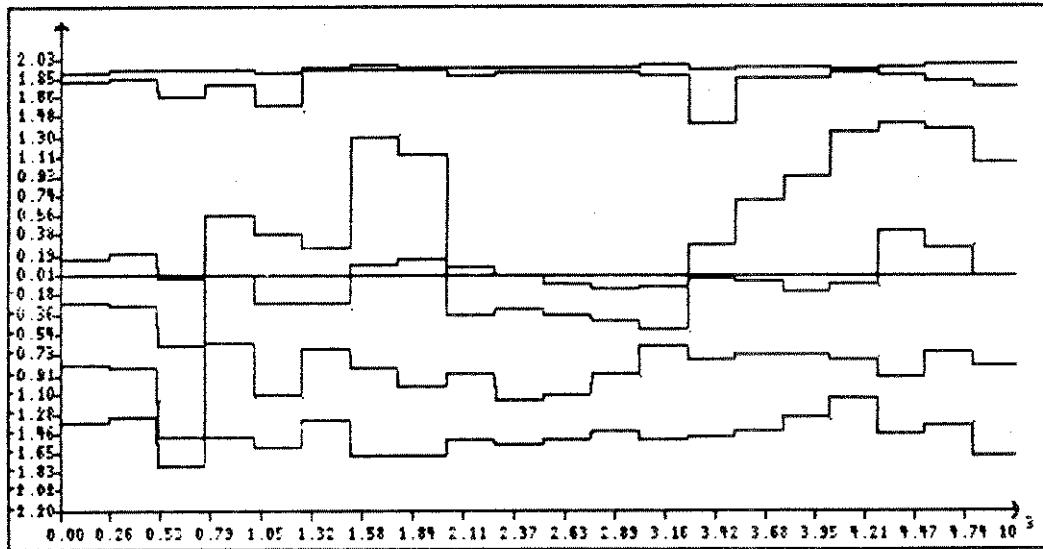


(b) Por Levinson-Durbin

Fig. 5.10 - Coeficientes LSP (2º arquivo de voz) para duas Secções



(a) $W = 0.98$



(b) Por Levinson-Durbin

Fig. 5.11 - Coeficientes LSP (2º arquivo de voz) para três Secções

5.4. CONCLUSÕES

As técnicas adaptativas, com sua respectiva análise de convergência, propostas ao longo do trabalho, encontram neste capítulo uma interessante aplicação numa área importante em processamento de sinais. Uma nova contribuição também aparece, uma vez que foi necessário o estudo dos parâmetros LSP e suas relações com a predição linear, além do uso de uma estrutura que garantisse a propriedade de fase linear. Dessa forma, a Figura 5.3 sintetiza claramente a proposta deste capítulo.

O emprego de uma cascata de secções de 2ª ordem permite a obtenção direta dos parâmetros LSP evitando cálculos polinomiais. Além disso, o algoritmo adaptativo permite que esses parâmetros sejam fornecidos a cada iteração, sem fazer uso de estimativas de correlação por bloco de dados. Chega-se assim a um método eficiente e de complexidade aceitável para análise ou codificação de voz em tempo real, como mostrado nos resultados das simulações.

As simulações mostraram, também, o comportamento estável da estrutura proposta, sem necessidade de controle adicional para atender as propriedades discutidas por Schüssler [45].

CAPÍTULO 6

CONCLUSÕES

Este trabalho foi baseado em estudos preliminares feitos por Jackson e Wood, no que se refere a predição linear utilizando estruturas na forma em cascata com aplicações a voz, e posteriormente por Bellanger e Romano, introduzindo algoritmos "on line" à aplicação de localização de frequências. Segundo Jackson e Wood, a possibilidade da determinação direta dos parâmetros do sinal de voz, como: pitch, formantes, etc., tornavam mais interessantes essas estruturas ao invés da forma tradicional FIR. No caso de localização de frequências por predição adaptativa, é notório o interesse em se obter, sem necessidade de cálculos polinomiais, os zeros do filtro.

As vantagens das estruturas em cascata de secções de segunda ordem, no que diz respeito a aspectos de implementação, também já foram citadas na literatura de processamento de sinais. A implementação de estruturas adaptativas FIR possui as conhecidas vantagens de estabilidade e convergência sobre as IIR, sendo que os avanços da Tecnologia VLSI tendem a reduzir o problema da complexidade computacional.

Neste trabalho, foi apresentada uma proposta conjunta de algoritmo e estrutura para filtragem adaptativa em cascata, complementada pelo estudo de aspectos tais como complexidade, velocidade de convergência e aplicações em detecção de frequência e em modelagem LSP. Um destaque é dado à análise de convergência do algoritmo, incluindo o estudo de seu comportamento assintótico e das curvas de erro quadrático.

A idéia básica partiu de se alterar a estrutura em cascata de forma a se apresentar uma estrutura opcional multidimensional com filtros FIR de segunda ordem. Para isso, uma aproximação é utilizada de maneira a superar a não linearidade decorrente do fato dos coeficientes dessa nova estrutura não serem relacionados como uma regressão linear dos diversos estados do novo filtro.

A primeira contribuição significativa deste trabalho está na forma de apresentação dos conceitos de filtragem adaptativa, abordados paralelamente às

técnicas clássicas de otimização. Todos os itens consistentes da filtragem adaptativa, tais como o critério de minimização, o processo de adaptação, o processo de filtragem e a forma de introdução dos sinais que compõem o sistema, foram tratados utilizando-se estrutura direta FIR.

No Capítulo 3, foi desenvolvida a proposta da filtragem em cascata, desde a visão de filtragem ótima e suas nuances, até chegar à estrutura adaptativa multidimensional, utilizando o algoritmo rápido FLS-MD. Nessa ótica, a aplicação do método se mostrou bastante eficiente na determinação de senóides na presença de ruído, por predição linear. Essa aplicação é de grande importância em diversas áreas de processamento de sinais, tais como sinalização telefônica, sonar e radar, sinais biomédicos e geofísicos, assim como em análise espectral em geral.

Nesse sentido, os resultados das simulações mostraram a grande vantagem das estruturas em cascata com relação à forma direta, inclusive com o algoritmo do gradiente, onde o fato de se trabalhar com estágios de segunda ordem reduz o espalhamento da matriz de autocorreção, aumentando assim a velocidade de convergência dos algoritmos. Quanto ao algoritmo dos mínimos quadrados, este se mostrou rápido e de convergência estável, embora a aproximação na utilização da recursão da matriz de intercorrelação, feita para sanar os problemas da não linearidade, implicou em uma certa instabilidade no algoritmo no início do processo. A proposta de um algoritmo modificado, utilizando também um passo de adaptação, corrigiu esse problema sem grandes perdas na velocidade de convergência.

A análise de convergência foi feita a partir da curva do erro médio quadrático, inserindo esses filtros dentro de uma classe mais genérica de filtros alternativos, os quais mantêm uma relação sobrejetora com estruturas na forma direta. Essa relação, de acordo com os trabalhos de Nayeri e Forssén, garante a manutenção das relações entre os pontos extremos das curvas de erro, desde que a forma direta tenha curva de erro com comportamento regular (ordem suficiente).

Uma aplicação de particular interesse se dá com a modelagem de sinais de voz por pares de raias espectrais, a partir dos trabalhos iniciais de Itakura. Foi proposta uma estrutura singela para essa aplicação, de forma adaptativa, fazendo com que a estrutura multidimensional se encaixasse, nesse caso, perfeitamente à modelagem de voz. Para isso, foram utilizados trechos de voz de forma a se mostrar a alta capacidade de rastreamento do algoritmo proposto.

Como sugestões de trabalhos futuros, pode-se trabalhar em casos específicos de identificação, controle, análise espectral, etc., já comentados ao longo do trabalho, onde a obtenção adaptativa dos zeros do filtro é importante. A extensão dos resultados para filtros IIR em cascata, além de outras estruturas alternativas, surge também como um tema de grande interesse para aprofundar o estudo teórico e ampliar a gama de aplicações em processamento adaptativo.

BIBLIOGRAFIA

- [1] S. Haykin, Adaptive Filter Theory, Prentice Hall, New Jersey, 1986.
- [2] B. Widrow and S.D. Stearns, Adaptive Signal Processing, Prentice Hall, New Jersey, 1985.
- [3] C. Cowan and P. Grant, Adaptive Filters, Prentice Hall, New Jersey, 1985.
- [4] J.M. Cioffi, "Limited Precision Effects in Adaptive Filtering", *IEEE Trans. CAS* - (1987).
- [5] L. Ljung and T. Söderström, Theory and Practice of Recursive Identification, MIT-Press, 1983.
- [6] C.R. Johnson Jr., "Adaptive IIR Filtering: Current Results and Open Issues", *IEEE Transactions on Information Theory*, vol. IT-30, nº 2, March - 1984, pp. 237-250.
- [7] L.B. Jackson and S.L. Wood, "Linear Prediction in Cascade Form", *IEEE Trans. on ASSP*, vol. 26, nº 6, December-1978, pp. 518-528.
- [8] A.V. Oppenheim and R.W. Schaffer, Digital Signal Processing, Prentice-Hall, Englewood Cliffs, New Jersey, 1975.
- [9] J.D. Markel and A.H. Gray Jr., Linear Prediction of Speech, Springer-Verlag, 1976.
- [10] L.R. Rabiner, and R.W. Schaffer. Digital Processing of Speech Signals, Prentice-Hall, Englewood Cliffs, New Jersey, 1978.
- [11] J.M.T. Romano et M. Bellanger, "Póles et Zéros des Prédicteurs Numériques", *Traitement du Signal*, vol. 3, nº 2, 1986.
- [12] S.M. Kay and S.L. Marple Jr., "Spectrum Analysis - a Modern Perspective", *Proceedings of the IEEE*, vol. 69, nº 11, Nov. 1981, pp. 1380-1419.
- [13] J.M.T. Romano et M. Bellanger, "Algorithm MCR pour Filtre Adaptatif en Cellule du Second Ordre", *Onzième Colloque GRETSI*, Nice, Juin - 1987, pp. 293-296.
- [14] L.C. Coradine e J.M.T. Romano, "Filtragem Adaptativa em Cascata com Aplicações em Modelos LSP", *Relatório Final*, Contrato UNICAMP-TELEBRÁS nº 987/90, Pub. agosto/1992.
- [15] F. Itakura, "Line Spectrum Representations of Linear Predictive Coefficients of Speech Signals", *J. of Acoustical Society of America*, vol. 57, 1975, pp. 535.
- [16] P. Kabal and R.P. Ramachandran, "The Computation of Line Spectral Frequencies Using Chebyshev Polynomials", *IEEE Trans. on ASSP*, vol. 34, nº 6, December 1986, pp. 1419-1426.
- [17] B.M.G. Cheethan, "Adaptive LSP Filter", *Electronics Letters*, vol. 23, nº 2, January 1987, pp. 89-90.

- [18] M. Nayeri and W.K. Jenkins, "Alternative Realizations to Adaptive IIR Filters and Properties of Their Performance Surfaces", *IEEE Transactions on Circuits and Systems*, vol. CAS 36, n° 4, April 1989, pp. 485-496.
- [19] U. Forssén and B. Gudmundson, "Adaptive Algorithms for Arbitrary Filter Structures", *URSI*, 1989, pp. 466-471.
- [20] H.L. Van Trees, Detection, Estimation and Modulation Theory, Part I, Wiley, New York, 1968.
- [21] A.A. Giordano and F.M. Hsu - Least Square Estimation with Applications to Digital Signal Processing, John Wiley, New York, 1985.
- [22] D.G. Luenberger, Introduction to Linear and Nonlinear Programming, Addison-Wesley, MA, 1973.
- [23] M. Bellanger, Adaptive Digital Filters and Signal Analysis, Marcel Dekker, New York, 1987.
- [24] B.D.O. Anderson and J.B. Moore - Linear Optimal Control, Prentice-Hall, New Jersey, 1979.
- [25] J. Makhoul, "Linear Prediction: A Tutorial Review", *Proc. IEEE*, vol. 62, April 1975, pp. 561-580.
- [26] G.C. Goodwin and R.L. Payne, Dynamic System Identification: Experiment Design and Data Analysis, Academic Press, New York, 1977.
- [27] L. Ljung, "Analysis of Recursive Stochastic Algorithms", *IEEE Trans. Automatic Control*, vol. AC 22, pp. 551-575, 1977.
- [28] S. Ljung, M. Morf and D. Falconer, "Fast Calculation of Gain Matrices for Recursive Estimation Schemes", *Int. J. Control*, 1978, vol. 27, n° 1, pp. 1-19.
- [29] G. Carayannis, D. Monolakis and N. Kalouptsidis, "A Fast Sequential Algorithm for Least-Square Filtering Prediction", *IEEE Trans. on ASSP*, vol. 31, n° 6, Dec. 1983, pp. 1394-1402.
- [30] J. Cioffi and T. Kailath, "Fast Recursive Least Squares Transversal Filters for Adaptive Filtering", *IEEE Trans. on ASSP*, vol. 32, n° 2, April 1984, pp. 304-337.
- [31] S. Ljung and L. Ljung, "Error Propagation Properties of Recursive Least Squares Adaptation Algorithms", *Automatica*, vol. 21, n° 2, 1985, pp. 157-167.
- [32] J.L. Botto and G.V. Moustakides, "Stabilizing the Fast Kalman Algorithms", *IEEE Trans. on ASSP*, vol. 37, n° 9, September 1989, pp. 1342 - 1348.
- [33] D.T.M. Slock and T. Kailath, "Numerically Stable Fast Transversal Filters for Recursive Least Squares Adaptive Filtering", *IEEE Transactions on Signal Processing*, vol. 39, n° 1, January 1991, pp. 92-113.

- [34] L.C. Coradine e J.M.T. Romano, "Filtragem Adaptativa na Forma em Cascata: Proposta de Estrutura e do Algoritmo Rápido FLS-MD". *11º Simpósio Brasileiro de Telecomunicações*, Natal - RN, Set. 1993.
- [35] M.G. Larimore *et al.*, "SHARF: An Algorithm for Adapting IIR Digital Filters", *IEEE Trans. ASSP*, vol. ASSP-28, 1980, pp. 428-440.
- [36] H. Fan and W.K. Jenkins, "A New Adaptive IIR Filter", *IEEE Transactions on Circuits and Systems*, vol. CAS 33, nº 10, October 1986, pp. 939-947.
- [37] J.J. Shynk, "Adaptive IIR Filtering Using Parallel-Form Realizations", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-37, nº 4, April 1989, pp. 519-533.
- [38] S.D. Stearns, "Error Surfaces of Recursive Adaptive Filters", *IEEE Transactions on Circuits and Systems*, vol. CAS-28, nº 6, June 1981, pp. 603-606.
- [39] T. Söderström and P. Stoica, "Some Properties of the Output Error Method", *Automatica*, vol. 18, 1982, pp. 93-99.
- [40] M. Nayeri, "Improvements in Adaptive IIR Filtering Theory and Application", Ph.D. dissertation, Dep. Elect. Comp. Eng. Univ. Illinois, Urbana, 1988.
- [41] U. Forssén, "Adaptive Filters with Arbitrary Structures, Part 2: Some Asymptotic Properties", Report nº TRITA-TTT - 8915, Royal Inst. of Tech., Stockholm, Sweden, Dec. 1989.
- [42] K. Hosoda and A. Fukasawa, "32 kbps ADPCM CODEC Composed by the Prediction Filter Including Poles and Zeros", *Proc. EUSIPCO - 83*, Elsevier, 1983, pp. 391-394.
- [43] J.M.T. Romano, M. Bellanger and L.C. Coradine, "Least Squares Adaptive in Cascade for Line Pairs Spectrum Modelling", *Fifth European Signal Processing Conference (EUSIPCO 90)*, Barcelona, September 1990, L23-4.
- [44] J.M.T. Romano e L.C. Coradine, "Uma estrutura Adaptativa para Modelagem de Voz por Pares de Linhas Espectrais (LSP) Utilizando o Algoritmo FLS-MD", *11º Simp. Bras. de Telecomunicações*, Natal - RN, Set. 1993.
- [45] H.W. Schussler, "A Stability Theorem for Discrete Systems", *IEEE Trans. on ASSP-24*, 1976.
- [46] S. Sagayama and F. Itakura, "Duality Theory of Composite Sinusoidal Modelling and Linear Prediction", *Proc. of ICASSP-86*, Tokyo 1986, pp. 1261-1264.
- [47] G. Carayannis, D. Manolakis and N. Kalouptsidis, "A Unified View of Parametric Processing Algorithms for Prewindowed Signals", *Signal Processing 10*, 1986, pp. 335-368.