



UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO  
DEPARTAMENTO DE COMUNICAÇÕES

## Desenvolvimento e Análise de Novos Algoritmos Preditivos para Equalização Cega

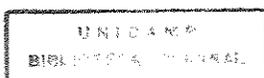
Este trabalho foi submetido à defesa final da tese  
defendida por Renato da Rocha Lopes  
e aprovada pela Comissão  
Julgadora em 20 / 06 / 87.  
João Marcos Travassos Romano  
Orientador

Autor: Renato da Rocha Lopes

Orientador: João Marcos Travassos Romano

Dissertação submetida à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos exigidos para a obtenção do Título de Mestre em Engenharia Elétrica.

Campinas, junho de 1997



UNIDADE	BC
N.º CHAMADA	L881d
V.	Es.
TOMBO BC	31329
PROC.	281/97
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	R\$ 11,00
DATA	13/02/97
N.º CPD	

CM-0005585-1

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

L881d      Lopes, Renato da Rocha  
Desenvolvimento e análise de novos algoritmos  
preditivos para equalização cega / Renato da Rocha  
Lopes.--Campinas, SP: [s.n.], 1997.

Orientador: João Marcos Travassos Romano  
Dissertação (mestrado) - Universidade Estadual de  
Campinas, Faculdade de Engenharia Elétrica e de  
Computação.

1. Filtros adaptativos. 2. Comunicações digitais. 3.  
Algoritmos.. I. Romano, João Marcos Travassos. II.  
Universidade Estadual de Campinas. Faculdade de  
Engenharia Elétrica e de Computação. III. Título.

# Sumário

<b>Agradecimentos</b>	<b>v</b>
<b>Resumo</b>	<b>vi</b>
<b>Abstract</b>	<b>vii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Equalização de Canais em Transmissão Digital . . . . .	1
1.2 Novos Algoritmos para Técnicas Preditivas . . . . .	2
1.3 Análise de Convergência . . . . .	3
1.4 Resultados de Simulações . . . . .	4
<b>2 Equalização de Canais em Transmissão Digital</b>	<b>5</b>
2.1 Sistemas de Transmissão Digital . . . . .	5
2.2 Distorções Provocadas pelos Canais de Comunicações . . . . .	6
2.2.1 Representação Discreta do Canal de Comunicações . . . . .	7
2.2.2 Interferência Entre Símbolos . . . . .	8
2.3 Equalizadores . . . . .	9
2.3.1 Técnicas Clássicas de Equalização . . . . .	10
2.3.2 Equalizadores Adaptativos . . . . .	11
2.4 Equalizadores Cegos . . . . .	16
2.4.1 Técnicas de Bussgang . . . . .	18
2.5 Técnicas Preditivas . . . . .	19
2.5.1 Predição e Retropredição . . . . .	21
2.5.2 Equalização de Magnitude e Fase . . . . .	23
2.6 Conclusão . . . . .	25
<b>3 Novos Algoritmos para Técnicas Preditivas</b>	<b>26</b>
3.1 Os Algoritmos RLS . . . . .	26
3.1.1 Equalizador de Fase . . . . .	27
3.1.2 Ajuste de Fase . . . . .	31
3.1.3 Preditor IIR . . . . .	32
3.2 Algoritmos de Gauss-Newton . . . . .	37
3.2.1 Equalizador de Fase . . . . .	38
3.2.2 Equalizador de Magnitude . . . . .	39
3.3 Conclusão . . . . .	40

<b>4</b>	<b>Análise de Convergência</b>	<b>41</b>
4.1	Análise da Função Custo $J_\varepsilon(H)$	41
4.1.1	Justificativa para a Escolha de $J_\varepsilon(H)$	41
4.1.2	Unimodalidade de $J_\varepsilon(H)$	45
4.2	Análise dos Algoritmos Preditivos	47
4.2.1	O Método da ODE para Algoritmos de Ganho Constante	48
4.2.2	“Explicação Intuitiva” para o Método da ODE	50
4.2.3	Funções de Lyapunov	51
4.2.4	Análise do Algoritmo LMS	51
4.2.5	O Método da ODE de Ljung	53
4.2.6	Análise do Algoritmo GN	54
4.3	Conclusão	56
<b>5</b>	<b>Resultados de Simulações</b>	<b>58</b>
5.1	Canal Passa-Tudo	58
5.2	Canal Rádio-Móvel: Ambiente Estacionário	59
5.3	Canal de Microondas	62
5.4	Canal Rádio-Móvel: Ambiente Não-Estacionário	63
5.5	Conclusão	65
<b>6</b>	<b>Conclusões</b>	<b>69</b>
	<b>Referências Bibliográficas</b>	<b>72</b>

# Lista de Figuras

2.1	Sistema de Comunicações . . . . .	5
2.2	Representação Discreta de um Sistema de Comunicações . . . . .	8
2.3	Estrutura do Filtro Transversal Linear . . . . .	10
2.4	Filtro de Erro de Predição IIR . . . . .	21
2.5	Equalizador Baseado em um Retropreditor e um Preditor . . . . .	22
2.6	Equalizadores de Magnitude e Fase em Cascata . . . . .	24
4.1	Representação Alternativa do Equalizador de Fase. . . . .	42
4.2	Estimativa da Função de Autocorrelação de $\{\varepsilon(n)\}$ . . . . .	45
4.3	Estimativa da Correlação Cruzada Normalizada entre $\{s(n)\}$ e $\{\varepsilon(n)\}$ (Ref. [1]). . . . .	45
4.4	Estimativa da Densidade (a) e Distribuição (b) de Probabilidade de $y(n)$ , Comparados com uma Distribuição Gaussiana. . . . .	46
5.1	<i>Learning Curves</i> para o Canal Passa-Tudo . . . . .	59
5.2	Localização dos Zeros do Canal GSM . . . . .	60
5.3	<i>Learning Curves</i> para o Canal Rádio-Móvel Estacionário . . . . .	61
5.4	Evolução dos Valores Absolutos dos Coeficientes do Preditor IIR Adaptados pelos Algoritmos LMS (a) e GN (b). . . . .	61
5.5	Resposta em Freqüência do Canal de Microondas . . . . .	62
5.6	<i>Learning Curves</i> para o Canal de Microondas . . . . .	63
5.7	Modelo de Dois Raios do Canal Rádio-Móvel (Ref. [2]). . . . .	64
5.8	Potência da Saída do Canal Rádio-Móvel . . . . .	65
5.9	Evolução do Erro DD para o Canal Rádio-Móvel: Algoritmo LMS . . . . .	66
5.10	Evolução do Erro DD para o Canal Rádio-Móvel: Algoritmo GN . . . . .	67
5.11	Constelação na Saída do Equalizador: Algoritmo LMS . . . . .	67
5.12	Constelação na Saída do Equalizador: Algoritmo GN . . . . .	68

# Lista de Tabelas

3.1	Algoritmo RLS para o Filtro Passa-Tudo . . . . .	31
3.2	Algoritmo RLS para o Ajuste de Fase . . . . .	32
3.3	Algoritmo RLS para o Preditor IIR . . . . .	37
3.4	Algoritmo de Gauss-Newton para o Filtro Passa-Tudo . . . . .	39
3.5	Algoritmo de Gauss-Newton para o Preditor IIR . . . . .	40
5.1	Parâmetros Usados na Simulação do Canal Passa-Tudo . . . . .	59
5.2	Parâmetros Usados na Simulação do Canal Rádio-Móvel Estacionário . . . . .	60
5.3	Coefficientes da Resposta ao Impulso $h(n)$ do Canal de Microondas . . . . .	62
5.4	Parâmetros Usados na Simulação do Canal de Microondas . . . . .	63
5.5	Parâmetros Usados na Simulação do Canal Rádio-Móvel Não-Estacionário . . . . .	65

# Agradecimentos

Agradeço inicialmente aos meus pais, pelo apoio, pelos incentivos e, principalmente, pelo amor que eles me deram. Sem isto, este trabalho nunca teria se concretizado. Pela participação decisiva e sempre positiva em toda a minha vida, dedico a eles esta tese.

A Túria, minha futura esposa, pelo amor e compreensão nos momentos mais difíceis, e por ter sido sempre minha companheira. Que nós sejamos muito felizes.

Ao meu orientador, Prof. João Marcos Travassos Romano, pelas discussões esclarecedoras, pelas opiniões construtivas e por ter sido também um ótimo orientador da minha carreira.

Aos Profs. Dalton Soares Arantes, Carlos Aurélio Faria da Rocha e Michel Daoud Yacoub, por terem se disposto a participar da Comissão Julgadora deste trabalho, e por terem me apoiado e auxiliado em vários momentos.

À “galera” do Laboratório, pelo apoio, amizade, e por terem feito de nosso departamento um lugar de trabalho muito agradável.

Ao CNPq, pelo apoio financeiro dado a este trabalho.

Finalmente, gostaria de expressar meu agradecimento sincero a todos os professores, amigos e funcionários que, de maneira direta ou indireta, tornaram a execução deste trabalho possível e até mesmo prazerosa.

# Resumo

Nesta tese serão estudadas algumas técnicas de equalização cega para um canal de comunicações. Serão abordadas principalmente as técnicas que se baseiam na teoria de predição linear e que, por isto, são chamadas técnicas preditivas de equalização cega.

Inicialmente, será discutida a necessidade de um equalizador em um sistema de comunicações digitais, bem como as vantagens e desvantagens dos equalizadores cegos quando comparados aos equalizadores tradicionais. Em seguida, serão apresentadas algumas técnicas clássicas de equalização cega. Os princípios das técnicas preditivas de equalização cega serão, então, estudados e serão apresentados algoritmos adaptativos para os equalizadores preditivos. Estes algoritmos se baseiam no método do gradiente (LMS).

Após esta apresentação inicial, a técnica preditiva de equalização cega baseada na equalização de magnitude e fase será estudada em maior profundidade. Nesta técnica, um preditor IIR é usado para corrigir as distorções que o canal provoca na magnitude do espectro de frequência do sinal transmitido, enquanto que um filtro passa-tudo não linear é usado para corrigir as distorções de fase provocadas pela cascata do canal com o preditor IIR. Para esta estrutura, serão apresentados dois novos algoritmos, baseados nas técnicas de mínimos quadrados (RLS, do inglês *Recursive Least Squares*) e de Gauss-Newton (GN).

Será feito em seguida um estudo teórico do equalizador de fase e dos algoritmos adaptativos propostos para esta estrutura. Será provado que os critérios de minimização associados ao equalizador de fase são unimodais e que os algoritmos propostos de fato convergem para este mínimo.

Finalmente, simulações demonstrarão o bom desempenho desta técnica preditiva de equalização cega.

# Abstract

In this work, the blind equalization of a communication channel will be considered. The work will focus on the discussion of blind equalization techniques that are based on linear prediction theory. These are called predictive techniques for blind equalization.

Firstly, the need for an equalizer in a communication system will be discussed, as well as the advantages and disadvantages of blind equalizers as opposed to supervised ones. Some traditional blind equalization techniques will be presented. Then, the principles of predictive blind equalization techniques will be studied, and some adaptive algorithms for predictive equalizers will be presented. These algorithms are based on the steepest descent method (LMS).

After this introductory presentation, this work will focus on a predictive technique based on magnitude and phase equalization. In this technique, a linear IIR predictor is used to compensate for the magnitude distortion caused by the channel, and a non-linear all-pass filter is used to compensate for the phase distortion introduced by the cascade of the channel and the predictor. Two new adaptive algorithms for this structure will be presented, one based on the least-squares method (RLS, Recursive Least Squares) and another one based on the Gauss-Newton method (GN).

Some theoretical analysis of both the phase equalizer and the algorithms for this structure will then be carried on. It will be shown that the minimization criterion associated with the phase equalizer is unimodal. Furthermore, it will be shown that the proposed algorithms do converge to a point where they equalize the channel.

Finally, some simulation results will confirm the effectiveness of these algorithms.



# Capítulo 1

## Introdução

O objetivo deste trabalho é o estudo de técnicas preditivas de equalização cega, bem como a proposição de novos algoritmos para esta técnica. Para atingir este objetivo, o trabalho é dividido em 4 capítulos principais. Apresentaremos na seqüência um resumo destes capítulos.

### 1.1 Equalização de Canais em Transmissão Digital

Os sistemas de transmissão de informações digitais são parte integrante de nosso cotidiano. Os exemplos são vários: sistemas digitais de telefonia celular, *links* de microondas e de satélites, etc. Esta também é uma área em rápido desenvolvimento, e constantemente surgem novos sistemas digitais, haja vista a iminente implementação comercial da televisão digital de alta definição. Em todos estes sistemas, uma seqüência de bits ou símbolos é gerada em um terminal de transmissão e, após passar por um canal de comunicações, é captada no terminal de recepção. Para o bom desempenho destes sistemas é fundamental que a seqüência de bits transmitida possa ser corretamente recuperada no terminal de recepção.

Um dos fatores que dificultam esta recuperação é a interferência entre símbolos (IES). A resposta do canal a um símbolo tem, em geral, duração maior que a do próprio símbolo. Ou seja, quando o canal começa a responder à transmissão de um símbolo, ele ainda está respondendo à transmissão de símbolos anteriores, de forma que estas respostas se sobrepõem em sua saída, impossibilitando a identificação imediata do símbolo transmitido. Este fenômeno pode ser visto como conseqüência de uma filtragem linear do sinal transmitido.

Ora, se o filtro que gera a IES puder ser invertido, este fenômeno pode ser facilmente combatido. O filtro linear que busca combater a IES através da filtragem inversa é chamado equalizador. O objetivo do equalizador não é inverter exatamente o canal, mas sim diminuir os efeitos da IES de forma que o símbolo transmitido possa ser corretamente identificado por um dispositivo de decisão. As técnicas clássicas de equalização, tais como os algoritmos adaptativos LMS (do inglês *Least Mean Squares*) e GN (algoritmo de Gauss-Newton), buscam determinar os coeficientes do equalizador que minimizam a média quadrática do erro de estimação. Este erro é a diferença entre o símbolo transmitido e o valor estimado para este símbolo, dado pela saída do equalizador.

Temos aqui um aparente contrasenso. Para poder minimizar a função custo, os algo-

ritmos adaptativos devem ser capazes de calcular o erro de estimação. Ora, para isto, eles devem conhecer o símbolo transmitido. Entretanto, se este conhecimento é possível, não há a necessidade de um equalizador. Para resolver este problema, a adaptação dos coeficientes do equalizador é dividida em dois períodos. No primeiro, uma seqüência pré-estabelecida de símbolos, chamada seqüência de treinamento, é transmitida. Como esta seqüência é conhecida no terminal de recepção, o cálculo do erro de estimação é possível. Ao final deste período, considera-se que o canal já esteja suficientemente equalizado, de forma que o dispositivo de decisão pode fazer uma estimativa suficientemente precisa do símbolo transmitido. Assim, no segundo período, o erro de estimação é calculado pela diferença entre a saída do equalizador e a saída do dispositivo de decisão. Este erro de estimação é chamado *Erro de Decisão Direta*.

Entretanto, a necessidade da transmissão da seqüência de treinamento pode ser muito custosa. Em sistemas de telefonia celular, por exemplo, ela representa uma redução significativa na capacidade do sistema (número de usuários que podem usar o sistema simultaneamente). Neste contexto, surgem as técnicas de equalização cega. Elas buscam a equalização do canal conhecendo apenas a sua saída e algumas informações estatísticas a respeito dos símbolos transmitidos.

Os chamados algoritmos de Bussgang são talvez as técnicas mais difundidas de equalização cega. Estes algoritmos apresentam algumas semelhanças com as técnicas supervisionadas de equalização, pois também se baseiam em uma função custo a ser minimizada pelos coeficientes do equalizador. Esta função custo deve, de alguma maneira, refletir a intensidade da IES presente na saída do equalizador, sem, no entanto, se basear no conhecimento dos símbolos transmitidos.

Recentemente, uma nova abordagem para o problema da equalização cega foi apresentada em [3]. Neste trabalho são introduzidas duas novas técnicas de equalização cega, que se valem de alguns resultados da teoria de predição linear. A primeira se baseia no fato de que um preditor constitui um sistema de fase mínima, enquanto um retropreditor constitui um sistema de fase máxima. Desta forma, o equalizador proposto nesta técnica é composto de uma cascata de um preditor, que compensa os zeros de fase mínima do canal, e de um retropreditor, que compensa os zeros de fase máxima do canal.

A outra técnica preditiva parte do pressuposto de que o sinal transmitido seja branco, ou seja, que a magnitude de seu espectro de frequência seja plana. Esta técnica se baseia no fato de que a saída de um preditor também é branca. Assim, este equalizador é composto pela cascata de um preditor IIR, que compensa a distorção de magnitude introduzida pelo canal, com um equalizador de fase, que compensa a distorção de fase introduzida pelo canal. O equalizador de fase é implementado como um filtro passa-tudo não linear. Em [3], os algoritmos apresentados para esta técnica são baseados na aplicação do método do gradiente (LMS) na minimização das funções custo associadas ao preditor e ao equalizador de fase. No caso do preditor, o critério de minimização é baseado no erro de predição, e, no caso do equalizador de fase, este critério se baseia no erro de decisão direta.

## 1.2 Novos Algoritmos para Técnicas Preditivas

Nesta parte do trabalho são desenvolvidos dois novos algoritmos para o equalizador baseado na cascata de um preditor e de um filtro passa-tudo não linear.

O primeiro algoritmo é baseado na técnica de quadrados mínimos (RLS). A função custo associada a esta técnica não é mais a esperança estatística de um sinal de erro, como no caso do LMS, mas sim em uma estimativa temporal desta esperança.

O desenvolvimento do algoritmo RLS para o equalizador de fase é bastante simplificado se supusermos que a saída do dispositivo de decisão é independente dos coeficientes do filtro passa-tudo não-linear. Esta hipótese clássica também é feita na dedução dos algoritmos de adaptação para o DFE. Assumindo que isto seja verdade, o erro de decisão direta pode ser escrito de uma maneira muito conveniente, que permite a obtenção de um algoritmo RLS para o equalizador de fase semelhante ao RLS tradicional.

O desenvolvimento de um algoritmo RLS para o preditor IIR não é tão simples. A presença de realimentações dificulta sobremaneira os cálculos das derivadas envolvidas no processo de minimização do erro de predição. Entretanto, com algumas aproximações, é possível obter um algoritmo RLS para a adaptação dos coeficientes do preditor IIR.

O segundo algoritmo é baseado na técnica de Gauss-Newton. Da mesma maneira que o LMS, esta técnica busca minimizar a média quadrática do erro de decisão direta. Para isto, entretanto, ela não se baseia apenas no gradiente da função custo, como no caso do LMS, mas também em informações sobre a derivada segunda do critério de minimização. Através desta técnica, obtém-se algoritmos bastante semelhantes àqueles baseados no RLS. De fato, os algoritmos RLS podem ser vistos como um caso particular do GN, onde um determinado parâmetro  $\alpha$  tem seu valor fixado em 1.

### 1.3 Análise de Convergência

Neste capítulo buscaremos analisar teoricamente o comportamento do equalizador de fase, bem como o dos algoritmos propostos para esta estrutura.

Assim, revisaremos alguns resultados teóricos obtidos em [3], onde utilizam-se certas hipóteses simplificadoras para se demonstrar que o critério de minimização do erro de decisão direta de fato leva à equalização do canal. Em outras palavras, mostra-se que o critério de decisão direta é unimodal.

Em seguida, faremos uma análise dos algoritmos LMS e GN associados ao equalizador de fase. Esta análise será baseada na técnica da ODE (equação diferencial ordinária, do inglês *Ordinary Differential Equation*), que consiste em associar uma ODE ao algoritmo em estudo de forma que as propriedades de convergência do algoritmo estejam intimamente relacionadas à estabilidade da ODE. Existem duas técnicas de ODE, uma para a análise de algoritmos com ganho de adaptação desvanecente e outra para algoritmos de ganho constante. Esta segunda técnica será usada na análise do LMS, e através dela será demonstrado que este algoritmo de fato minimiza o erro de decisão direta e, portanto, equaliza o canal.

Os algoritmos de ganho constante são preferidos para aplicações em telecomunicações por permitirem que o equalizador se adapte a não estacionariedades do canal. Assim, seria desejável estudar as propriedades do algoritmo GN neste contexto. Entretanto, a análise do GN com ganho constante se revela muito difícil e ela só se torna matematicamente tratável se supusermos que seu ganho de adaptação tenda a zero. Baseados nesta hipótese, demonstraremos que o GN com ganho desvanecente leva de fato à equalização do canal.

## 1.4 Resultados de Simulações

Uma grande desvantagem do método da ODE é que ele não fornece nenhuma indicação de velocidade de convergência dos algoritmos estudados. As simulações continuam sendo as ferramentas mais adequadas para analisar esta característica de um algoritmo.

Assim, buscaremos neste capítulo estudar o desempenho dos vários algoritmos de equalização propostos. Compararemos também seu desempenho com alguns algoritmos tradicionais de equalização cega, tais como o de Godard.

Para isto, simularemos a transmissão de uma seqüência de símbolos por canais próximos àqueles encontrados na prática: um canal passa-tudo, um canal de microondas e modelos estacionários e não-estacionários de um canal rádio-móvel como o encontrado em sistemas de telefonia celular.

## Capítulo 2

# Equalização de Canais em Transmissão Digital

O objetivo do presente capítulo é introduzir o problema da equalização cega de canais de transmissão digital. Nele, explicaremos a necessidade dos equalizadores, em particular dos equalizadores cegos, apresentando uma visão geral das técnicas estudadas na literatura. O capítulo é organizado da seguinte maneira: na seção 2.1, faremos uma introdução aos sistemas de transmissão digital. Na seção 2.2 mostraremos que os canais de comunicações provocam distorções no sinal transmitido. Na seção 2.3, mostraremos que algumas destas distorções podem ser atenuadas com o uso de equalizadores e apresentaremos algumas técnicas clássicas de equalização. Na seção 2.4, mostraremos uma deficiência destas técnicas clássicas, que motivou o desenvolvimento dos equalizadores cegos. Apresentaremos, também uma visão geral das técnicas de equalização cega. Finalmente, na seção 2.5, apresentaremos as técnicas preditivas de equalização cega, uma técnica recentemente desenvolvida e que será estudada nesta tese.

### 2.1 Sistemas de Transmissão Digital

O processo de transmissão de informações digitais pode ser dividido em vários blocos distintos, representados na figura 2.1 (cf. [4, 5])

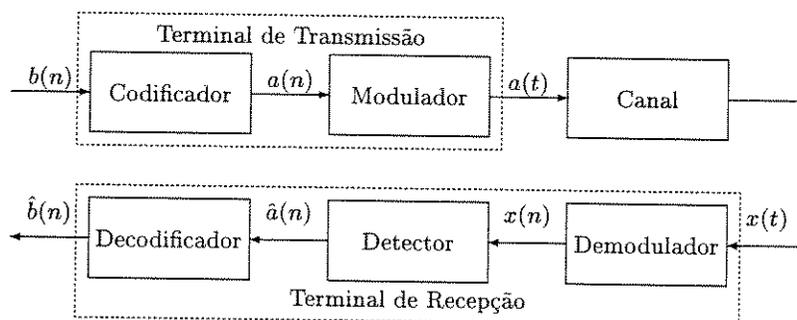


Figura 2.1: Sistema de Comunicações

Nesta figura, vemos que a seqüência binária a ser transmitida,  $\{b(n)\}$ , passa inicialmente por um codificador, cujo objetivo é transformá-la em uma outra seqüência com características mais favoráveis à transmissão. A codificação pode envolver, por exemplo, a introdução de um código corretor de erros, ou a encriptação do sinal. Em seguida, a seqüência codificada é modulada. Este processo transforma o sinal digital em um sinal analógico com um espectro adequado para sua transmissão pelo canal.

O sinal modulado é, então, enviado para o terminal de recepção. Essa transmissão se dá através de um meio físico pelo qual o sinal se propaga, e que pode ser o ar, um cabo telefônico, fibras óticas, etc. Este meio físico é denominado canal e introduz algumas distorções no sinal transmitido, conforme veremos adiante.

No terminal de recepção busca-se desfazer, uma a uma, as operações realizadas na transmissão, recuperando a seqüência originalmente transmitida. Assim, o sinal recebido é primeiramente demodulado, processo inverso à modulação e que transforma o sinal analógico recebido em um sinal digital. A saída do demodulador passa pelo detector, dispositivo que tenta compensar algumas distorções introduzidas pelo canal. Finalmente, o sinal é decodificado, e a seqüência transmitida, recuperada.

## 2.2 Distorções Provocadas pelos Canais de Comunicações

O canal de comunicações caracteriza o meio físico pelo qual será transmitido o sinal modulado. Esta transmissão raramente é ideal, e a passagem do sinal analógico pelo canal impõe-lhe uma série de distorções, tais como:

- Adição de ruído: este fenômeno é causado pelo acúmulo de várias distorções aleatórias que se sobrepõem ao sinal durante todo o seu processamento, seja o ruído térmico ou o ruído *shot* dos equipamentos, sejam descargas elétricas ou outras interferências externas que surgem durante a transmissão. Devido ao grande número de fontes de ruído e invocando o teorema central do limite, este ruído é geralmente modelado como um sinal branco gaussiano de média zero que se soma ao sinal transmitido, na entrada do terminal de recepção.
- Interferência Entre-Símbolos (IES): na maioria dos casos, a resposta de um canal de comunicações a um símbolo transmitido tem duração maior que a do próprio símbolo. Ou seja, quando o canal começa a responder à transmissão de um símbolo, ele ainda está respondendo à transmissão de símbolos anteriores, de forma que estas respostas se sobrepõem em sua saída. Assim, a resposta do canal a um símbolo sofre a interferência dos símbolos adjacentes, resultando na IES. Este fenômeno ficará mais claro quando apresentarmos o modelo discreto do canal.
- Outras distorções: várias outras distorções poderiam ser citadas, algumas específicas de determinados canais. Consideremos, por exemplo, os canais rádio-móvel, cujo meio de transmissão é o ar. Estes canais estão presentes, por exemplo, em sistemas de telefonia celular. Um sinal transmitido pelo ar vai de uma antena a outra por vários percursos, sendo refletido em paredes, no chão etc. Na antena receptora os sinais advindos dos vários percursos interferem um no outro. Devido à característica essencialmente não-estacionária do canal rádio móvel, esta interferência pode ser destrutiva, gerando desvanecimento do sinal (cf. [6]).

O combate ao problema do ruído é feita pela inclusão de um filtro casado na entrada do terminal de recepção. Os filtros casados têm a propriedade de maximizar a Relação Sinal Ruído (SNR) em sua saída (cf. [7]). Em sua implementação, o filtro deve ser casado com a portadora do sinal analógico que traz a informação para o terminal de recepção. No caso de sistemas de transmissão digital, este sinal corresponde ao sinal modulador, que é, em geral, um sinal sinusoidal. Entretanto, até chegar ao terminal de recepção, o sinal modulado passa pelo canal, sofrendo algumas distorções. Assim, o filtro ideal deveria ser casado com a resposta do canal ao sinal modulador. Entretanto, o conhecimento desta resposta pressupõe o conhecimento do canal, o que impossibilita a implementação prática do filtro ideal. Desta forma, implementa-se um filtro aproximado, casado com o sinal modulador e sem levar em consideração o efeito do canal sobre este.

As outras interferências mencionadas são combatidas pelo dispositivo de detecção mostrado na figura 2.1. Entre os possíveis procedimentos adotados para combater a IES, por exemplo, encontram-se os estimadores de máxima verossimilhança (e sua versão sub-ótima, implementada com o algoritmo de Viterbi) e os equalizadores (cf. [4, 5]). Para combater os efeitos do desvanecimento em sistemas de telefonia móvel, são adotadas, entre outras, técnicas que diversificam a recepção, através do uso de várias antenas (cf. [6]).

### 2.2.1 Representação Discreta do Canal de Comunicações

Conforme vimos anteriormente, os equalizadores são dispositivos destinados a combater a IES. Para melhor esclarecer a atuação dos equalizadores, é necessária uma melhor compreensão do fenômeno da IES. No que se segue, buscaremos mostrar como um canal de comunicações pode ser modelado como um filtro digital e como a IES pode ser vista como uma consequência da filtragem, pelo canal, do sinal digital transmitido.

Assim, consideremos inicialmente o processo de modulação, que transforma a seqüência de bits a serem transmitidos  $\{b(n)\}$  em um sinal analógico com características adequadas à transmissão. Todo o procedimento de modulação é executado sobre blocos de  $K$  bits consecutivos. Ou seja, o modulador toma  $K$  bits, associa a eles um sinal analógico, transmite este sinal, toma os próximos  $K$  bits, e assim sucessivamente. Estes blocos de  $K$  bits são chamados símbolos. Existem  $L = 2^K$  diferentes seqüências de bits de comprimento  $K$  e, portanto,  $L$  diferentes símbolos. Cada um destes símbolos é denotado  $s_i$  e o conjunto dos  $L$  possíveis símbolos é chamado alfabeto.

A associação de um sinal analógico a um símbolo é biunívoca. Assim, a cada símbolo  $s_i$  corresponde uma e apenas uma forma de onda, denotada por  $y_i(t)$ . Estas formas de onda são, em geral, de faixa estreita<sup>1</sup> e resultam da modulação de uma portadora. De maneira geral, podemos escrever:

$$y_i(t) = \Re \left\{ u_i(t) e^{j 2\pi f_c t} \right\} = A_i(t) \cos(2\pi f_c t + \phi_i(t)), \quad (2.1)$$

onde  $f_c$  é a frequência do sinal portador e  $\Re$  indica a parte real. Nesta equação, o sinal  $u_i(t)$  é a chamada representação em banda básica de  $y_i(t)$ . Note que  $u_i(t)$  é, em geral, uma função complexa. Como a relação entre  $u_i(t)$  e  $y_i(t)$  também é biunívoca, podemos caracterizar a modulação simplesmente pelo mapeamento de  $s_i$  em  $u_i(t)$ .

<sup>1</sup>Um sinal de faixa estreita é aquele cuja faixa de frequência é muito menor que a frequência da portadora

Muitas vezes, a associação de um símbolo  $s_i$  com um sinal  $u_i(t)$  é feita através de mudanças em alguma característica de uma mesma função  $u(t)$ . Se tivermos, por exemplo, uma modulação em fase e/ou amplitude, podemos representar cada função  $u_i(t)$  como

$$u_i(t) = \alpha_i u(t), \quad (2.2)$$

onde  $\alpha_i$  é um valor, em geral complexo, que contém a informação de amplitude e fase de cada  $u_i(t)$ . Mais uma vez, a relação entre  $u_i(t)$  e  $\alpha_i$  é biunívoca, e o processo de modulação pode ser visto como aquele que associa a cada símbolo  $s_i$  um valor  $\alpha_i$ .

Os possíveis valores assumidos por  $\alpha_i$  são chamados constelação, devido ao aspecto de sua representação no plano complexo. Por exemplo, numa modulação 4-QAM, a constelação pode ser formada pelo conjunto  $\{+1, +j, -1, -j\}$ . No que se segue, o termo símbolo se refere indistintamente a  $s_i$  e  $\alpha_i$ .

Desta forma, podemos visualizar o modulador como um sistema que, inicialmente, mapeia  $s_i$  em  $\alpha_i$  e, em seguida, aplica o sinal discreto  $\alpha_i \delta(t)$  a um filtro linear cuja saída é  $y_i(t)$ . Ora, no terminal de recepção, o sinal na saída do demodulador é amostrado e, portanto, também é discreto. Assim, o canal, os filtros de transmissão e de recepção e qualquer outro processamento executado entre a entrada do modulador e a saída do demodulador correspondem a um sistema com entrada e saída discretas e, portanto, podem ser representados por um filtro linear discreto, cuja resposta ao impulso é dada por uma seqüência  $\{f_k\}$ . Daqui em diante, chamaremos este filtro simplesmente de canal.

A figura 2.2 traz uma representação deste modelo discreto. Nela, o valor de  $a(n)$  representa o símbolo  $\alpha_i$  a ser transmitido no instante  $n$ ,  $\{f_k\}$  representa a resposta ao impulso do canal e  $\{v(n)\}$  é o ruído aditivo anteriormente mencionado. A seqüência  $\{x(n)\}$  mostrada nesta figura é a seqüência obtida no terminal de recepção após a amostragem do sinal recebido. Nesta figura, mostramos também um equalizador e um dispositivo de decisão, que são componentes do bloco de detecção e serão discutidos na próxima seção.

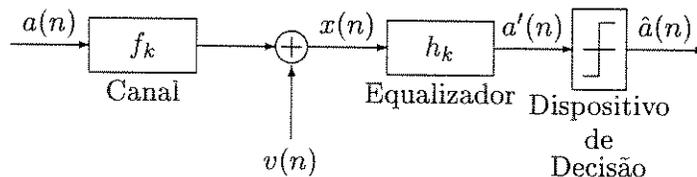


Figura 2.2: Representação Discreta de um Sistema de Comunicações

### 2.2.2 Interferência Entre Símbolos

A representação discreta do canal de comunicações nos permite escrever o sinal discreto recebido como:

$$x(n) = \sum_{k=-\infty}^{\infty} f_k a(n-k) + v(n). \quad (2.3)$$

Conforme podemos ver, o sinal recebido no instante  $n$ ,  $x(n)$  é uma combinação de símbolos passados e futuros. Ou seja, não podemos recuperar  $a(n)$  diretamente de  $x(n)$ . Escrevendo separadamente alguns termos desta equação, o problema se torna ainda mais claro:

$$x(n) = a(n-\delta) f_\delta + \sum_{k \neq \delta} f_k a(n-k) + v(n). \quad (2.4)$$

O primeiro termo do lado direito da equação 2.4 representa um símbolo transmitido, a menos de uma constante multiplicativa  $f_\delta$  e um atraso  $\delta$ . Este símbolo representa a informação que se deseja recuperar. O segundo termo representa a distorção ocasionada pela IES. Ele pode ser visto como um ruído adicional que deve ser eliminado. A razão pela qual se opta por recuperar  $a(n - \delta)$  e não algum outro símbolo qualquer é que assume-se que  $|f_\delta| = \max_k(|f_k|)$ . Desta forma,  $a(n - \delta)$  é o símbolo transmitido que tem o maior peso na formação da saída  $x(n)$ .

Uma medida da intensidade de IES é a chamada *closed eye measure*, dada por:

$$\rho = \frac{\sum_k |f_k| - |f_\delta|}{|f_\delta|}, \quad (2.5)$$

onde ainda temos  $|f_\delta| = \max_k(|f_k|)$ . Esta medida está associada à abertura do diagrama de olho do canal (cf. [4]). Se  $\rho < 1$ , considera-se que o olho do canal está aberto. Nesta situação, supondo-se a ausência de ruído, o símbolo transmitido pode ser corretamente identificado simplesmente através de um dispositivo de decisão. Este dispositivo, mostrado na figura 2.2, é um quantizador sem memória, cuja saída é o símbolo da constelação transmitida que esteja mais próximo do sinal na sua entrada. Entretanto, se  $\rho > 1$ , algum processamento deve ser feito com o sinal recebido para permitir a correta identificação do símbolo transmitido. Na próxima seção, veremos um procedimento que pode ser adotado para a minimização da IES.

## 2.3 Equalizadores

Pode-se constatar, a partir do modelo discreto do canal, que a IES é o resultado de uma filtragem linear do sinal transmitido. Ora, se conhecêssemos a função de transferência do canal, poderíamos colocar, em sua saída, um filtro cuja função de transferência fosse a inversa da do canal. Assim, a função de transferência da cascata do canal com este filtro seria dada por:

$$F(z^{-1}) H(z^{-1}) = \kappa z^{-\delta}, \quad 0 \neq \kappa \in \mathbb{C}, \quad \delta \in \mathbb{Z}_+, \quad (2.6)$$

onde  $F(z^{-1})$  e  $H(z^{-1})$  representam, respectivamente, as funções de transferência do canal e de seu filtro inverso,  $\kappa$  é uma constante multiplicativa e  $\delta$  é um atraso, que garante que o filtro inverso seja causal e estável. Note que a equação 2.6 implica que a cascata canal+filtro inverso transmite sinais sem distorcê-los. Em outras palavras, ao filtrarmos a saída do canal com o filtro inverso, estaremos “desfazendo” a IES. Um filtro que possua a propriedade de inverter as distorções provocadas por um canal de comunicações é chamado equalizador. O uso de um equalizador para combater a IES está mostrado na figura 2.2. Nela,  $\{h_k\}$  é a resposta ao impulso do equalizador.

A partir de 2.6, vemos que, dada a transmissão de uma seqüência  $\{a(n)\}$ , a saída do equalizador é dada por  $a'(n) = \kappa a(n - \delta)$ , além de um componente de ruído. Este valor é igual a um símbolo transmitido, a menos de uma constante multiplicativa e um atraso. Em outras palavras, a saída do equalizador contém informações sobre apenas um símbolo transmitido, permitindo sua correta estimação através do uso de um dispositivo de decisão. Com isto, notamos claramente que o uso de um equalizador elimina a IES.

O equalizador é implementado, na maioria das vezes, após a saída do demodulador. Dessa forma, o sinal em sua entrada é digital. Assim, o equalizador é geralmente implementado

como um filtro digital. Sem dúvida, a estrutura de equalização mais estudada e utilizada é o filtro transversal linear, mostrado na figura 2.3. Este filtro também é conhecido por filtro FIR (do inglês *Finite Impulse Response*).

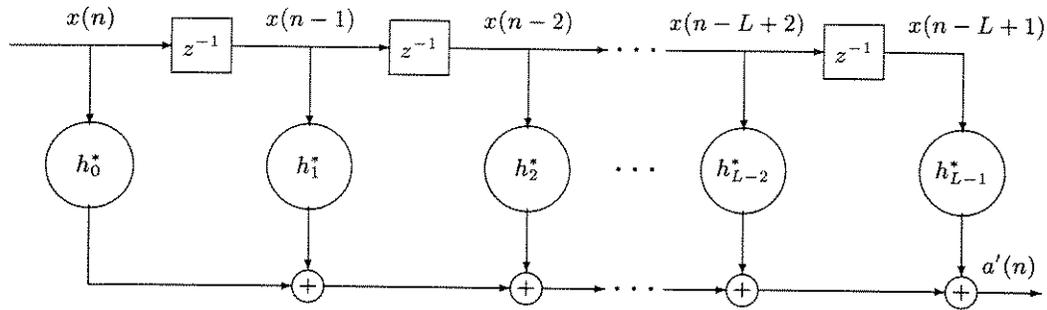


Figura 2.3: Estrutura do Filtro Transversal Linear

A saída deste filtro é dada por:

$$a'(n) = \sum_{k=0}^{L-1} h_k^* x(n-k) \quad (2.7)$$

ou vetorialmente por

$$a'(n) = H^H X(n), \quad (2.8)$$

onde o expoente  $H$  indica as operações de transposição e conjugação e os vetores  $H$  e  $X(n)$  correspondem, respectivamente, aos coeficientes do filtro e às últimas amostras do sinal de entrada do equalizador, e são definidos por

$$\begin{aligned} H &= [h_0, \dots, h_{L-1}]^T, \\ X(n) &= [x(n), \dots, x(n-L+1)]^T. \end{aligned} \quad (2.9)$$

Na seqüência, veremos algumas maneiras de obter valores de  $H$  que levem à equalização do canal.

### 2.3.1 Técnicas Clássicas de Equalização

A equação 2.6 nos dá a definição do equalizador ideal. A partir dela, podemos constatar que sua função de transferência é dada por

$$H(z^{-1}) = \kappa z^{-\delta} F^{-1}(z^{-1}). \quad (2.10)$$

Assim, se conseguirmos obter um vetor  $H$  tal que o filtro FIR da figura 2.3 possua a função de transferência desejada, teremos obtido o equalizador ideal. Entretanto, a equação 2.10 deixa claro que a obtenção de  $H(z^{-1})$  envolve o conhecimento da função de transferência do canal. Em situações práticas, este conhecimento raramente é possível, e outras soluções para o problema da equalização devem ser encontradas.

Uma das técnicas mais difundidas para obtenção do vetor  $H$  consiste em definir uma função  $J(H)$  do vetor de coeficientes do filtro, de forma que  $J(H)$  esteja relacionada à

quantidade de IES presente na saída do equalizador. Esta função é chamada função custo ou função objetivo, e o valor de  $H$  que a minimiza é definido como  $H^o$ . Este valor é usado para equalizar o canal, pois ele deve levar a uma IES baixa na saída do equalizador.

A função custo mais clássica é a esperança do erro quadrático, também conhecida por critério de Wiener. Ela é definida como a média quadrática da diferença entre o valor desejado para saída do equalizador e aquele que se obtém de fato. Matematicamente, temos:

$$J(H) = E(|e(n)|^2), \quad (2.11)$$

onde  $e(n) = a(n - \delta) - a'(n)$ .

Esta é uma função quadrática do vetor de coeficientes e possui, portanto, um único mínimo (cf. [8]). A minimização deste critério leva às *equações de Wiener-Hopf*:

$$H^o = R^{-1} r_{ax}, \quad (2.12)$$

onde

$$R = E(X(n) X^H(n)) \quad (2.13)$$

$$r_{ax} = E(a^*(n) X(n)). \quad (2.14)$$

O filtro obtido da solução destas equações é conhecido como filtro de Wiener. Entretanto, a solução das equações de Wiener-Hopf exige o conhecimento de características estatísticas do sinal recebido  $x(n)$ . Ora, conhecê-las *a priori* é equivalente a conhecer o canal, o que torna esta técnica inviável na maioria das aplicações práticas. Para contornar este problema, foram desenvolvidos os algoritmos adaptativos, que tentam estimar recursivamente as grandezas estatísticas necessárias para a solução das equações de Wiener-Hopf. Estes algoritmos serão discutidos a seguir.

### 2.3.2 Equalizadores Adaptativos

O problema da equalização pode ser encarado como um problema de otimização, pois envolve a minimização de  $J(H)$ . Desta forma, técnicas iterativas de otimização, tais como o método do gradiente e método de Newton, podem ser aplicadas na sua solução. Estas técnicas adaptam recursivamente o vetor  $H$ , e devem, teoricamente, convergir para um valor que minimiza a função custo. Assim, a cada iteração destes algoritmos obtemos um diferente vetor de coeficientes, que por isso passarão a ser chamados de  $H(n)$ . Idealmente, a aplicação dos métodos iterativos na minimização do critério de Wiener deve levar a uma seqüência de vetores  $H(n)$  que convergem para a solução de Wiener-Hopf, independente de suas condições iniciais.

Tomemos, por exemplo, o método do gradiente, que adapta o vetor  $H(n)$  na direção oposta à dada pelo gradiente da função custo. O princípio deste algoritmo é que a direção oposta ao gradiente é a direção de maior decréscimo da função custo. Assim, a cada iteração, fazemos

$$H(n+1) = H(n) - \mu \nabla_H J, \quad \mu > 0, \quad (2.15)$$

onde  $\nabla_H J$  é o gradiente de  $J(H)$  em relação a  $H$  e  $\mu$  é chamado passo de adaptação, indicando o quanto o algoritmo deve andar na direção oposta ao gradiente. Desta forma, a

cada iteração, obtém-se um valor de  $H(n)$  ao qual corresponderá um menor valor de  $J(H)$ . Iterativamente, este algoritmo acaba convergindo para um valor mínimo de  $J(H)$ .

Para o critério de Wiener, a gradiente da função custo é dado por  $\nabla_H J = -E(X(n)e^*(n))$ . Conforme vemos, o uso de um método iterativo de otimização não resolveu o problema apresentado pelo filtro de Wiener, pois o cálculo de  $\nabla_H J$  ainda exige o conhecimento de características estatísticas do sinal recebido e, conseqüentemente, do canal. O método de Newton, além de usar o gradiente de  $J(H)$ , usa sua matriz Hessiana, constituída pelas suas derivadas segundas. O cálculo da Hessiana de  $J(H)$  também envolve o conhecimento de características estatísticas de  $X(n)$ .

No problema prático de equalização estas informações não estão disponíveis, mas podem ser estimadas (cf. [8]). Os algoritmos que usam estimativas do gradiente ou da matriz Hessiana para a minimização de  $J(H)$  são chamados algoritmos adaptativos. Os equalizadores obtidos por técnicas adaptativas são chamados equalizadores adaptativos.

Uma propriedade interessante dos algoritmos adaptativos é que as características estatísticas do canal são estimadas a cada iteração. Desta forma, se elas sofrerem alterações, como ocorre freqüentemente em canais não-estacionários tais como os canais dos sistemas de telefonia celular, o algoritmo será capaz de estimar estas novas características do canal e, portanto, obter um filtro que equalize o novo canal. Isto torna os equalizadores adaptativos muito úteis em ambientes não-estacionários.

Na seqüência, ilustraremos alguns métodos de equalização adaptativa: os algoritmos LMS (do inglês *Least Mean Squares*), RLS (do inglês *Recursive Least Squares*) e o de Gauss-Newton.

### Algoritmo LMS

O algoritmo LMS é baseado no método do gradiente. O valor exato do gradiente de  $J(H)$  é dado por  $\nabla_H J = -E(X(n)e^*(n))$ . O algoritmo LMS usa uma estimativa para este valor que consiste simplesmente em ignorar o operador esperança no cálculo do gradiente, de forma que  $\hat{\nabla}_H J = -X(n)e^*(n)$ . Esta é a chamada estimativa instantânea da esperança. Assim, o algoritmo fica:

$$e(n) = a(n - \delta) - H^H(n)X(n) \quad (2.16)$$

$$H(n+1) = H(n) + \mu X(n)e^*(n) \quad (2.17)$$

Este método é também chamado método do gradiente estocástico e é um dos mais populares algoritmos adaptativos, pois apresenta um bom desempenho em situações práticas, aliado a uma notável simplicidade computacional.

### Algoritmo RLS

Outro algoritmo importante, caracterizado por sua alta velocidade de convergência, é o RLS. Ao invés de estimar o gradiente da função custo, este algoritmo trabalha com uma estimativa da própria função custo. Assim, em cada instante de tempo  $n$  busca-se o vetor de coeficientes  $H(n)$  que minimiza não mais a esperança estatística do sinal de erro, mas a sua média temporal. Desta forma, o algoritmo RLS se baseia na função custo:

$$\mathcal{E}(n) = \sum_{i=1}^n w^{n-i} |e(i, n)|^2. \quad (2.18)$$

O termo  $e(i, n)$  indica a diferença entre o valor desejado na saída do equalizador no instante  $i$  e seu valor real no mesmo instante, calculado usando o vetor de coeficientes obtido no instante  $n$ . Em outras palavras,  $e(i, n) = a(i - \delta) - H(n)^H X(i)$ . Ou seja, a cada instante  $n$ ,  $\mathcal{E}(n)$  indica a média temporal dos erros quadráticos como se todos os erros tivessem sido obtidos com o uso dos mesmos coeficientes  $H(n)$ . Conforme veremos, a cada iteração o algoritmo RLS minimiza  $\mathcal{E}(n)$ . Assim, a cada iteração teremos um vetor ótimo no sentido dos mínimos quadrados.

O fator  $w < 1$  é o chamado fator de esquecimento, cuja função é dar um peso maior para os erros mais recentes. A razão disto é que quanto menor o valor de  $i$ , ou seja, quando mais o erro estiver no passado, menor o valor de  $w^{n-i}$  e menor o peso de  $e(i, n)$  no cálculo de  $\mathcal{E}(n)$ .

A função custo  $\mathcal{E}(n)$  caracteriza um problema de mínimos quadrados ponderado, cuja solução é dada por (cf. [8]):

$$H(n) = \hat{R}^{-1}(n) \hat{r}_{ax}(n), \quad (2.19)$$

onde  $\hat{R}(n)$  e  $\hat{r}_{ax}(n)$  são dados respectivamente por:

$$\hat{R}(n) = \sum_{i=1}^n w^{n-i} X(i) X^H(i) \quad (2.20)$$

$$\hat{r}_{ax}(n) = \sum_{i=1}^n w^{n-i} a^*(i - \delta) X(i). \quad (2.21)$$

É interessante notar as semelhanças que estas equações guardam com as de Wiener-Hopf. A matriz  $\hat{R}(n)$  pode ser vista como uma estimativa da matriz de autocorrelação de  $x(n)$  (o fator  $R$  nas equações de Wiener-Hopf) e  $\hat{r}_{ax}(n)$  como uma estimativa da correlação cruzada entre  $x(n)$  e  $a(n)$  (o fator  $r_{ax}$  nas equações de Wiener-Hopf), onde, em ambos os casos, a esperança estatística é substituída por uma média temporal. Assim, o RLS pode ser visto como um algoritmo que, a cada iteração, usa todas as informações disponíveis para estimar as grandezas envolvidas nas equações de Wiener-Hopf.

Este algoritmo possui ainda mais uma propriedade interessante, conforme veremos adiante: ele consegue evitar o cálculo da inversa de  $\hat{R}(n)$ . A inversão de matrizes é uma operação computacionalmente cara e instável, e evitá-la possui um grande interesse prático. Isto é conseguido com o uso de um resultado importante de álgebra matricial, o *Lema de Inversão de Matrizes*:

**Lema 2.1** *Sejam  $A$ ,  $B$  e  $D$  matrizes definidas positivas e  $C$  uma matriz tais que suas dimensões sejam compatíveis com a igualdade*

$$A = B + C D C^H. \quad (2.22)$$

Então, é verdade que

$$A^{-1} = B^{-1} - B^{-1} C \left( D^{-1} + C^H B^{-1} C \right)^{-1} C^H B^{-1}. \quad (2.23)$$

Este lema permite que, no instante  $n$ , ao invés de calcularmos  $\hat{R}(n)$  e depois sua inversa, calculemos diretamente  $\hat{R}^{-1}(n)$  a partir de seu valor no instante  $n - 1$ ,  $\hat{R}^{-1}(n - 1)$ . Para

ver como isto é feito, primeiramente reescrevemos a definição de  $\hat{R}(n)$ :

$$\hat{R}(n) = X(n) X^H(n) + \sum_{i=1}^{n-1} w^{n-i} X(i) X^H(i) \quad (2.24)$$

e notamos que o segundo termo do lado direito da equação acima,  $\sum_{i=1}^{n-1} w^{n-i} X(i) X^H(i) = w \hat{R}(n-1)$ .

Em seguida, fazemos as associações:

$$\begin{aligned} A &= \hat{R}(n) \\ B &= w \hat{R}(n-1) \\ C &= X(n) \\ D &= 1 \end{aligned}$$

Fazendo  $P(n) = \hat{R}^{-1}(n)$  e aplicando o lema de inversão de matrizes, obtemos:

$$P(n) = \frac{1}{w} \left( P(n-1) - \frac{P(n-1) X(n) X(n)^H P(n-1)}{w + X(n)^H P(n-1) X(n)} \right). \quad (2.25)$$

Usando mais algumas manipulações simples, obtemos a forma final do RLS (cf. [8]):

$$e(n) = a(n - \delta) - H^H(n) X(n) \quad (2.26)$$

$$U(n) = P(n-1) X(n) \quad (2.27)$$

$$k(n) = \frac{1}{w + X^H(n) U(n)} \quad (2.28)$$

$$G(n) = k(n) U(n) \quad (2.29)$$

$$P(n) = \frac{1}{w} \left( P(n-1) - k(n) U(n) U^H(n) \right) \quad (2.30)$$

$$H(n) = H(n-1) + e^*(n) G(n) \quad (2.31)$$

onde definimos, para simplicidade de notação,  $e(n) = e(n, n)$ .

### Algoritmo de Gauss-Newton

O algoritmo de Gauss-Newton é uma aproximação estocástica para o método de otimização de Newton. Este método é semelhante ao do gradiente, mas a direção de adaptação leva em conta a matriz Hessiana da função custo.

Seja  $S(n)$  a Hessiana de  $J(H)$ . O método de Gauss-Newton é dado por:

$$H(n+1) = H(n) - \mu S^{-1}(n) \nabla_H J. \quad (2.32)$$

A estimação do gradiente já foi vista no método LMS. No caso do critério de Wiener, a Hessiana é a própria função de autocorrelação  $R(n)$ , que pode ser estimada através de uma média temporal:

$$\hat{S}(n) = \alpha X(n) X^H(n) + \alpha \sum_{i=1}^{n-1} (1 - \alpha)^{n-i} X(i) X^H(i). \quad (2.33)$$

O fator  $\alpha$  funciona como um fator de esquecimento e vale, em geral,  $1 - w$ . Observe a semelhança que esta equação guarda com a equação 2.24. De fato, o algoritmo RLS pode ser visto como um caso especial do algoritmo de Gauss-Newton, onde não se leva em conta o fator  $\alpha$ . Uma outra maneira de estabelecer este paralelo entre os algoritmos GN e RLS é observarmos que o primeiro pode ser obtido a partir do segundo se no algoritmo RLS o vetor de entrada  $X(n)$  for normalizado por um fator  $1/\sqrt{\alpha}$ . Fica mais uma vez evidente a possibilidade de uma maior flexibilidade na escolha dos parâmetros do algoritmo GN.

A partir de 2.32 e usando o Lema de Inversão de Matrizes, obtemos o algoritmo de Gauss-Newton:

$$e(n) = a(i - \delta) - H^H(n) X(n) \quad (2.34)$$

$$U(n) = P(n-1) X(n) \quad (2.35)$$

$$k(n) = \frac{1}{w + \alpha X^H(n) U(n)} \quad (2.36)$$

$$G(n) = k(n) U(n) \quad (2.37)$$

$$P(n) = \frac{1}{w} \left( P(n-1) - \alpha k(n) U(n) U^H(n) \right) \quad (2.38)$$

$$H(n) = H(n-1) + \alpha e^*(n) P(n) X(n) \quad (2.39)$$

### Treinamento de Equalizadores

Da forma como foram apresentados, os algoritmos adaptativos apresentam um aparente contra-senso, pois sua aplicação envolve o conhecimento de  $\{a(n)\}$  para permitir a formação do erro  $e(n)$ . Ora, para que  $\{a(n)\}$  seja conhecida no terminal de recepção, é necessário que o canal esteja equalizado. Para resolver este aparente paradoxo, a solução adotada na prática é dividir a adaptação em dois períodos.

No primeiro, dito *Período de Treinamento*, transmite-se uma seqüência pré-estabelecida, chamada *seqüência de treinamento*. Uma cópia desta seqüência, chamada *seqüência de referência*, é gerada internamente no receptor, possibilitando o cálculo de  $e(n)$ , que chamaremos de *erro verdadeiro*. O período de treinamento tem duração suficiente para que o algoritmo leve os coeficientes  $H(n)$  a um valor próximo a  $H^o$ , de forma a abrir o olho do canal.

Após este período, começa a transmissão efetiva dos bits de informação. Entretanto, o algoritmo adaptativo pode não convergir completamente durante o treinamento, ou ainda o canal pode apresentar variações durante a transmissão. Desta forma, o algoritmo deve continuar adaptando os coeficientes do equalizador mesmo durante a transmissão dos bits de informação. Este segundo período de adaptação é chamado *Período de Rastreio*. Para possibilitar o cálculo do sinal de erro  $e(n)$  necessário aos algoritmos adaptativos, assume-se que, após o término do período de treinamento, o canal está "suficientemente" equalizado, de forma que a saída do dispositivo de decisão,  $\hat{a}(n)$ , pode ser considerada uma estimativa bastante confiável de  $a(n)$ . O erro usado nos algoritmos adaptativos passa a ser dado por  $\varepsilon(n) = \hat{a}(n) - a(n)$ . Este erro é chamado *erro de Decisão Direta* (DD) e, no período de rastreio, diz-se que o equalizador está operando no modo de Decisão Direta.

## 2.4 Equalizadores Cegos

O período de rastreo recebe este nome pois nele o algoritmo adaptativo consegue acompanhar (rastrear) variações lentas do canal. Entretanto, canais altamente não estacionários, como os de telefonia móvel, apresentam variações muito velozes, impossíveis de serem acompanhadas pelo algoritmo funcionando no modo de decisão direta. Nestes casos, para garantir a equalização do canal, o treinamento deve ser feito periodicamente. No sistema GSM (sistema europeu de telefonia móvel), por exemplo, uma sequência de treinamento de 26 bits é enviada dentro de cada quadro de 148 bits. No sistema D-AMPS americano, são 28 bits de treinamento a cada quadro de 324.

Isto representa uma séria deficiência dos algoritmos clássicos de equalização, pois a transmissão da sequência de treinamento diminui a taxa de bits úteis e, conseqüentemente, o número de usuários que o sistema pode atender. Ainda citando os sistemas GSM e D-AMPS, as taxas de bits úteis são, respectivamente, 82,43% e 91,83%. Para contornar este problema, foram desenvolvidos algoritmos de adaptação, chamados algoritmos cegos, que dispensam o uso da sequência de treinamento. Estes algoritmos levam aos chamados *equalizadores autodidatas* ou *equalizadores cegos*. Obviamente, estes algoritmos não podem equalizar o canal sem nenhuma informação a respeito do sinal transmitido. Como as informações estatísticas deste sinal são sempre conhecidas, é nelas que os algoritmos cegos se baseiam para buscar a equalização do canal.

Conforme vimos anteriormente, os cálculos para a obtenção do filtro de Wiener e os algoritmos adaptativos que levam a ele usam as matrizes de autocorrelação e de correlação cruzada, ou suas estimativas. Ou seja, eles se baseiam em momentos de segunda ordem para equalizar o canal. Consideremos estes momentos, agora no contexto de equalização cega. Assim, seja  $S_{aa}(\omega)$  a densidade espectral de potência do sinal transmitido  $\{a(n)\}$ , que é dada pela transformada de Fourier da função de autocorrelação de  $\{a(n)\}$  e, portanto, baseia-se nas estatísticas de 2ª ordem do sinal transmitido. Seja também  $F(\omega)$  a resposta em frequência do canal. Vale, então, a igualdade:

$$S_{xx}(\omega) = |F(\omega)|^2 S_{aa}(\omega), \quad (2.40)$$

onde  $S_{xx}(\omega)$  é a densidade espectral de potência do sinal recebido  $\{x(n)\}$ . Verificamos, então, que a estimação de  $S_{xx}(\omega)$  (e, portanto, o uso das estatísticas de 2ª ordem do sinal recebido) nos permite identificar o módulo da resposta em frequência do canal.

Ora, para equalizar um canal devemos conhecer sua resposta em amplitude e em fase. Em um sistema de fase mínima<sup>2</sup>, a amplitude e a fase da resposta em frequência estão biunivocamente relacionadas pelas transformadas de Hilbert (cf. [9]). Por outro lado, se o canal não for de fase mínima, não podemos determinar a fase de sua resposta em frequência conhecendo apenas a sua amplitude. Ou seja, não podemos equalizar cegamente um canal de fase não-mínima usando apenas as estatísticas de 2ª ordem do sinal recebido. É necessário, então, usar as informações das estatísticas de ordem superior (EOS).

Isto nos leva a outra conclusão: se o sinal transmitido for gaussiano e o canal não for de fase mínima, ele não pode ser identificado por um algoritmo cego. Isto porque toda a informação estatística de um sinal gaussiano está contida nos seus dois primeiros momentos.

<sup>2</sup>Um sistema de fase mínima é caracterizado por possuir todos os seus polos e zeros no interior da circunferência de raio unitário

Felizmente, na maioria dos casos, o sinal transmitido assume com igual probabilidade valores dentro de uma constelação de símbolos e possui, portanto, uma distribuição de probabilidade uniforme.

Da mesma maneira que os algoritmos clássicos de equalização, os algoritmos cegos devem ser adaptativos. Conforme vimos, os algoritmos adaptativos surgem da aplicação de um método iterativo de otimização (gradiente, Gauss-Newton, etc.) na minimização de uma função custo. Desta forma, o projeto de um equalizador cego envolve as seguintes etapas:

- Determinar a estrutura do equalizador (filtro FIR, IIR, etc.).
- Achar alguma função custo associada aos coeficientes da estrutura e que não dependa do sinal transmitido, mas apenas de suas características estatísticas. Esta função deve refletir a intensidade da IES na saída do equalizador, de forma que os parâmetros do equalizador associados ao ponto que a minimiza correspondam a uma boa abertura do olho do canal.
- Escolha de um método iterativo para minimização desta função custo e determinação de sua aproximação estocástica.

Diferentes procedimentos nas etapas acima levam a diferentes técnicas de equalização. Faremos aqui apenas uma referência às principais, dividindo-as em grupos:

- Técnicas que fazem uso explícito das EOS ou de suas transformadas de Fourier, os poliespectros. Apesar de apresentarem convergência global, estas técnicas precisam estimar os momentos de ordem superior, aumentando muito sua complexidade computacional. Dois resultados importantes são obtidos a partir do estudo destas técnicas:
  - Em [10] está provado que uma condição suficiente para que o canal esteja equalizado é que todos os momentos do sinal transmitido,  $a(n)$ , e da saída do equalizador,  $a'(n)$ , sejam iguais.
  - Uma condição necessária e suficiente para que o canal esteja equalizado é que (cf. [11]):

$$\begin{aligned} E(|a|^2) &= E(|a'|^2) \\ |K(a)| &= |K(a')|, \end{aligned} \tag{2.41}$$

onde  $K(a) = E(|a|^4) - 2E^2(|a|^2) - |E(|a|^2)|^2$  é a chamada curtose de  $a$ .

- Técnicas cicloestacionárias, que se baseiam na amostragem do sinal analógico recebido a uma taxa maior que a de transmissão. Estas técnicas usam apenas os momentos de segunda ordem do sinal super-amostrado e, portanto, permitem equalizar canais quando o sinal transmitido for gaussiano.
- As técnicas ditas de Bussgang, que fazem uso indireto das EOS. Devido à sua importância, elas serão discutidas separadamente a seguir.
- Técnicas baseadas na teoria de predição linear. Estas técnicas compõem o tema desta tese e também serão discutidas separadamente, na próxima seção.

### 2.4.1 Técnicas de Bussgang

Conforme vimos, o cálculo da direção de adaptação no algoritmo LMS clássico usa o sinal de erro  $e(n)$ , e o cálculo deste erro pressupõe o conhecimento do sinal transmitido. Isto impossibilita o uso deste algoritmo no contexto de equalização cega. Por outro lado, se alguma estimativa do sinal transmitido puder ser feita, o sinal de erro poderá ser calculado, através da substituição do sinal transmitido por esta estimativa. É esta a motivação das técnicas de Bussgang: tentar usar algoritmos como o LMS no contexto de equalização cega. Para contornar a falta da sequência de treinamento, estas técnicas tentam, de alguma maneira, estimar o símbolo transmitido a partir das saídas do equalizador. Em geral, esta estimativa é sem memória, de forma que só o último valor de  $a'(n)$  é usado. O sinal de erro necessário para o algoritmo LMS é formado pela diferença entre a saída do equalizador e a estimativa do símbolo transmitido. Denotemos por  $\Gamma$  a função que realiza esta estimação. As técnicas de Bussgang usam o LMS, simplesmente substituindo  $e(n)$  por  $\Psi(a'(n)) = a'(n) - \Gamma(a'(n))$

O algoritmo de Bussgang propriamente dito tenta estimar o símbolo transmitido através de um estimador de máxima verossimilhança. Outras escolhas da função  $\Gamma$  determinam diferentes métodos com diferentes características (cf. [12]). Entre outros, podemos citar os de Sato [13], Godard [14] e Stop-and-Go [15]. Em todas estas técnicas, a escolha da função  $\Gamma$  é tal que a saída do equalizador satisfaz a condição de Bussgang, dada por:

$$E(a'(n)a'(n+m)) = E(a'(n)\Gamma(a'(n+m))) \quad (2.42)$$

para algum valor de  $m$ . É por esta razão que elas são conhecidas como técnicas de Bussgang.

Observe que quando os algoritmos clássicos estão no modo de decisão direta eles estão, na realidade, executando um algoritmo do tipo Bussgang, onde a estimação de símbolo transmitido é feita pela simples quantização da saída do equalizador e a função  $\Gamma$  é implementada pelo dispositivo de decisão. O algoritmo de decisão direta é uma técnica de Bussgang simples e de rápida convergência. Entretanto, sua convergência é garantida apenas quando sua condição inicial já abre o olho do canal (cf. [16]).

Outro ponto a ressaltar é que o estimador do símbolo transmitido tem que ser não linear. Conforme vimos, os algoritmos de Bussgang são simplesmente uma versão dos algoritmos adaptativos clássicos que substituem  $e(n) = a(n - \delta) - a'(n)$  por  $\Psi(a'(n)) = \Gamma(a'(n)) - a'(n)$ . Sabemos também que os algoritmos clássicos convergem para o mínimo de  $J(H) = E(|e(n)|^2)$ . Portanto, os algoritmos de Bussgang devem convergir para o mínimo da função custo

$$J(H) = E(|\Psi(a'(n))|^2). \quad (2.43)$$

Ora, se  $\Gamma$  é uma função linear, é fácil ver que  $J(H)$  envolve apenas as estatísticas de 2ª ordem do sinal. Entretanto, sabemos que a estimação cega do filtro inverso do canal deve usar as EOS e, portanto, o uso de uma estimativa linear é inviável.

Por outro lado, assumindo que  $\Psi$  é uma função não linear de  $a'(n)$ , a série de Taylor de  $|\Psi(a'(n))|^2$  possui componentes de ordem maior ou igual a dois, de forma que podemos escrever

$$J(H) = E(c_1 a'(n) + c_2 a'(n)^2 + c_3 a'(n)^3 + c_4 a'(n)^4 + \dots), \quad (2.44)$$

onde  $c_i$  é o *i*ésimo componente da expansão em série de Taylor de  $|\Psi(a'(n))|^2$ . Devido à linearidade do operador esperança, fica claro que  $J(H)$  é uma soma, ponderado por  $c_i$ , de

vários momentos de  $a'(n)$ . É por isto que se diz que as técnicas de Bussgang fazem uso indireto das estatísticas de ordem superior.

Em [10] está provado que os algoritmos de Bussgang são globalmente convergentes, desde que o comprimento do equalizador seja infinito. Em situações práticas, onde equalizadores de comprimento infinito não são realizáveis, os algoritmos de Bussgang podem apresentar mínimos locais, pois a função custo envolvida não é convexa. Apesar disto, as técnicas de Bussgang são largamente usadas na prática.

### Algoritmo de Godard

Apresentaremos aqui aquela que é talvez a mais popular técnica de equalização cega, o algoritmo de Godard (cf. [14]). Este algoritmo é baseado na função custo

$$J_G(H) = \frac{1}{2p} E \left( \left| |a'(n)|^p - R_p \right|^2 \right), \quad p = 1, 2, \dots \quad (2.45)$$

onde

$$R_p = \frac{E(|a(n)|^{2p})}{E(|a(n)|^p)}. \quad (2.46)$$

A minimização de  $J_G(H)$  pode ser vista como uma tentativa de minimizar a distância média entre a  $p^{\text{ésima}}$  potência do módulo de  $a'(n)$  e a circunferência de raio  $R_p$ . Note que  $R_p$  pode ser visto como o “raio médio” da constelação transmitida. Em outras palavras, o algoritmo de Godard tenta manter constante o módulo da saída do equalizador. Por esta razão, este algoritmo também é conhecido por *Algoritmo de Módulo Constante* (CMA, do inglês *Constant Modulus Algorithm*). Aparentemente, esta função custo só faz sentido quando a seqüência transmitida também tem módulo constante, como nas modulações 4-QAM e 2-PAM. Entretanto, este algoritmo se mostra eficiente na equalização de uma classe mais abrangente de constelações. Em sua versão mais usada,  $p = 2$  e o algoritmo fica

$$H(n+1) = H(n) - \mu \left( |a'(n)|^2 - R_2 \right) a'(n) X(n). \quad (2.47)$$

## 2.5 Técnicas Preditivas

Apesar de sua popularidade, as técnicas de Bussgang apresentam algumas desvantagens. Uma delas, já mencionada, é a possível existência de mínimos locais. Ou seja, os equalizadores baseados nas técnicas de Bussgang podem convergir para um valor que não leva à equalização do canal.

Outra desvantagem se refere à estrutura de filtragem usada por estas técnicas, os filtros FIR. Apesar dos filtros FIR levarem a algoritmos simples, os equalizadores baseados nestes filtros possuem um desempenho que se degrada à medida que os zeros da função de transferência do canal se aproximam da Circunferência de Raio Unitário (CRU) (cf. [4, 5]). Para canais com zeros próximos à CRU, outras estruturas, como o DFE, possuem melhor desempenho.

As técnicas preditivas se propõem a solucionar estes problemas, bem como fornecer uma nova abordagem para o problema da equalização cega. Elas são baseadas na teoria da predição e retropredição linear. Os preditores lineares são filtros que tentam, a partir das

últimas  $L$  amostras do sinal na sua entrada, prever a amostra seguinte. De maneira análoga, um retropreditor linear é um filtro que tenta, a partir das últimas  $N$  amostras do sinal na sua entrada, “prever” (talvez “recordar” fosse um termo mais apropriado) a amostra no instante  $n - N$ . Assim, seja  $x(n)$  um sinal qualquer a ser predito,  $B$  e  $C$  os vetores de coeficientes de um retropreditor FIR e de um preditor FIR, respectivamente. Suas saídas  $r(n)$  e  $p(n)$ , respectivamente, são dadas por

$$r(n) = B^H X_R(n) \quad (2.48)$$

$$p(n) = C^H X_P(n-1), \quad (2.49)$$

onde

$$X_R(n) = [x(n), \dots, x(n-N+1)]^T \quad (2.50)$$

$$X_P(n-1) = [x(n-1), \dots, x(n-L)]^T. \quad (2.51)$$

O objetivo do preditor é fazer de  $p(n)$  a melhor estimativa possível de  $x(n)$  enquanto que o do retropreditor é fazer de  $r(n)$  a melhor estimativa possível de  $x(n-N)$ . Observe que  $p(n)$  só usa os valores de  $x$  até o instante  $n-1$ , enquanto  $r(n)$  só usa valores de  $x$  a partir de  $n-N+1$ .

Estes filtros estão detalhadamente descritos em [8, 17]. Nestas referências, desenvolvem-se algoritmos adaptativos para os preditores, e demonstra-se que os erros de predição, dados por  $x(n) - p(n)$ , no caso do preditor, e por  $x(n-N) - r(n)$ , no caso do retropreditor, tendem a ser sinais descorrelacionados. Ou seja, os filtros cuja saída é o erro de predição, chamados filtros de erro de predição, são filtros branqueadores. Como assumimos que o sinal transmitido  $\{a(n)\}$  é i.i.d. (independente e identicamente distribuído, ou seja, branco), a saída do equalizador também deve sê-lo. Assim, a propriedade de branqueamento é exigida dos equalizadores, daí o interesse nos filtros de erro de predição. Na seqüência, abusaremos da linguagem e usaremos o termo preditor para nos referirmos indistintamente aos preditores e aos filtros de erro de predição.

Uma outra propriedade importante destes filtros é que os preditores correspondem a filtros de fase mínima enquanto os retropreditores correspondem a filtros de fase máxima<sup>3</sup>, e que estes são únicos. Imagine, então, um canal de fase mínima pelo qual será transmitida uma seqüência  $\{a(n)\}$  branca. O filtro inverso deste canal, o equalizador, deve, a partir da saída do canal, gerar um sinal igualmente branco em sua saída. Mais ainda, este filtro também deve ser de fase mínima. Ora, um filtro de fase mínima com saída branca caracteriza um filtro de erro de predição, e como os preditores são únicos, podemos dizer que o equalizador de um canal de fase mínima corresponde ao preditor da saída do canal. O algoritmo de adaptação de um preditor é dado por

$$e(n) = x(n) - p(n) \quad (2.52)$$

$$C(n+1) = C(n) + \mu X_P(n-1) e^*(n) \quad (2.53)$$

e não envolve nenhuma informação sobre o sinal transmitido. Assim, a predição linear pode ser considerada uma técnica de equalização cega para canais de fase mínima.

<sup>3</sup>Um sistema de fase máxima é caracterizado por possuir todos os seus polos e zeros no exterior da circunferência de raio unitário. Obviamente, um sistema de fase máxima só é estável e causal se não possuir polos.

Restam algumas considerações sobre o uso de preditores lineares como equalizadores cegos de canais de fase mínima. Em primeiro lugar, um filtro de erro de predição não é capaz de combater ganhos de potência e fase introduzidos pelo canal, sendo necessário, para isso, o uso de um CAG na saída do filtro.

Além disso, assume-se, em geral, que um canal de telecomunicações possui apenas zeros, de forma que o equalizador ideal deve possuir apenas polos. Assim, o uso de filtros FIR fornece apenas uma solução aproximada para o problema de equalização. A solução exata envolve o uso de filtros IIR (do inglês *Infinite Impulse Response*). Um filtro IIR é caracterizado pela presença de realimentações e, conseqüentemente, de polos. A figura 2.4 mostra o esquema de um preditor IIR.

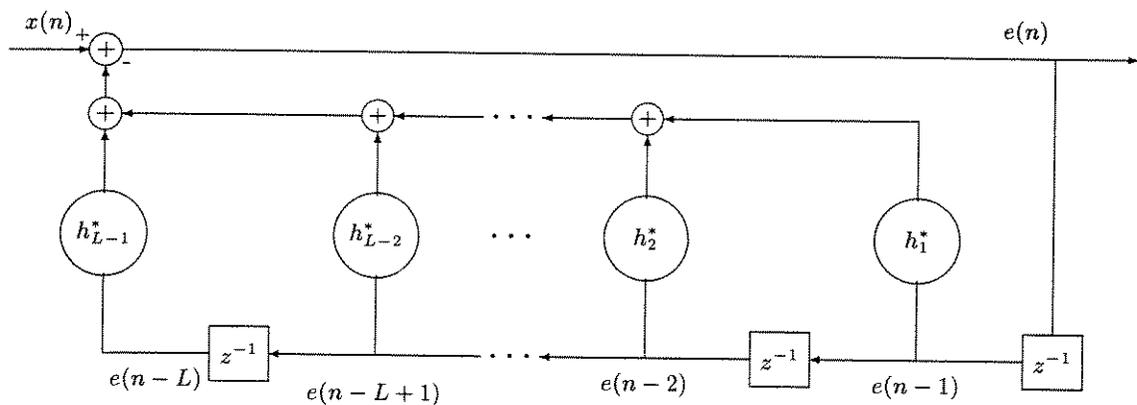


Figura 2.4: Filtro de Erro de Predição IIR.

O erro de predição deste filtro é dado por:

$$e(n) = x(n) - C^H E(n-1), \quad (2.54)$$

onde  $E(n-1) = [e(n-1), \dots, e(n-L)]$ . Observe que este filtro tenta estimar  $x(n)$  a partir do vetor  $E(n-1)$ , que não possui nenhuma informação sobre  $x(n)$ . Assim, ele também funciona como um preditor. Os preditores IIR serão preferidos aos FIR devido à sua capacidade de inverter perfeitamente um canal que apresente apenas zeros, o que corresponde ao modelo mais usual para os canais de telecomunicações.

De maneira análoga ao preditor, um retropreditor pode ser usado como equalizador cego de um canal de fase máxima. Note que tanto o canal quanto o retropreditor são sistemas de fase máxima, de forma que, para serem causais e estáveis, não podem apresentar polos. Conseqüentemente, um retropreditor não pode ser implementado como um filtro IIR e a equalização neste caso não pode ser exata.

Na prática, os canais de comunicações são de fase mista, apresentando polos e zeros dentro e fora de CRU. Assim, o uso isolado de preditores e retropreditores não é suficiente para equalizar o canal. Na seqüência, veremos duas técnicas de equalização cega para canais de fase mista usando princípios de predição linear. Ambas foram desenvolvidas em [3].

### 2.5.1 Predição e Retropredição

Conforme vimos, a maioria dos canais de telecomunicações é de fase mista, apresentando zeros no interior e no exterior da circunferência de raio unitário, e não pode ser equalizado

com o uso isolado de um preditor ou de um retropreditor. Entretanto, sabemos que todo sistema linear pode ser decomposto como uma cascata de um sistema de fase mínima e um de fase máxima (cf. [9]). Surge então, a idéia de usar uma cascata de um retropreditor e de um preditor para tentar equalizar um canal genérico. O preditor equalizaria a parte de fase mínima do canal e o retropreditor se encarregaria da parte de fase máxima. Este esquema está mostrado na figura 2.5

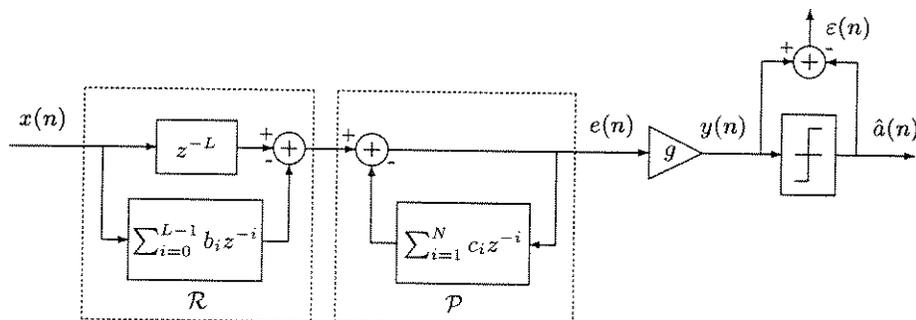


Figura 2.5: Equalizador Baseado em um Retropreditor e um Preditor

A definição de um critério a ser minimizado nas adaptações dos coeficientes deste equalizador buscou levar em conta a simplicidade. Sabe-se que o equalizador deve branquear o sinal na sua saída, e que a minimização do erro de predição leva a este branqueamento. Assim, o erro de predição deve ser considerado na formação da função custo. Sabe-se também que o equalizador funcionando no modo de decisão direta converge para a solução ótima a partir de uma condição de olho aberto. Ou seja, o erro DD também pode ser levado em consideração no cálculo da função custo. Desta forma, optou-se por uma função custo que leva em conta apenas dois erros quadráticos médios: o erro de predição ( $e(n)$ ) e o de decisão direta ( $\varepsilon(n)$ ), o que colabora para a simplicidade dos algoritmos associados à estrutura.

Estes dois erros devem ser combinados de alguma forma na definição da função custo. Ora, se o erro de decisão direta for muito grande, a probabilidade de que o símbolo decidido esteja correto e, portanto, que o olho do canal esteja aberto, é baixa. Nesta condição, deve-se dar um maior peso ao critério de branqueamento (minimizar o erro de predição), deixando de lado o erro de decisão direta. À medida que o erro de decisão direta diminui, o olho do canal se abre. Nesta circunstância, a minimização do critério DD já pode levar à equalização do canal, e o erro de decisão direta deve passar a ter uma maior participação na formação da função custo. Com base neste raciocínio, define-se o seguinte critério de minimização:

$$J_{\alpha}(H) = \alpha \gamma^2 J_P(H) + (1 - \alpha) J_{\varepsilon}(H), \quad (2.55)$$

onde  $J_P(H)$  é a função custo associada ao critério de branqueamento:

$$J_P(H) = E \left( |e(n)|^2 \right), \quad (2.56)$$

$J_{\varepsilon}(H)$  é a função custo associada ao critério de decisão direta:

$$J_{\varepsilon}(H) = E \left( |\varepsilon|^2 \right) \quad (2.57)$$

e a constante  $\gamma^2$  está presente para assegurar a homogeneidade do critério.

O fator  $\alpha$  é o responsável pela transição entre o critério de branqueamento e o de decisão direta. Seu valor deve variar entre 0 e 1, sendo próximo de 1 quando o critério de branqueamento merecer um maior peso e próximo de 0 quando maior peso for dado ao critério de decisão direta. Assim, quando o erro de decisão direta for grande,  $\alpha$  deve ser próximo de 1, diminuindo à medida que  $\varepsilon(n)$  diminui. Define-se então:

$$\alpha_n = \tanh\left(|\varepsilon(n)|^2\right). \quad (2.58)$$

Em [3] estão apresentadas as equações para adaptação dos coeficientes dos preditores e do ganho complexo, bem como está demonstrada a unimodalidade deste critério. Resultados de simulação mostram que esta técnica consegue equalizar canais para os quais as técnicas de Bussgang falham. Além disto, sua velocidade de convergência mostrou-se superior à dos algoritmos de Bussgang e dos que usam diretamente as EOS.

Uma desvantagem desta técnica é sua incapacidade de equalizar canais com nulos espectrais. Sabe-se que, nestes casos, alguma não linearidade deve ser introduzida pelo equalizador, o que é feito em geral pelo uso de uma estrutura realimentada por decisão, o DFE (cf. [4, 5]). Uma possível solução para este problema é a introdução do quantizador na malha de decisão do preditor na figura 2.5, tornando a estrutura semelhante à de um DFE. O desempenho do equalizador, neste caso, fica muito ligado à correção dos símbolos decididos e, por isto, é pior do que o esquema original mesmo em situações de nulos espectrais. Na seqüência, apresentaremos uma outra técnica preditiva que contorna este problema.

### 2.5.2 Equalização de Magnitude e Fase

Esta técnica de equalização baseia-se no uso de filtros passa-tudo (cf. [9]). Um sistema passa-tudo é caracterizado pelo fato de que o módulo de sua resposta em frequência é constante e igual a 1. Sua função de transferência é dada por:

$$H(z) = e^{j\theta} \frac{z^{-L} + \sum_{i=1}^L h_i^* z^{-L+i}}{1 + \sum_{i=1}^L h_i z^{-i}} \quad (2.59)$$

Assumiremos, como sempre, que o sinal transmitido  $a(n)$  seja branco. Ou seja, seu espectro de potência é plano. Conforme vimos, a saída de um preditor também é branca. Assim, se colocarmos um preditor na saída do canal, o módulo da resposta em frequência da saída do preditor é igual ao do sinal transmitido e estes sinais só diferem pela fase de suas respostas em frequência. Para tentar recuperar o sinal transmitido, poderíamos, então, passar o sinal que sai do preditor por um filtro que não altere sua resposta em amplitude (que já tem a forma correta), mas corrija sua resposta em fase. Ora, filtros passa-tudo possuem justamente a característica de não alterar a resposta em magnitude do sinal em sua entrada. Podemos pensar, assim, em um equalizador composto pela cascata de um preditor linear, que pode ser visto como um equalizador de magnitude, e de um filtro passa-tudo, que pode ser visto como um equalizador de fase. Seu funcionamento pode ser melhor compreendido quando consideramos a decomposição do canal como um sistema de fase mínima em cascata com um sistema passa-tudo. Desta forma, a parte de fase mínima do canal seria compensada pelo preditor linear enquanto sua parte passa-tudo seria compensada pelo equalizador de fase.

A estrutura do equalizador proposto está mostrada na figura 2.6. Note que, nesta figura, o dispositivo de decisão está introduzido na malha de realimentação do passa-tudo. Como isto torna o passa-tudo muito semelhante a um DFE, já estamos prevendo a aplicação deste equalizador em canais com nulos espectrais severos.

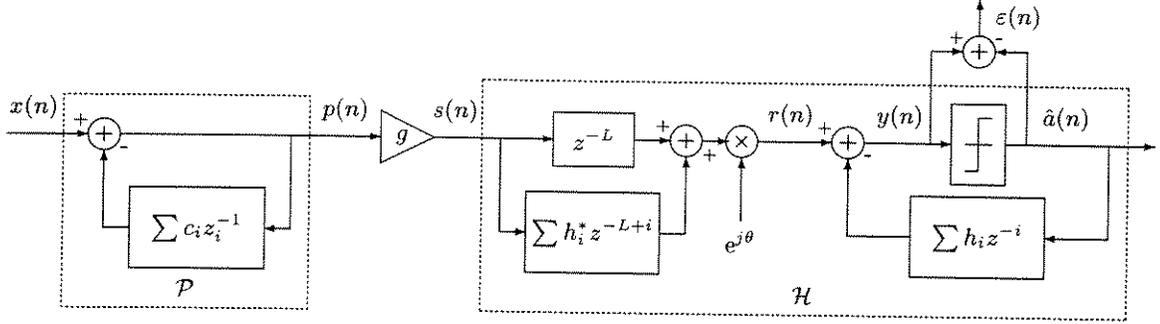


Figura 2.6: Equalizadores de Magnitude e Fase em Cascata

Uma propriedade interessante desta estrutura é que os parâmetros do preditor e do filtro passa-tudo podem ser adaptados de maneira independente. Ao preditor, cabe apenas branquear o sinal transmitido. Desta forma, seus parâmetros são adaptados de forma a minimizar a média quadrática do erro de predição, e o algoritmo de adaptação LMS resultante é dado por:

$$C(n) = C(n-1) + \mu p(n) P^*(n-1), \quad (2.60)$$

onde  $C = [c_1, \dots, c_L]^T$  é o vetor de coeficientes do preditor IIR e  $P(n-1) = [p(n-1), \dots, p(n-L)]^T$ .

Note que, da mesma forma que no equalizador baseado na predição e retropredição, o ganho do preditor linear e do passa-tudo é fixo. Desta forma, para possibilitar a equalização de um canal genérico, adota-se um CAG em cascata com o preditor. O CAG busca igualar a energia do sinal na saída do preditor com a do sinal transmitido. Assim, seu algoritmo de adaptação fica

$$G(n) = G(n-1) + \lambda \left( E(|a(n)|^2) - |s(n)|^2 \right) \quad (2.61)$$

$$g(n) = \sqrt{G(n)} \quad (2.62)$$

Para a adaptação do equalizador de fase, devemos escolher um critério a ser otimizado. Felizmente, o critério mais evidente apresenta um ótimo desempenho. Simulações deste equalizador utilizando um simples critério de decisão direta (minimizar  $J(H) = E(|\epsilon(n)|^2)$ ) mostraram uma ótima velocidade de convergência desta estrutura, além de uma capacidade de equalizar mesmo canais com profundos nulos espectrais. Além disso, utilizando-se certas hipóteses simplificadoras, prova-se que este é um critério unimodal (cf. [3]). O algoritmo LMS que leva à minimização deste critério é dado por

$$\xi_j(n) = \epsilon^*(n) e^{j\theta(n)} s(n-N+j) - \epsilon(n) \hat{a}^*(n-j) \quad (2.63)$$

$$H(n+1) = H(n) - \kappa \Xi(n), \quad (2.64)$$

Vamos desenvolver agora uma seqüência de raciocínio que nos levará ao algoritmo RLS para o equalizador de fase. Esta seqüência é muito semelhante àquela seguida na dedução do RLS clássico. Inicialmente, observe que a equação 3.18 pode ser reescrita como

$$R(n) = w R(n-1) + Y(n) Y^T(n). \quad (3.20)$$

Podemos agora aplicar o Lema de Inversão de Matrizes apresentado no capítulo anterior. Consideremos, então

$$\begin{aligned} A &= R(n) \\ B &= w R(n-1) \\ C &= Y(n) \\ D &= I_2 \end{aligned}$$

onde  $I_2$  corresponde à matriz identidade de ordem 2. Em seguida, definimos

$$A^{-1} = P(n) = R^{-1}(n) \quad (3.21)$$

$$B^{-1} C (D + C^H B^{-1} C)^{-1} = K(n) = P(n-1) Y(n) (w I_2 + Y^T(n) P(n-1) Y(n))^{-1}. \quad (3.22)$$

Assim, aplicando o lema de inversão de matrizes à equação 3.20, chegamos, finalmente, a uma recursão para  $P(n) = R^{-1}(n)$ :

$$P(n) = w^{-1} P(n-1) - w^{-1} K(n) Y^T(n) P(n-1). \quad (3.23)$$

Esta recursão corresponde a uma das equações do algoritmo RLS. Resta agora determinar uma recursão para o vetor de coeficientes.

Para isto, precisaremos usar uma propriedade importante do vetor  $K(n)$ , que será obtida agora. Assim, multiplicando ambos os lados de 3.22 por  $(w I_2 + Y^T(n) P(n-1) Y(n))$ , concluímos que

$$K(n) (w I_2 + Y^T(n) P(n-1) Y(n)) = P(n-1) Y(n) \quad (3.24)$$

e, portanto, que

$$\begin{aligned} K(n) &= w^{-1} P(n-1) Y(n) - w^{-1} K(n) Y^T(n) P(n-1) Y(n) \\ &= (w^{-1} P(n-1) - w^{-1} K(n) Y^T(n) P(n-1)) Y(n) \end{aligned}$$

Ora, o primeiro termo do lado direito da equação acima corresponde à recursão para  $P(n)$  obtida na equação 3.23. Assim, concluímos que

$$K(n) = P(n) Y(n) \quad (3.25)$$

onde  $H = [h_1, \dots, h_N]^T$  é o vetor de coeficientes do equalizador de fase e  $\Xi = [\xi_1, \dots, \xi_N]^T$  é um vetor auxiliar.

Observe que na definição de um filtro passa-tudo dada pela equação 2.59 existe um ganho de fase  $\theta$ . Este ganho está associado à fase da portadora do sinal e também deve ser adaptado, de forma a evitar erros de decisão direta muito elevados. O algoritmo de adaptação do ajuste de fase  $\theta(n)$  é dado por

$$\theta(n+1) = \theta(n) + \eta \Im(\varepsilon^*(n) r(n)), \quad (2.65)$$

onde  $\Im$  indica a parte imaginária.

## 2.6 Conclusão

Concluimos assim este capítulo introdutório ao problema de equalização cega.

Inicialmente, apresentamos os sistemas de transmissão digital, mostrando suas características principais e explicando o problema da Interferência Inter-Simbólica. Em seguida, mostramos que a IES pode ser vista como uma consequência de uma filtragem linear.

Com isto em vista, uma maneira de combater a IES consiste em usar um filtro inverso ao que causa esta interferência. Este filtro é chamado equalizador. Mostramos que os equalizadores são filtros digitais cujos coeficientes minimizam alguma função custo associada à IES na saída do equalizador. Vimos também que a aplicação de métodos iterativos de minimização levam às técnicas adaptativas.

Em seguida, mostramos que as técnicas adaptativas clássicas necessitam de uma sequência de treinamento, o que diminui a eficiência do sistema. Para contornar este problema, apresentamos as técnicas de equalização cega. Mostramos algumas propriedades destas técnicas e discutimos os mais populares algoritmos de equalização cega, os algoritmos de Bussgang.

Finalmente, apresentamos as técnicas preditivas de equalização cega. Mostramos inicialmente alguns princípios de predição linear que motivaram o desenvolvimento destas técnicas. Apresentamos, em seguida, os equalizadores por predição e retropredição e os equalizadores de magnitude e fase, duas estruturas de equalização baseadas em técnicas preditivas. Discutimos com maior profundidade os equalizadores de magnitude e fase, por se tratarem da mais eficiente estrutura de equalização cega baseada em técnicas preditivas. No contexto desta tese, esta é a seção mais importante deste capítulo. Na sequência, desenvolveremos novos algoritmos para os equalizadores de magnitude e fase, e estudaremos seu desempenho.

## Capítulo 3

# Novos Algoritmos para Técnicas Preditivas

NO capítulo anterior, apresentamos uma estrutura de equalização cega baseada na cascata de equalizadores de magnitude e fase, mostrada na figura 2.6. Apresentamos também algoritmos de adaptação para os coeficientes desta estrutura. Os algoritmos apresentados foram obtidos a partir da aplicação do método do gradiente estocástico, discutido na seção 2.3.1, à minimização das funções custo associadas à referida estrutura. A proposta do presente capítulo é aplicar as outras técnicas de otimização apresentadas na seção 2.3.1, os métodos RLS e GN, na minimização destas funções custo. Assim, na seção 1 desenvolveremos algoritmos RLS para adaptação dos coeficientes dos equalizadores de magnitude e fase. Na seção 2, desenvolveremos algoritmos GN para estas estruturas.

### 3.1 Os Algoritmos RLS

Apresentaremos nesta seção o desenvolvimento dos algoritmos RLS para a adaptação dos coeficientes dos equalizadores de magnitude e fase. O desenvolvimento destes algoritmos é baseado na equação 2.18, onde se define a função custo a ser minimizada nos algoritmos RLS. Quando comparamos a função custo dos algoritmos RLS com a do LMS, vemos que o RLS busca minimizar a média temporal do sinal de erro, enquanto o LMS busca minimizar sua esperança estatística (ver seção 2.3.1). Conforme vimos, para a estrutura de equalização em questão, busca-se minimizar duas funções custo: uma baseada no erro de predição, no caso do preditor IIR, e a outra baseada no erro de decisão direta, no caso de equalizador de fase e do ajuste de fase ( $\theta$ ).

Inicialmente, vamos desenvolver os algoritmos RLS para o equalizador de fase e o ajuste de fase. Os algoritmos LMS desenvolvidos anteriormente para estas estruturas buscavam minimizar o erro DD quadrático médio,  $J(H) = E(|\varepsilon(n)|^2)$ . Assim, os algoritmos RLS devem buscar valores dos parâmetros  $H$  do equalizador de fase e do parâmetro  $\theta$  que minimizem

$$\mathcal{E}_{DD}(n) = \sum_{i=1}^n w^{n-i} |\varepsilon(i, n)|^2. \quad (3.1)$$

O erro de decisão direta  $\varepsilon(i, n)$  é dado por

$$\varepsilon(i, n) = y(i, n) - \hat{a}(i), \quad (3.2)$$

onde  $y(i, n)$  é o valor do sinal na entrada do dispositivo de decisão obtido a partir das entradas presentes no instante  $i$ , mas supondo o uso dos parâmetros  $H$  e  $\theta$  calculados no instante  $n$ .

### 3.1.1 Equalizador de Fase

Da figura 2.6, podemos escrever vetorialmente  $y(i, n)$  como

$$y(i, n) = s(i - L) e^{j\theta i} + H^H(n) S(i) - H^T(n) A(i), \quad (3.3)$$

onde

$$\begin{aligned} H(n) &= [h_1(n) \dots h_L(n)]^T \\ A(i) &= [\hat{a}(i - 1) \dots \hat{a}(i - L)]^T \\ S(i) &= e^{j\theta i} [s(i - L + 1) s(i - L + 2) \dots s(i)]^T \end{aligned} \quad (3.4)$$

são, respectivamente, o vetor de coeficientes do equalizador de fase, o vetor com as últimas  $L$  saídas do dispositivo de decisão e o vetor de entrada do equalizador de fase, já considerando o ajuste de fase  $\theta$ .

Observe que, na equação 3.3, aparecem tanto o vetor de coeficientes quanto o seu conjugado. Matematicamente, problemas desta forma são tratados através da decomposição das variáveis envolvidas em sua parte real e imaginária. Usaremos respectivamente os índices  $\Re$  e  $\Im$  para denotar estas partes. Assim,  $\varepsilon(i, n) = y(i, n) - \hat{a}(i)$  pode ser escrito como:

$$\varepsilon(i, n) = s(i - L) e^{j\theta i} + \left( H_{\Re}^T(n) - j H_{\Im}^T(n) \right) S(i) - \left( H_{\Re}^T(n) + j H_{\Im}^T(n) \right) A(i) - \hat{a}(i). \quad (3.5)$$

O termo  $s(i - L) e^{j\theta i} - \hat{a}(i)$  não depende do vetor de coeficientes e, portanto, não influencia o processo de minimização. Desta forma, definiremos uma variável independente

$$d(i) = \hat{a}(i) - s(i - L) e^{j\theta i}. \quad (3.6)$$

Definindo também

$$\mathcal{H}(n) = \begin{bmatrix} H_{\Re}(n) \\ H_{\Im}(n) \end{bmatrix} \quad \text{e} \quad \mathcal{Y}(i) = \begin{bmatrix} S(i) - A(i) \\ -j S(i) - j A(i) \end{bmatrix}, \quad (3.7)$$

podemos reescrever a equação 3.5 como

$$\varepsilon(i, n) = \mathcal{H}^T(n) \mathcal{Y}(i) - d(i) = \mathcal{Y}(i)^T \mathcal{H}(n) - d(i). \quad (3.8)$$

Lembrando que  $|\varepsilon(n)|^2 = \varepsilon(n) \varepsilon^*(n)$ , o critério a ser minimizado (equação 3.1) pode ser escrito como

$$\mathcal{E}_{DD}(n) = \sum_{i=1}^n w^{n-i} \left[ |d(i)|^2 - d^*(i) \mathcal{Y}^T(i) \mathcal{H}(n) - d(i) \mathcal{Y}^H(i) \mathcal{H}(n) + \mathcal{H}^T(n) \mathcal{Y}(i) \mathcal{Y}^H(i) \mathcal{H}(n) \right] \quad (3.9)$$

Os termos  $d^*(i) \mathcal{Y}^T(i) \mathcal{H}(n)$  e  $d(i) \mathcal{Y}^H(i) \mathcal{H}(n)$  são complexos conjugados, de forma que

$$d^*(i) \mathcal{Y}^T(i) \mathcal{H}(n) + d(i) \mathcal{Y}^H(i) \mathcal{H}(n) = 2 \Re\{d^*(i) \mathcal{Y}(i)\} \mathcal{H}, \quad (3.10)$$

que é um valor real.

Além disso, é fácil verificar que

$$\mathcal{Y}(i) \mathcal{Y}^H(i) = \mathcal{Y}_{\Re}(i) \mathcal{Y}_{\Re}^T(i) + \mathcal{Y}_{\Im}(i) \mathcal{Y}_{\Im}^T(i). \quad (3.11)$$

Se definirmos

$$Y(i) = [\mathcal{Y}_{\Re}(i) \quad \mathcal{Y}_{\Im}(i)] \quad (3.12)$$

teremos, então,

$$\mathcal{Y}(i) \mathcal{Y}^H(i) = Y(i) Y^T(i), \quad (3.13)$$

que também é um valor real.

Substituindo estes resultados na equação 3.9, chegamos finalmente a

$$\mathcal{E}_{DD}(n) = \sum_{i=1}^n w^{n-i} \left[ |d(i)|^2 - 2 \Re\{d^*(i) \mathcal{Y}(i)\} \mathcal{H}(n) + \mathcal{H}^T(n) Y(i) Y^T(i) \mathcal{H}(n) \right]. \quad (3.14)$$

Os vetores  $\mathcal{Y}(i)$  e  $Y(i)$  na equação acima envolvem valores de  $\hat{a}(i)$ . Estes valores correspondem à quantização da saída do filtro passa-tudo e são, portanto, função de seus coeficientes. Levar esta dependência em consideração no processo de otimização implicaria em não-linearidades que tornariam os cálculos impraticáveis. Entretanto, os valores de  $\hat{a}(i)$  são constantes salvo em um número finito de pontos correspondentes aos limiares de decisão. Como a probabilidade de  $y(i)$ , à entrada do dispositivo de decisão, assumir algum destes valores é nula, podemos considerar que:

$$\frac{\partial \hat{a}(i)}{\partial h_j} = 0. \quad (3.15)$$

Em outras palavras, podemos considerar que os símbolos decididos não dependem do valor dos coeficientes do equalizador. Note que esta mesma hipótese é feita na dedução dos algoritmos LMS e RLS para um DFE (cf. [4, 5]).

Usando a hipótese acima, concluímos que os vetores  $\mathcal{Y}(i)$  e  $Y(i)$  não dependem de  $\mathcal{H}$ . Assim, temos que a equação 3.14 corresponde a uma espécie de forma quadrática em  $\mathcal{H}$ , que é um vetor de coeficientes de valores reais. Seu gradiente pode ser facilmente calculado e chegamos, finalmente, a

$$\nabla_{\mathcal{H}} \mathcal{E}_{DD}(n) = \sum_{i=1}^n w^{n-i} \left[ 2 Y(i) Y^T(i) \mathcal{H}(n) - 2 \Re\{d^*(i) \mathcal{Y}(i)\} \right]. \quad (3.16)$$

O valor de  $\mathcal{H}(n)$  que minimiza  $\mathcal{E}_{DD}(n)$  satisfaz  $\nabla_{\mathcal{H}} \mathcal{E}_{DD}(n) = 0$ , e deve, portanto, satisfazer

$$R(n) \mathcal{H}(n) = b(n) \implies \mathcal{H}(n) = R^{-1}(n) b(n) \quad (3.17)$$

onde definimos

$$R(n) = \sum_{i=1}^n w^{n-i} Y(i) Y^T(i) \quad (3.18)$$

$$b(n) = \sum_{i=1}^n w^{n-i} \Re\{d^*(i) \mathcal{Y}(i)\}. \quad (3.19)$$

Podemos agora obter a recursão para  $\mathcal{H}(n)$ . Para começar, notemos que a equação 3.19 pode ser reescrita como

$$\begin{aligned} b(n) &= wb(n-1) + \text{Re}\{d^*(n)\mathcal{Y}(n)\} \\ &= wb(n-1) + d_{\mathbb{R}}(n)\mathcal{Y}_{\mathbb{R}}(n) + d_{\mathbb{I}}(n)\mathcal{Y}_{\mathbb{I}}(n) \\ &= wb(n-1) + Y(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix} \end{aligned}$$

Podemos então aplicar a recursão acima à equação 3.17, de forma a escrever

$$\begin{aligned} \mathcal{H}(n) &= P(n)b(n) \\ &= wP(n)b(n-1) + P(n)Y(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix}. \end{aligned}$$

Mas  $P(n)Y(n) = K(n)$ . Usando este fato e a recursão para  $P(n)$  obtida em 3.23, temos que

$$\begin{aligned} \mathcal{H}(n) &= wP(n)b(n-1) + K(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix} \\ &= w \left( w^{-1}P(n-1) - w^{-1}K(n)Y^T(n)P(n-1) \right) b(n-1) + K(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix} \\ &= P(n-1)b(n-1) - K(n)Y^T(n)P(n-1)b(n-1) + K(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix}. \end{aligned}$$

Observe que  $P(n-1)b(n-1) = \mathcal{H}(n-1)$ . Assim,

$$\begin{aligned} \mathcal{H}(n) &= \mathcal{H}(n-1) - K(n)Y^T(n)\mathcal{H}(n-1) + K(n) \begin{bmatrix} d_{\mathbb{R}}(n) \\ d_{\mathbb{I}}(n) \end{bmatrix} \\ &= \mathcal{H}(n-1) - K(n) \begin{bmatrix} \mathcal{Y}_{\mathbb{R}}(n)\mathcal{H}(n-1) - d_{\mathbb{R}}(n) \\ \mathcal{Y}_{\mathbb{I}}(n)\mathcal{H}(n-1) - d_{\mathbb{I}}(n) \end{bmatrix}. \end{aligned}$$

Finalmente, podemos nos referir à equação 3.8 e notar que  $\mathcal{Y}_{\mathbb{R}}(n)\mathcal{H}(n-1) - d_{\mathbb{R}}(n)$  e  $\mathcal{Y}_{\mathbb{I}}(n)\mathcal{H}(n-1) - d_{\mathbb{I}}(n)$  correspondem, respectivamente, às partes reais e imaginárias do erro de decisão direta observado no instante  $n$ ,  $\varepsilon(n, n)$ . Por comodidade, denotaremos  $\varepsilon(n, n)$  simplesmente por  $\varepsilon(n)$ . Assim, chegamos finalmente a uma recursão para  $\mathcal{H}(n)$ :

$$\mathcal{H}(n) = \mathcal{H}(n-1) - K(n) \begin{bmatrix} \varepsilon_{\mathbb{R}}(n) \\ \varepsilon_{\mathbb{I}}(n) \end{bmatrix}. \quad (3.26)$$

Este algoritmo está resumido na tabela 3.1. Nela, foi introduzida uma matriz auxiliar  $U(n) = P(n-1)Y(n)$ . Isto nos permite usar a simetria de  $P(n)$  e reduzir um pouco o esforço computacional do algoritmo, eliminando a necessidade de calcular, em operações distintas, os produtos  $P(n-1)Y(n)$  e  $Y^T(n)P(n-1)$ , pois um é o transposto do outro.

$$\begin{aligned}
& \text{Disponível no instante } n \\
& P(n-1), \mathcal{H}(n-1), Y(n), \mathcal{Y}(n) \text{ e } d(n) \\
& \text{Cálculos} \\
& \varepsilon(n) = \mathcal{H}^T(n-1) \mathcal{Y}(n) - d(n) \\
& U(n) = P(n-1) Y(n) \\
& k(n) = (w I_2 + Y^T(n) U(n))^{-1} \\
& P(n) = \frac{1}{w} (P(n-1) - U(n) k(n) U^T(n)) \\
& \mathcal{H}(n) = \mathcal{H}(n-1) - U(n) k(n) \begin{bmatrix} \varepsilon_{\Re}(n) \\ \varepsilon_{\Im}(n) \end{bmatrix}
\end{aligned}$$

Tabela 3.1: Algoritmo RLS para o Filtro Passa-Tudo

### 3.1.2 Ajuste de Fase

Os cálculos envolvendo a dedução de um algoritmo RLS para o ajuste de fase ( $\theta$ ) são bem mais simples que os apresentados na seção anterior. Inicialmente, devemos definir

$$x(i) = r(i) e^{-j\theta i}, \quad (3.27)$$

onde  $r(i)$  é o sinal na saída do ajuste de fase mostrado na figura 2.6. Observe que a definição acima executa a operação inversa à passagem do sinal pelo ajuste de fase. Assim,  $x(i)$  corresponde ao sinal na entrada do ajuste de fase e, portanto, não depende de  $\theta$ .

A equação 3.27 nos permite reescrever a equação 3.2 como

$$\varepsilon(i, n) = x(i) e^{j\theta n} - H^T(i) A(i) - \hat{a}(i). \quad (3.28)$$

Lembrando novamente que  $|\varepsilon(n)|^2 = \varepsilon(n) \varepsilon^*(n)$ , o critério a ser minimizado (equação 3.1) fica

$$\begin{aligned}
\mathcal{E}_{DD}(n) &= \sum_{i=1}^n w^{n-i} \left[ |x(i)|^2 - x^*(i) e^{-j\theta n} (H^T(i) A(i) + \hat{a}(i)) \right. \\
&\quad \left. - x(i) e^{j\theta n} (H^H(i) A^*(i) + a^*(i)) + |H^T(i) A(i) + \hat{a}(i)|^2 \right]
\end{aligned}$$

Repetindo a hipótese feita anteriormente (equação 3.15), assumiremos que:

$$\frac{\partial \hat{a}(i)}{\partial \theta} = 0. \quad (3.29)$$

Assumiremos também que o vetor de coeficientes do equalizador de fase não depende de  $\theta$ .

Assim,

$$\begin{aligned}
\frac{\partial \mathcal{E}_{DD}(n)}{\partial \theta} &= \sum_{i=1}^n \left[ j w^{n-i} x^*(i) e^{-j\theta n} (H^T(i) A(i) + \hat{a}(i)) \right. \\
&\quad \left. - j w^{n-i} x(i) e^{j\theta n} (H^H(i) A^*(i) + a^*(i)) \right].
\end{aligned} \quad (3.30)$$

O valor de  $\theta_n$  que minimiza o critério da equação 3.28 satisfaz

$$\frac{\partial \mathcal{E}_{DD}(n)}{\partial \theta} = 0 \quad (3.31)$$

o que é equivalente a

$$e^{-j\theta_n} z(n) = e^{j\theta_n} z^*(n), \quad (3.32)$$

onde definimos

$$z(n) = \sum_{i=1}^n w^{n-i} \left( H^T(i) A(i) + \hat{a}(i) \right) x^*(i). \quad (3.33)$$

A equação 3.32 nos permite concluir que

$$e^{2j\theta_n} = \frac{z(n)}{z^*(n)} = e^{2j\angle z(n)}, \quad (3.34)$$

de forma que obtemos:

$$\theta_n = \angle z(n). \quad (3.35)$$

Observando que  $z(n)$  pode ser calculado recursivamente como

$$z(n) = w z(n-1) + \left( H^T(n) A(n) + \hat{a}(n) \right) x^*(n), \quad (3.36)$$

chegamos ao algoritmo que está resumido na tabela 3.2

Disponível no instante $n$	$z(n-1), \theta_{n-1}, H(n), A(n)$ e $\hat{a}(n)$
Cálculos	$z(n) = w z(n-1) + \left( H^T(n) A(n) + \hat{a}(n) \right) x^*(n)$
	$\theta_n = \angle z(n)$

Tabela 3.2: Algoritmo RLS para o Ajuste de Fase

### 3.1.3 Preditor IIR

Nesta seção, deduziremos um algoritmo RLS para o preditor IIR correspondente ao equalizador de magnitude da figura 2.6. Nesta figura, vemos que o erro de predição do preditor IIR em questão é dado por  $p(n)$ . Conforme vimos, o preditor deve minimizar a média quadrática do erro de predição. Assim, o critério de mínimos quadrados associado a esta estrutura é dado por:

$$\mathcal{E}_P(n) = \sum_{i=1}^n w^{n-1} |p(i, n)|^2. \quad (3.37)$$

Vimos também que o erro de predição de um preditor IIR pode ser escrito vetorialmente como

$$p(i, n) = x(i) - C^H(n) P(i), \quad (3.38)$$

onde  $x(n)$  é a saída do canal,  $C(n)$  é o vetor de coeficientes do preditor e  $P(i) = [p(i-1), \dots, p(i-N)]^T$ . Assim, a equação 3.37 pode ser reescrita como

$$\mathcal{E}_P(n) = \sum_{i=1}^n w^{n-i} \left[ |x(i)|^2 - x^*(i) C^H(n) P(i) - x(i) P^H(i) C(n) + C^H(n) P(i) P^H(i) C(n) \right]. \quad (3.39)$$

A equação acima, apesar de semelhante à equação 3.14, guarda em relação a esta algumas diferenças cruciais. Na seqüência, discutiremos as implicações destas diferenças, antes de passarmos à dedução do algoritmo RLS propriamente dito.

### Minimização de uma Função de Variáveis Complexas

A primeira diferença a ser notada entre 3.39 e 3.14 é que os termos envolvidos em 3.39 têm valores complexos. A minimização de uma função de variáveis complexas apresenta algumas particularidades, discutidas em [8]. Apresentamos, a seguir, algumas delas:

- Seja  $f(x)$  uma função da variável complexa  $x = x_{\Re} + j x_{\Im}$ . Definimos então a derivada parcial de  $f(x)$  em relação a  $x$  como

$$\frac{\partial f(x)}{\partial x} = \frac{1}{2} \left( \frac{\partial f(x)}{\partial x_{\Re}} - j \frac{\partial f(x)}{\partial x_{\Im}} \right) \quad (3.40)$$

e a derivada parcial de  $f(x)$  em relação ao conjugado de  $x$ ,  $x^*$ , como

$$\frac{\partial f(x)}{\partial x^*} = \frac{1}{2} \left( \frac{\partial f(x)}{\partial x_{\Re}} + j \frac{\partial f(x)}{\partial x_{\Im}} \right) \quad (3.41)$$

- Seja  $f(C)$  uma função de um vetor de valores complexos  $C$ . Se  $C^o$  é um ponto de mínimo de  $f(C)$ , então

$$\nabla_{C^*} f(C^o) = \left. \frac{\partial f(C)}{\partial C^*} \right|_{C=C^o} = 0,$$

onde  $\nabla_{C^*} f(C)$  é o vetor das derivadas de  $f(C)$  em relação ao conjugado dos componentes de  $C$ . Matematicamente,

$$\nabla_{C^*} f(C) = \left[ \frac{\partial f(C)}{\partial c_1^*}, \dots, \frac{\partial f(C)}{\partial c_N^*} \right]^T.$$

- Dado um vetor vetores complexo  $C$ , um vetor constante  $B$  e uma matriz constante  $A$ , valem as seguintes igualdades:

$$\text{a) } \frac{\partial B^H C}{\partial C^*} = 0.$$

$$\text{b) } \frac{\partial C^H B}{\partial C^*} = B.$$

$$\text{c) } \frac{\partial C^H A C}{\partial C^*} = A C.$$

- A regra da derivada do produto também é válida. Assim, dados vetores complexos  $B(x)$  e  $C(x)$ , temos

$$\frac{\partial B^H(x) C(x)}{\partial x^*} = B^H(x) \frac{\partial C(x)}{\partial x^*} + \frac{\partial B^H(x)}{\partial x^*} C(x), \quad (3.42)$$

onde

$$\frac{\partial B(x)}{\partial x^*} = \left[ \frac{\partial b_1(x)}{\partial x^*}, \dots, \frac{\partial b_N(x)}{\partial x^*} \right]^T. \quad (3.43)$$

**Cálculo do Gradiente de  $\mathcal{E}_P(n)$** 

Outra diferença importante entre 3.39 e 3.14 é que, nesta última, podemos usar a equação 3.15 e considerar o vetor  $\Re\{d^*(i) \mathcal{Y}(i)\}$  e a matriz  $Y(i) Y^T(i)$  como valores independentes de  $\mathcal{H}(n)$ . Com isto, o critério RLS para o equalizador de fase se torna uma “forma quadrática” em  $\mathcal{H}(n)$ , cujo mínimo é facilmente calculado. Na equação 3.39 isto não é possível.

Para entender o porquê disto, vamos escrever o gradiente de  $\mathcal{E}_P(n)$ . Lembrando que  $|p(i, n)|^2 = p^*(i, n) p(i, n)$ ,

$$\begin{aligned} \nabla_{C^*} \mathcal{E}_P(n) &= \sum_{i=1}^n w^{n-i} \left[ p^*(i, n) \frac{\partial p(i, n)}{\partial C^*(n)} + p(i, n) \frac{\partial p^*(i, n)}{\partial C^*(n)} \right] \\ &= p^*(n) \frac{\partial p(n)}{\partial C^*(n)} + p(n) \frac{\partial p^*(n)}{\partial C^*(n)} + w \frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n)}. \end{aligned} \quad (3.44)$$

Consideremos, então, os cálculos das derivadas de  $p^*(n)$  e  $p(n)$  em relação ao  $k^{\text{esimo}}$  componente de  $C^*(n)$ ,  $c_k^*(n)$ . Assim, lembrando que podemos escrever  $p(n) = x(n) - \sum_{i=1}^N c_i^*(n) p(n-i)$ , temos

$$\begin{aligned} \frac{\partial p(n)}{\partial c_k^*(n)} &= - \sum_{i=1}^N p(n-i) \frac{\partial c_i^*(n)}{\partial c_k^*(n)} - \sum_{i=1}^N c_i^*(n) \frac{\partial p(n-i)}{\partial c_k^*(n)} \\ &= -p(n-k) - \sum_{i=1}^N c_i^*(n) \frac{\partial p(n-i)}{\partial c_k^*(n)} \end{aligned} \quad (3.45)$$

$$\begin{aligned} \frac{\partial p^*(n)}{\partial c_k^*(n)} &= - \sum_{i=1}^N p^*(n-i) \frac{\partial c_i(n)}{\partial c_k^*(n)} - \sum_{i=1}^N c_i(n) \frac{\partial p^*(n-i)}{\partial c_k^*(n)} \\ &= - \sum_{i=1}^N c_i^*(n) \frac{\partial p^*(n-i)}{\partial c_k^*(n)}. \end{aligned} \quad (3.46)$$

Observe que o segundo termo da equação 3.45 envolve o cálculo da derivada de valores passados de  $p(n)$  em relação a valores presentes dos coeficientes do filtro. Nos cálculos do gradiente do critério de mínimos quadrados para um filtro FIR este valor é nulo, pois ele corresponde à derivada da saída do canal em relação aos coeficientes do filtro, e estas grandezas são independentes. Os filtros IIR, por outro lado, são recursivos. Por isso, os valores de suas “entradas”, correspondentes aos valores passados de suas saídas, foram calculados a partir de valores passados de seus coeficientes. O cálculo de  $p(n-i)$ , por exemplo, envolveu o vetor  $C(n-i)$ . Por outro lado,  $C(n)$  também é obtido a partir de valores passados do vetor de coeficientes. Isto implica em uma dependência entre os valores passados de  $p(n)$  e o valor presente do vetor de coeficientes e, portanto, em uma necessidade de se avaliarem as derivadas parciais das equações 3.45 e 3.46.

É interessante notar que esta dependência torna o critério de mínimos quadrados algo sem muito sentido no contexto de preditores IIR. Isto porque este critério busca, a cada instante de tempo  $n$ , minimizar a média temporal dos erros de predição, como se estes erros tivessem sido obtidos, desde o instante inicial, com o uso do vetor de coeficientes  $C(n)$ . Ora, o valor de  $p(i)$ , observado em um dado instante  $i$ , foi obtido com o uso do vetor de coeficientes  $C(i)$ . Se  $C(n)$  tivesse sido usado desde o instante inicial, um valor diferente de

$p(i)$  teria sido obtido. O vetor  $P(i+1)$ , que usa o valor de  $p(i)$ , seria, portanto, diferente e o cálculo exato de  $p(i+1, n)$  envolveria a reavaliação deste vetor. Assim, um critério de mínimos quadrados exato não poderia usar os valores observados de  $p(i)$  e deveria, a cada instante  $n$ , recalculá-los desde o instante inicial, supondo o uso do vetor de coeficientes  $C(n)$ . Isto é inviável. Entretanto, veremos que é possível obter um algoritmo dito RLS com características semelhantes ao RLS para filtros FIR.

Esta dependência também torna a avaliação das derivadas da equação 3.45 muito difícil, e para seu cálculo exato também precisaríamos recalculá-los os valores passados do erro de predição. Entretanto, supondo que a adaptação dos coeficientes do filtro é lenta, podemos considerar  $C(n) \approx C(n-1) \approx \dots \approx C(n-N)$  (cf. [18]), de forma a reescrever a equação 3.45 como

$$\frac{\partial p(n)}{\partial c_k^*(n)} = -p(n-k) - \sum_{i=1}^N c_i^*(n) \frac{\partial p(n-i)}{\partial c_k^*(n-i)}. \quad (3.47)$$

Se definirmos

$$\bar{\phi}_k(n) = -\frac{\partial p(n)}{\partial c_k^*(n)}, \quad k = 1, \dots, N, \quad (3.48)$$

poderemos escrever

$$\bar{\phi}_k(n) = p(n-k) + \sum_{i=1}^N c_i^*(n) \bar{\phi}_k(n-i). \quad (3.49)$$

Assim, a derivada parcial de  $p(n)$  em relação a  $c_k^*(n)$  pode ser vista como a saída de um filtro IIR com vetor de coeficientes  $C(n)$  e com entrada  $-p(n-k)$ . Finalmente,

$$\nabla_{C^*} \mathcal{E}_P(n) = -p^*(n) \bar{\Phi}(n) + w \frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n)}, \quad (3.50)$$

onde  $\bar{\Phi}(n) = [\bar{\phi}_1(n), \dots, \bar{\phi}_N(n)]^T$ .

Vamos acompanhar os cálculos de  $\bar{\phi}_k(n)$  a partir de um sinal  $a = -p(m)$ , onde  $m$  é um instante de tempo qualquer. No instante  $m+1$ , este sinal se encontra na entrada de um filtro IIR com vetor de coeficientes  $C(m+1)$ , cuja saída é  $\bar{\phi}_1(m+1)$ . Este sinal se propaga pela entrada de sucessivos filtros até que, no instante  $m+N$ , ele se encontra na entrada de um filtro IIR com vetor de coeficientes  $C(m+N)$ , cuja saída é  $\bar{\phi}_N(m+N)$ . Observe, então, que este sinal passa por  $N$  filtros distintos para produzir cada um dos  $\bar{\phi}_k(n)$  e, portanto, o cálculo dos  $N$  valores de  $\bar{\phi}_k(n)$  envolve a implementação de  $N$  filtros IIR distintos. Esta implementação, entretanto, é computacionalmente cara. Se considerarmos, novamente, que a adaptação dos coeficientes do filtro é lenta, e, portanto, que  $C(n) \approx C(n+i)$ , para  $i = 1, \dots, N$ , vemos que o sinal  $a = -p(m)$  passa, em sucessivos instantes de tempo, por filtros com vetores de coeficientes praticamente iguais. No instante  $m+1$ , por exemplo, este sinal se encontra na entrada de um filtro IIR com vetor de coeficientes  $C(m+1)$ , cuja saída é  $\bar{\phi}_1(m+1)$ . No instante  $m+2$ , ele se encontra na entrada de um filtro IIR com vetor de coeficientes  $C(m+2)$ , cuja saída é  $\bar{\phi}_2(m+2)$ . Mas  $C(m+2) \approx C(m+1)$  e, portanto  $\bar{\phi}_2(m+2) \approx \bar{\phi}_1(m+1)$ . Assim, podemos considerar que  $\bar{\phi}_k(n) \approx \bar{\phi}_1(n-k+1)$ . Se definirmos

$$\phi(n) = p(n) + \sum_{i=1}^L c_i^*(n) \phi(n-i), \quad (3.51)$$

teremos  $\bar{\phi}_k(n) \approx \phi(n-k)$ , o que permite o cálculo aproximado de  $\bar{\Phi}(n)$  com a implementação de apenas um filtro IIR. Desta forma, definindo  $\Phi(n) = [\phi(n-1), \dots, \phi(n-N)]^T$ , temos:

$$\nabla_{C^*} \mathcal{E}_P(n) = -p^*(n) \Phi(n) + w \frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n)}. \quad (3.52)$$

Finalmente, podemos supor outra vez que a adaptação dos coeficientes é lenta. Assim,

$$\frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n)} \approx \frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n-1)}. \quad (3.53)$$

Ora, sabemos que, a cada iteração, o algoritmo RLS busca minimizar a função custo. Assim, o gradiente de  $\mathcal{E}_P(n)$  pode ser considerado nulo após cada iteração, de forma que

$$\frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n)} \approx \frac{\partial \mathcal{E}_P(n-1)}{\partial C^*(n-1)} \approx 0. \quad (3.54)$$

A mesma conclusão pode ser obtida se considerarmos os termos de primeira ordem da série de Taylor de  $\mathcal{E}_P(n-1)$  ao redor do ponto  $C(n-1)$  (cf. [19]).

Para avaliar as derivadas presentes na equação 3.46, podemos também considerar  $C(n) \approx C(n-1) \approx \dots \approx C(n-N)$ . Assim, podemos reescrever 3.46 como:

$$\frac{\partial p^*(n)}{\partial c_k^*(n)} = - \sum_{i=1}^N c_i^*(n) \frac{\partial p^*(n-i)}{\partial c_k^*(n-i)}. \quad (3.55)$$

Seguindo o mesmo raciocínio desenvolvido anteriormente, vemos que esta equação corresponde à filtragem do sinal  $\frac{\partial p^*(n)}{\partial c_k^*(n)}$  por um filtro IIR. Neste caso, entretanto, esta filtragem não possui nenhum termo forçante, como o termo  $-p(n-k)$  em 3.47. Assim, como o filtro IIR é estável, o valor de  $\frac{\partial p^*(n)}{\partial c_k^*(n)}$  tende assintoticamente a zero, e pode ser desprezado.

### O Algoritmo RLS

Conforme vimos, o critério RLS não faz muito sentido no contexto de preditores IIR. Consideremos, entretanto, o algoritmo RLS para filtros FIR. Para minimizar a cada iteração a critério de mínimos quadrados, este algoritmo de certa forma estima recursivamente o gradiente e a Hessiana da função custo. Assim, podemos pensar em um algoritmo RLS para filtros IIR em que a adaptação do vetor de coeficientes se dá de acordo com

$$C(n+1) = C(n) - R^{-1}(n) \nabla_{C^*} \mathcal{E}_P(n), \quad (3.56)$$

onde  $R(n)$  será tomada como a matriz das derivadas segundas de  $\mathcal{E}_P(n)$ , dada por

$$R(n) = \frac{\partial \nabla_{C^*} \mathcal{E}_P(n)^T}{\partial C}. \quad (3.57)$$

A partir de 3.52, é fácil ver que

$$R(n) = w R(n-1) - \Phi(n) \Phi^H(n) - p^*(n) \frac{\partial \Phi(n)^T}{\partial C}. \quad (3.58)$$

O último termo da equação acima envolve o produto de  $p^*(n)$  pelas derivadas segundas de  $P(n)$ . Estes valores tendem a ser descorrelacionados, pois a saída do preditor tende a ser um ruído branco, de forma que  $p(n)$  não depende de seus valores passados, contidos no vetor  $P(n)$ , nem de suas derivadas. Assim, ainda que o erro de predição não possa ser considerado branco no início das iterações, este termo será desprezado.

Finalmente, considerando as aproximações discutidas, o algoritmo RLS para o preditor IIR fica:

$$R(n) = w R(n-1) - \Phi(n) \Phi^H(n) \quad (3.59)$$

$$C(n+1) = C(n) - R^{-1}(n) \Phi(n) p^*(n) \quad (3.60)$$

Observe que, nesta forma, o algoritmo RLS para o preditor FIR é muito semelhante ao algoritmo RLS para o filtro FIR. Aqui,  $p(n)$  faz o papel do sinal de erro  $e(n)$  e  $\Phi(n)$  faz o papel de  $X(n)$ . Podemos também fazer uso do lema de inversão de matrizes, obtendo o algoritmo resumido na tabela 3.3. De maneira semelhante ao filtro RLS para o equalizador de fase, foi introduzida uma matriz auxiliar  $U(n)$  que permitem reduzir o esforço computacional do algoritmo.

Disponível no instante  $n$

$$R^{-1}(n), C(n), P(n), \Phi(n) \text{ e } x(n)$$

Cálculos

$$p(n) = x(n) - C^H(n) P(n)$$

$$\phi(n) = p(n) + C^H(n) \Phi(n)$$

$$U(n) = R(n) \Phi(n)$$

$$k(n) = (w + \Phi^H(n) U(n))^{-1}$$

$$R^{-1}(n+1) = \frac{1}{w} \left( R^{-1}(n) - \frac{U(n) U^H(n)}{k(n)} \right)$$

$$C(n+1) = C(n) - R^{-1}(n) \Phi(n) p^*(n)$$

Tabela 3.3: Algoritmo RLS para o Preditor IIR

## 3.2 Algoritmos de Gauss-Newton

Conforme vimos no capítulo anterior, o algoritmo de Gauss-Newton é uma aproximação estocástica para o método de Newton de otimização. Este método busca minimizar a função custo  $J(H)$ , adaptando o vetor  $H$  segundo a equação:

$$H(n+1) = H(n) - \mu S^{-1}(n) \nabla_H J, \quad (3.61)$$

onde  $S(n)$  é a Hessiana de  $J(H)$ .

Nesta seção, desenvolveremos algoritmos adaptativos para os equalizadores de magnitude e fase baseados no método de Newton. Para isto, deveremos estabelecer aproximações estocásticas para o gradiente e a Hessiana da função custo associada a cada uma destas estruturas, o que será feito na seqüência.

### 3.2.1 Equalizador de Fase

A função custo a ser minimizada, no caso do equalizador de fase, é dada por:

$$J_{DD}(H) = \text{E} \left( |\varepsilon(n)|^2 \right). \quad (3.62)$$

Baseado na equação 3.14 e lembrando que a esperança é um operador linear, podemos reescrever esta função custo como:

$$J_{DD}(H) = \text{E} \left( |d(n)|^2 \right) - 2 \text{E} \left( \Re \{ d^*(n) \mathcal{Y}(n) \} \right) \mathcal{H} + \mathcal{H}^T \text{E} \left( Y(n) Y^T(n) \right) \mathcal{H}. \quad (3.63)$$

Nesta equação, fizemos mais uma vez uso da hipótese de independência entre os símbolos decididos e o vetor de coeficientes  $\mathcal{H}$  (eq. 3.15).

Notamos, assim, que, de maneira semelhante ao critério de mínimos quadrados, a função custo corresponde a uma “forma quadrática” em  $\mathcal{H}$ . Temos, então, que:

$$S(n) = \text{E} \left( Y(n) Y^T(n) \right) \quad (3.64)$$

$$\nabla_H J = -2 \text{E} \left( \Re \{ d^*(n) \mathcal{Y}(n) \} \right) + 2 \text{E} \left( Y(n) Y^T(n) \right) \mathcal{H}. \quad (3.65)$$

Conforme vimos no caso de um filtro FIR, as aproximações estocásticas para estas grandezas são dadas por:

$$\hat{S}(n) = \alpha Y(n) Y^T(n) + \alpha \sum_{i=1}^{n-1} (1 - \alpha)^{n-i} Y(i) Y^T(i). \quad (3.66)$$

$$\hat{\nabla}_H J = -2 \Re \{ d^*(n) \mathcal{Y}(n) \} + 2 Y(n) Y^T(n) \mathcal{H}. \quad (3.67)$$

Com o objetivo de escrever a estimativa do gradiente em uma forma que facilite seu uso, podemos fazer as seguintes manipulações, que se baseiam nas matrizes definidas na seção 3.1:

$$\begin{aligned} \hat{\nabla}_H J &= -2 d_{\Re}(n) \mathcal{Y}_{\Re}(n) - 2 d_{\Im}(n) \mathcal{Y}_{\Im}(n) + 2 Y(n) Y^T(n) \mathcal{H} \\ &= -2 Y(n) \begin{bmatrix} d_{\Re}(n) \\ d_{\Im}(n) \end{bmatrix} + 2 Y(n) Y^T(n) \mathcal{H} \\ &= 2 Y(n) \left( \begin{bmatrix} -d_{\Re}(n) \\ -d_{\Im}(n) \end{bmatrix} + Y^T(n) \mathcal{H} \right) \\ &= 2 Y(n) \left( \begin{bmatrix} -d_{\Re}(n) \\ -d_{\Im}(n) \end{bmatrix} + \begin{bmatrix} \mathcal{Y}_{\Re}(n) \mathcal{H} \\ \mathcal{Y}_{\Im}(n) \mathcal{H} \end{bmatrix} \right). \end{aligned} \quad (3.68)$$

Finalmente, notando que  $\mathcal{Y}_{\Re}(n) \mathcal{H} - d_{\Re}(n)$  e  $\mathcal{Y}_{\Im}(n) \mathcal{H} - d_{\Im}(n)$  correspondem, respectivamente, às partes reais e imaginárias do erro de decisão direta (ver equação 3.8), chegamos a:

$$\hat{\nabla}_H J = 2 Y(n) \begin{bmatrix} \varepsilon_{\Re}(n) \\ \varepsilon_{\Im}(n) \end{bmatrix}. \quad (3.69)$$

Aplicando o lema de inversão de matrizes à estimativa da Hessiana, chegamos ao algoritmo de GN para o equalizador de fase. Este algoritmo está apresentado na tabela 3.4. É interessante notar que as semelhanças entre os algoritmos GN e RLS para o equalizador de fase são as mesmas observadas no caso de um filtro FIR.

$$\begin{array}{l}
\text{Disponível no instante } n \\
P(n-1), \mathcal{H}(n-1), Y(n), \mathcal{Y}(n) \text{ e } d(n) \\
\text{Cálculos} \\
\varepsilon(n) = \mathcal{H}^T(n-1) \mathcal{Y}(n) - d(n) \\
U(n) = P(n-1) Y(n) \\
k(n) = (w I_2 + \alpha Y^T(n) U(n))^{-1} \\
P(n) = \frac{1}{w} (P(n-1) - \alpha U(n) k(n) U^T(n)) \\
\mathcal{H}(n) = \mathcal{H}(n-1) - \alpha P(n) Y(n) \begin{bmatrix} \varepsilon_{\Re}(n) \\ \varepsilon_{\Im}(n) \end{bmatrix}
\end{array}$$

Tabela 3.4: Algoritmo de Gauss-Newton para o Filtro Passa-Tudo

### 3.2.2 Equalizador de Magnitude

Para o equalizador de magnitude, a função custo a ser minimizada é dada por:

$$J_P(C) = E \left( |p(n)|^2 \right). \quad (3.70)$$

Devemos, agora, determinar o gradiente e a Hessiana de  $J_P(C)$ .

Para calcular o gradiente, notamos inicialmente que  $|p(n)|^2 = p(n) p^*(n)$  e, portanto,

$$\nabla J_P(n) = E \left( p^*(n) \frac{\partial p(n)}{\partial C^*(n)} + p(n) \frac{\partial p^*(n)}{\partial C^*(n)} \right) = E \left( p^*(n) \frac{\partial p(n)}{\partial C^*(n)} \right). \quad (3.71)$$

Nesta última passagem, usamos um resultado obtido na dedução do algoritmo RLS para o preditor, a saber:

$$\frac{\partial p^*(n)}{\partial C^*(n)} = 0. \quad (3.72)$$

Finalmente, usando uma estimativa instantânea da esperança e lembrando que

$$\frac{\partial p(n)}{\partial C^*(n)} \approx \Phi(n), \quad (3.73)$$

chegamos à estimativa do gradiente, dada por:

$$\hat{\nabla} J_P(n) = p^*(n) \Phi(n). \quad (3.74)$$

A Hessiana de  $J_P(n)$  é a matriz de suas derivadas segundas, dada por

$$S(n) = \frac{\partial (\nabla J_P(n))^T}{\partial C(n)} = E \left[ \frac{\partial p(n)}{\partial C^*(n)} \left( \frac{\partial p^*(n)}{\partial C(n)} \right)^T + p(n) \frac{\partial^2 p^*(n)}{\partial C(n) \partial C^*(n)} \right]. \quad (3.75)$$

Assim como fizemos na dedução do algoritmo RLS para o preditor IIR, consideraremos que o erro de predição  $p(n)$  é um ruído branco de média nula e pode, portanto, ser considerado independente de seus valores passados e, conseqüentemente, de suas derivadas. Desta forma, podemos considerar nula a esperança do segundo termo da equação acima e, portanto,

$$S(n) = E \left[ \frac{\partial p(n)}{\partial C^*(n)} \left( \frac{\partial p(n)}{\partial C^*(n)} \right)^H \right] \approx E \left[ \Phi(n) \Phi^H(n) \right]. \quad (3.76)$$

A aproximação estocástica da matriz Hessiana, fica, assim,

$$\hat{S}(n) = \alpha \Phi(n) \Phi^H(n) + \alpha \sum_{i=1}^{n-1} (1 - \alpha)^{n-i} \Phi(i) \Phi^H(i). \quad (3.77)$$

Finalmente, aplicando o lema de inversão de matrizes a esta última recursão, chegamos ao algoritmo de GN para o equalizador de magnitude apresentado na tabela 3.5.

Disponível no instante $n$	$R(n), C(n), P(n), \Phi(n)$ e $x(n)$
Cálculos	
	$p(n) = x(n) - C^H(n) P(n)$
	$\phi(n) = p(n) + C^H(n) \Phi(n)$
	$U(n) = R(n) \Phi(n)$
	$k(n) = (w + \alpha \Phi^H(n) U(n))^{-1}$
	$R(n+1) = \frac{1}{w} \left( R(n) - \alpha \frac{U(n) U^H(n)}{k(n)} \right)$
	$C(n+1) = C(n) - \alpha R(n) \Phi(n) p^*(n)$

Tabela 3.5: Algoritmo de Gauss-Newton para o Preditor IIR

### 3.3 Conclusão

Neste capítulo, apresentamos novos algoritmos para as técnicas preditivas de equalização cega. A estrutura usada para a dedução destes algoritmos foi aquela baseada na equalização de magnitude e fase pois, como está mostrado em [3], esta estrutura se mostra mais eficiente do que a baseada em predição e retropredição.

Os algoritmos apresentados aqui foram obtidos a partir da aplicação dos métodos RLS e GN na minimização das funções custo associadas a esta estrutura. Estas funções custo são baseadas no erro de predição, no caso do equalizador de magnitude, e no erro de decisão direta, no caso do equalizador de fase.

A dedução dos algoritmos para o equalizador de fase parte do princípio de que a saída do dispositivo de decisão não depende dos coeficientes do equalizador. A partir daí, podemos escrever o erro de decisão direta de uma maneira que permite a dedução de um algoritmo RLS seguindo uma linha de raciocínio muito semelhante àquela da dedução do RLS para filtros FIR. Esta forma de escrever o erro de decisão direta também permite uma dedução quase imediata do algoritmo GN para o equalizador de fase.

Já a dedução dos algoritmos para o preditor IIR é mais complexa. O fato de sua saída ser realimentada dificulta muito o cálculo das derivadas do erro de predição, e algumas aproximações devem ser feitas para permitir a obtenção de um algoritmo com algum interesse prático. Estas aproximações, entretanto, são interessantes, pois revelam alguns aspectos do funcionamento de sistemas realimentados.

A obtenção dos algoritmos apresentados tanto aqui quanto em [3] envolve várias aproximações, que podem comprometer o comportamento dos algoritmos. Assim, é necessária uma análise que valide estas aproximações, verificando que, mesmo com elas, os algoritmos são capazes de cumprir sua tarefa de minimizar uma função custo. Esta análise será feita no próximo capítulo.

## Capítulo 4

# Análise de Convergência

NESTE capítulo, faremos uma análise teórica do equalizador de fase não-linear e de dois de seus algoritmos de adaptação, LMS e GN. O objetivo desta análise é justificar teoricamente algumas características apresentadas pela estrutura em questão e pelos algoritmos.

Com esta finalidade, na seção 4.1 revisaremos alguns importantes resultados teóricos obtidos em [3]. Entre outros, neste trabalho fica demonstrado que a anulação do erro de decisão direta  $\varepsilon(n)$  implica na correta equalização do canal passa-tudo  $\mathcal{FOP}$  formado pela cascata do canal propriamente dito com o equalizador de magnitude. Mais ainda, prova-se que a decorrelação do sinal na entrada do dispositivo de decisão,  $y(n)$ , leva à anulação do erro de decisão direta e, portanto, à equalização. Finalmente, usaremos este fato para mostrar que, sob certas condições muito pouco restritivas, o critério  $J_\varepsilon(H) = E(|\varepsilon(n)|^2)$  é unimodal.

Baseado nestes resultados, na seção 4.2 usaremos a técnica de ODE para mostrar que os algoritmos propostos de fato convergem para um ponto de mínimo de  $J_\varepsilon(H)$ , levando assim à equalização do canal. A técnica de ODE foi proposta em [19] e permite associar uma ODE (do inglês *Ordinary Differential Equation*, equação diferencial ordinária) ao algoritmo em estudo. As propriedades das soluções desta ODE podem ser, então, associadas às propriedades de convergência do algoritmo. Por exemplo, o algoritmo pode convergir para um determinado ponto se e somente se este for um ponto assintoticamente estável da equação diferencial.

### 4.1 Análise da Função Custo $J_\varepsilon(H)$

Nesta seção, revisaremos algumas propriedades do equalizador de fase não-linear e dos critérios de minimização associados a esta estrutura. Estas propriedades foram estudadas em [3]. Inicialmente, mostraremos que o critério escolhido,  $J_\varepsilon(H)$ , de fato leva à equalização do canal. Em seguida, trataremos do problema da unimodalidade deste critério.

#### 4.1.1 Justificativa para a Escolha de $J_\varepsilon(H)$

Para justificar a utilização de  $J_\varepsilon(H)$  como critério de adaptação dos coeficientes do equalizador de fase, devemos mostrar que sua minimização de fato leva à equalização do canal. Para isso, começamos notando, a partir da figura 2.6, que a entrada do dispositivo de decisão

$y(n)$  pode ser escrita como:

$$y(n) = e^{j\theta} \frac{q^{-L} + \sum_{i=1}^L h_i^* q^{-L+i}}{1 + \sum_{i=1}^L h_i q^{-i}} s(n) + \frac{\sum_{i=1}^L h_i q^{-i}}{1 + \sum_{i=1}^L h_i q^{-i}} \varepsilon(n). \quad (4.1)$$

Para chegar a esta conclusão, basta reescrever  $\hat{a}(n)$  de acordo com sua definição, ou seja,  $\hat{a}(n) = \varepsilon(n) - y(n)$ . A equação acima corresponde à soma da filtragem de  $s(n)$  por um filtro passa-tudo, cuja função de transferência será denotada  $H(q)$ , com a filtragem de  $\varepsilon(n)$  por um filtro cuja função de transferência será denotada  $H^0(q)$ . Assim, o equalizador de fase não-linear pode ser representado alternativamente como na figura 4.1

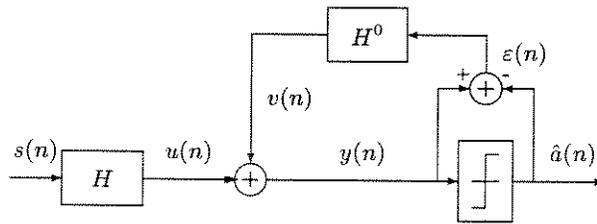


Figura 4.1: Representação Alternativa do Equalizador de Fase.

A partir desta figura, podemos notar que, se o erro DD se anula, o equalizador de fase não-linear se torna um filtro passa-tudo linear seguido pelo dispositivo de decisão. Ora, nesta situação,  $y(n)$  é o resultado da filtragem do sinal transmitido  $a(n)$  por uma cascata de dois filtros passa-tudo: o sistema canal + equalizador de magnitude e o filtro linear  $H$  da figura 4.1. Como esta cascata corresponde também a um filtro passa-tudo, podemos escrever:

$$y(n) = e^{j\varphi} \frac{q^{-M} + \sum_{i=1}^M b_i^* q^{-M+i}}{1 + \sum_{i=1}^M b_i q^{-i}} a(n). \quad (4.2)$$

Outra consequência importante da anulação de  $\varepsilon(n)$  é que isto significa que  $y(n)$  é igual à saída do dispositivo de decisão. Em outras palavras,  $y(n)$  pertence à constelação  $\mathcal{A}$  de símbolos transmitidos. Conforme demonstraremos no teorema a seguir, o fato de  $y(n)$  ser o resultado de uma filtragem linear de  $a(n)$  e ambos pertencerem à mesma constelação implica na equalização do canal.

**Teorema 4.1** *Considere um sistema linear com resposta ao impulso  $s_i = [s_0; s_1; \dots]$  excitado por uma seqüência de símbolos  $\{a(n)\}$  pertencentes a uma certa constelação  $\mathcal{A}$ , onde  $\mathcal{A}$  é um conjunto de valores complexos e  $\Re(\mathcal{A}), \Im(\mathcal{A}) \in \{1-M, 3-M, \dots, M-3, M-1\}$ ,  $M$  par. Se a saída deste sistema também pertence a  $\mathcal{A}$ , então  $s_i = e^{j\theta} \delta_{i-m}$ .*

**Prova** Seja  $y(n)$  a saída do sistema linear em questão. Podemos escrever:

$$y(n) = \sum_i s_i a(n-i). \quad (4.3)$$

Sejam  $k$  e  $K$  respectivamente o menor e o maior valor do módulo dos símbolos de  $\mathcal{A}$ . Então  $k \leq |y(n)| \leq K$ , ou

$$k \leq \left| \sum_i s_i a(n-i) \right| \leq K. \quad (4.4)$$

Considere agora a transmissão de uma seqüência de símbolos  $a(n) = \pm k e^{j\phi_k}$ , onde  $\phi_k$  é a fase de um dos símbolos de módulo  $k$ . Temos então:

$$1 \leq \left| \sum_i \pm s_i \right| \leq \frac{K}{k}. \quad (4.5)$$

De maneira semelhante, se  $a(n) = \pm K e^{j\phi_K}$ , onde  $\phi_K$  é a fase de um dos símbolos de módulo  $K$  temos:

$$\frac{k}{K} \leq \left| \sum_i \pm s_i \right| \leq 1. \quad (4.6)$$

Estas duas desigualdades nos permitem concluir que

$$\left| \sum_i \pm s_i \right| = 1. \quad (4.7)$$

A demonstração passa, agora, por duas etapas. Inicialmente, vamos demonstrar que 4.7 implica que apenas dois símbolos da constelação são não nulos. Em seguida, mostraremos que esta situação é incompatível com a outra hipótese do teorema, de que  $y(n) \in \mathcal{A}$ .

Para demonstrar que apenas dois símbolos da constelação são não nulos, vamos trabalhar com os complexos  $s_i$  como se estes fossem vetores em  $\mathbb{R}^2$ . Consideremos que  $s_n$  e  $s_m$  são dois elementos não nulos da seqüência  $s_i$ . Consideremos também

$$u(p, q) \triangleq \sum_{\substack{i \neq p \\ i \neq q}} \pm s_i. \quad (4.8)$$

Então, 4.7 nos permite escrever, para um índice  $j$  qualquer,

$$\begin{aligned} |s_n + s_j + u(n, j)|^2 &= 1 \\ |s_n - s_j - u(n, j)|^2 &= 1. \end{aligned}$$

Lembrando que, para um vetor  $x$ ,  $|x|^2 = x^T x$ , concluímos que:

$$\begin{aligned} s_n^T s_n + (s_j + u(n, j))^T (s_j + u(n, j))^T + 2 s_n^T (s_j + u(n, j)) &= 1 \\ s_n^T s_n + (s_j + u(n, j))^T (s_j + u(n, j))^T - 2 s_n^T (s_j + u(n, j)) &= 1. \end{aligned}$$

Subtraindo estas duas equações, chegamos finalmente a:

$$s_n^T s_j + s_n^T u(n, j) = 0. \quad (4.9)$$

De maneira análoga, se, a partir de 4.7, escrevermos:

$$\begin{aligned} |s_n - s_j + u(n, j)|^2 &= 1 \\ |s_n + s_j - u(n, j)|^2 &= 1, \end{aligned}$$

concluiremos que

$$s_n^T s_j - s_n^T u(n, j) = 0. \quad (4.10)$$

As equações 4.9 e 4.10 formam um sistema de duas equações cuja solução é

$$s_n^T s_j = s_n^T u(n, j) = 0. \quad (4.11)$$

Em outras palavras,  $s_n$  é ortogonal a  $s_j$ , para qualquer  $j$ . Em particular,  $s_n$  é ortogonal a  $s_m$ . Se repetirmos todo o raciocínio anterior tomando  $s_m$  no lugar de  $s_n$ , concluiremos que  $s_m$  também é ortogonal a todo  $s_j$ . Ora, se  $s_j$  é um vetor de  $\mathbb{R}^2$  ortogonal a  $s_n$  e  $s_m$ , sendo que estes são mutuamente ortogonais e não nulos, então  $s_j = 0$ . Fica assim demonstrado que a seqüência  $s_i$  possui apenas dois elementos não nulos, e que estes são ortogonais. Além disso, sua soma tem módulo 1.

Entretanto, é fácil ver que não é possível termos  $s_n$  e  $s_m$  satisfazendo as condições acima e ainda assim termos que a saída do canal pertence à constelação  $\mathcal{A}$ . Desta forma, devemos necessariamente ter que um dos dois elementos é nulo. Assim, se, por exemplo  $s_n = 0$ , devemos ter  $|s_m| = 1$ , ou  $s_m = e^{j\theta}$ . Assim,  $s_i = e^{j\theta} \delta_{i-m}$ . ■

Podemos agora demonstrar o seguinte resultado:

**Teorema 4.2** *Se  $E(y(n)y(n-i)) = 0$ ,  $i = 1, 2, \dots, N$ , então o canal está corretamente equalizado.*

**Prova** Seja

$$v(n) = \frac{\sum_{i=1}^N h_i q^{-i}}{1 + \sum_{i=1}^N h_i q^{-i}} \varepsilon(n) \quad (4.12)$$

o componente não-linear do equalizador de fase. Ora  $y(n)$  é a soma de um sinal branco,  $s(n)$ , com  $v(n)$ . Desta forma, para que  $y(n)$  seja branco, devemos ter que  $v(n)$  é também branco ou que  $v(n)$  é nulo.

Por outro lado, o erro DD pode ser visto como um ruído de quantização, que é o erro que se comete ao se quantizar um sinal, ou seja, ao passá-lo por um dispositivo de decisão. Este ruído pode ser modelado como um sinal branco, conforme evidenciado na figura 4.2, que mostra uma estimativa da função de autocorrelação do ruído de quantização, obtida a partir da quantização da saída do canal de microondas de [20]. Nesta figura vemos que, de fato, o erro DD pode ser considerado branco. Ora, como  $v(n)$  é o resultado de uma filtragem de  $\varepsilon(n)$ , para que  $v(n)$  também seja branco é necessário que esta filtragem seja executada por um filtro passa-tudo. Entretanto, de 4.12, vemos claramente que este não é o caso e, portanto,  $v(n)$  não pode ser branco. Assim,  $v(n)$  deve ser nulo, o que implica que  $\varepsilon(n)$  é nulo. Temos então que  $E(y(n)y(n-i)) = 0$  para  $i = 1, 2, \dots, N$  se e somente se  $\varepsilon(n) = 0$ . Por outro lado, se isso é verdade, então, de acordo com o teorema 4.1 o canal está equalizado. ■

Obviamente, o erro DD nunca será exatamente nulo. Entretanto, o objetivo do equalizador não é a inversão exata do canal, mas uma inversão boa o suficiente para permitir a correta identificação dos símbolos transmitidos pelo dispositivo de decisão. Os resultados apresentados aqui nos levam a crer que se o erro DD for pequeno o canal estará “suficientemente” equalizado. Justifica-se, desta maneira, a escolha da minimização do erro DD como critério para adaptação dos coeficientes do equalizador de fase. Mais ainda, cabe ressaltar a importância do teorema 4.2. O fato de que a decorrelação de  $y(n)$  implica na correta equalização do canal será usado na seqüência para demonstrar a unimodalidade do critério  $J_\varepsilon(H)$ .

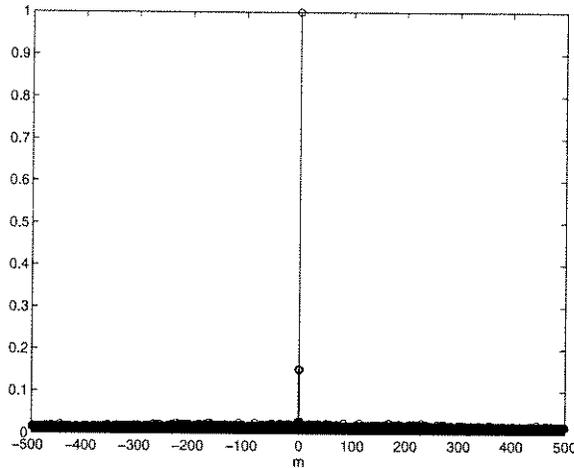


Figura 4.2: Estimativa da Função de Autocorrelação de  $\{\varepsilon(n)\}$ .

#### 4.1.2 Unimodalidade de $J_\varepsilon(H)$

Para constatar a unimodalidade de  $J_\varepsilon(H)$ , mostraremos que sempre que este critério é minimizado o canal está equalizado. Assim, considere um ponto de mínimo de  $J_\varepsilon(H)$ . A partir da equação 2.63, vemos que quando o gradiente de  $J_\varepsilon(H)$  se anula, temos

$$E(\varepsilon(n) \hat{a}^*(n-i)) = E(\varepsilon(n) s(n-L+i)), \quad i = 1, \dots, N. \quad (4.13)$$

Ora, em [1] vemos que os sinais  $\{s(n)\}$  e  $\{\varepsilon(n)\}$  podem ser considerados decorrelacionados quando  $J_\varepsilon(H)$  está minimizado. Esta hipótese pode ser confirmada na figura 4.3, onde mostramos a estimativa da correlação cruzada entre  $\{s(n)\}$  e  $\{\varepsilon(n)\}$  obtida de uma simulação do sistema onde a condição de gradiente nulo para  $J_\varepsilon(H)$  era satisfeita.

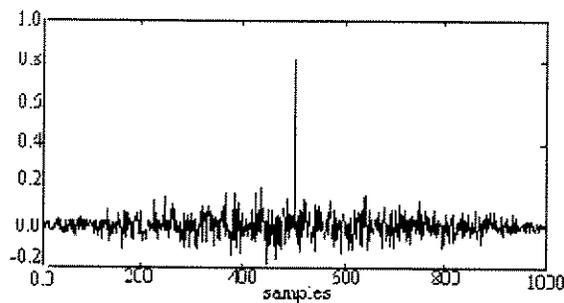


Figura 4.3: Estimativa da Correlação Cruzada Normalizada entre  $\{s(n)\}$  e  $\{\varepsilon(n)\}$  (Ref. [1]).

Com base nesta hipótese, concluímos, a partir de 4.13 que

$$E(\varepsilon(n) \hat{a}^*(n-i)) = 0, \quad i = 1, 2, \dots, N. \quad (4.14)$$

Como  $\varepsilon(n) = y(n) - \hat{a}(n)$ , podemos reescrever 4.14 como

$$E(y(n) \hat{a}(n-i)) = E(\hat{a}(n) \hat{a}(n-i)), \quad i = 1, 2, \dots, N. \quad (4.15)$$

Por outro lado, o teorema de Bussgang [7] diz que se a entrada de um dispositivo sem memória é um processo estacionário, normal e de média nula, então a correlação cruzada entre a saída e a entrada do dispositivo é proporcional à autocorrelação de sua entrada. Ora, se a adaptação dos coeficientes do equalizador de fase for suficientemente lenta, podemos considerar  $y(n)$  estacionário.

Quanto à normalidade de  $y(n)$ , em [3] está mostrado que se:

- a) O olho do canal está fechado.
- b) A resposta ao impulso do canal é suficientemente longa (maior que 4 amostras).
- c) A constelação transmitida é “rica” o suficiente, ou seja, possui uma quantidade suficientemente grande de símbolos (3 ou mais).

então, invocando o Teorema Central do Limite, podemos concluir que o sinal  $y(n)$  segue uma distribuição gaussiana. Para confirmar esta hipótese, apresentamos na figura 4.4 uma estimativa da densidade de probabilidade (histograma) e da distribuição de probabilidade de  $y(n)$ , comparando-as com as funções da distribuição gaussiana. Estas estimativas foram obtidas a partir da simulação da transmissão de uma seqüência 4-PAM em uma situação em que as hipóteses acima eram satisfeitas.

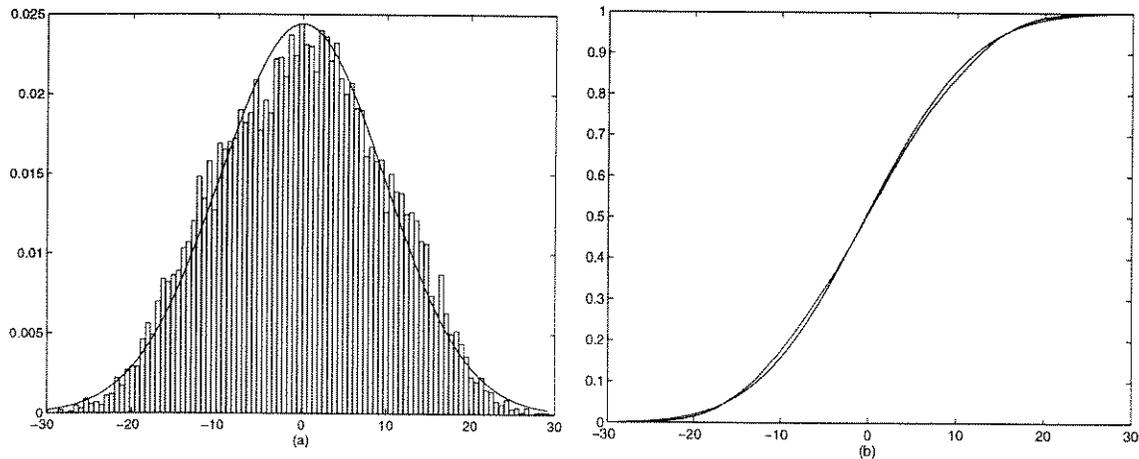


Figura 4.4: Estimativa da Densidade (a) e Distribuição (b) de Probabilidade de  $y(n)$ , Comparados com uma Distribuição Gaussiana.

Assim, se o olho do canal está fechado, podemos considerar que  $y(n)$  satisfaz as hipóteses do teorema de Bussgang. Podemos, então, escrever

$$E(\hat{a}(n) \hat{a}(n-i)) = K E(y(n) y(n-i)), \quad i = 0, 1, \dots, N. \quad (4.16)$$

Usando 4.15, concluímos que:

$$E(\hat{a}(n) \hat{a}(n-i)) = K E(y(n) y(n-i)), \quad i = 1, 2, \dots, N. \quad (4.17)$$

o que significa uma relação linear entre a autocorrelação dos símbolos de saída e a dos símbolos de entrada do dispositivo de decisão.

Entretanto, podemos construir um exemplo que mostra que a relação entre estas duas grandezas é não-linear. Isto é feito através da lei do arco-seno (cf. [7]). Consideremos a transmissão de um sinal 2-PAM. Temos, então  $\hat{a}(n) = \text{sgn}(y(n))$ . Neste caso, a função de autocorrelação de  $\hat{a}(n)$  é dada por:

$$E(\hat{a}(n)\hat{a}(n-i)) = \frac{2}{\pi} \arcsin\left(\frac{E(y(n)y(n-i))}{\sigma_y^2}\right), \quad i = 0, 1, 2, \dots, \quad (4.18)$$

o que corresponde a uma relação não-linear entre as autocorrelações dos sinais na entrada e na saída do dispositivo de decisão. Se tivermos constelações com mais elementos que a 2-PAM, a relação se torna ainda mais complexa (cf. [7]), mantendo a não-linearidade. Como 4.17 indica uma relação linear, temos aqui uma aparente contradição. A única maneira de resolvê-la é termos

$$E(\hat{a}(n)\hat{a}(n-i)) = E(y(n)y(n-i)) = 0, \quad i = 1, 2, \dots, N, \quad (4.19)$$

o que implica na decorrelação do sinal  $y(n)$ . Mas o teorema 4.2 garante que a decorrelação de  $y(n)$  implica na equalização do canal.

Concluimos assim que o critério  $J_\varepsilon(H)$  é de fato unimodal pois sempre que esta função custo for minimizada o canal estará equalizado. Na seqüência, mostraremos que os algoritmos propostos levam à minimização deste critério.

## 4.2 Análise dos Algoritmos Preditivos

O objetivo desta seção é mostrar que de fato os algoritmos propostos neste trabalho levam à equalização do canal. Para esta análise, usaremos o Método da ODE (do inglês *Ordinary Differential Equation*, equação diferencial ordinária). Este método foi desenvolvido com o intuito de permitir o estudo das propriedades de convergência de algoritmos recursivos estocásticos. Para este fim, ele permite associar uma ODE ao algoritmo em estudo, de forma que as propriedades de convergência deste algoritmo estejam relacionadas às trajetórias e à estabilidade da ODE. A vantagem desta associação é que o estudo das ODE consiste em uma área bem estabelecida do conhecimento (cf. [21]), e existem vários resultados poderosos que facilitam sua análise. Por outro lado, analisar diretamente um algoritmo recursivo é uma tarefa árdua e, em geral, infrutífera.

O método da ODE foi originalmente desenvolvido por Ljung em [22, 23] para a análise de algoritmos de identificação de sistemas estacionários. Uma característica destes algoritmos é que seu ganho de adaptação é variável e tende a zero quando  $t$  tende a infinito. Por isto, eles são chamados de algoritmos de ganho desvanecente (*Vanishing Gain Algorithms*). Esta característica facilita enormemente a sua análise e permite a obtenção de resultados muito fortes. Os teoremas propostos por Ljung garantem que, dadas algumas condições de regularidade, os algoritmos de ganho desvanecente convergem com probabilidade um para pontos estacionários estáveis da ODE associada. Por outro lado, esta mesma característica torna estes algoritmos inadequados para aplicações em ambientes não-estacionários, como é o caso dos sistemas de telecomunicações. A anulação assintótica do ganho de adaptação faz com que o algoritmo adapte cada vez menos os coeficientes do filtro adaptativo, até chegar a um ponto em que ele praticamente “desliga”. Assim, o algoritmo é cada vez menos capaz de acompanhar as variações do sistema.

Por isto, os algoritmos adaptativos usados em telecomunicações apresentam um ganho constante. A extensão do método da ODE para algoritmos de ganho constante foi proposta em [24] e [25]. Além de dificultar a análise dos algoritmos, o fato de o ganho ser constante enfraquece os resultados teóricos, e não podemos mais falar em convergência com probabilidade um, mas sim em convergência na média. No próximo item, apresentaremos o método da ODE para algoritmos de ganho constante e explicaremos alguns dos termos apresentados nesta introdução. Em seguida, mostraremos uma explicação intuitiva para este método, e depois discutiremos as funções de Lyapunov, que constituem uma poderosa ferramenta para a análise dos pontos estacionários de uma equação diferencial. No item 4.2.4 aplicaremos do método da ODE à análise do algoritmo LMS. Veremos em seguida que a aplicação deste método para o algoritmo GN é muito difícil. Assim, na seção 4.2.5 apresentaremos o método da ODE proposto por Ljung, e veremos porque é possível sua aplicação ao algoritmo GN. Finalmente, faremos a análise do algoritmo GN pelo método de Ljung.

Observe que centraremos a análise nos algoritmos LMS e GN. A razão para isto é que, como veremos no próximo capítulo, estes dois algoritmos se mostraram os mais eficientes para a equalização de canais.

#### 4.2.1 O Método da ODE para Algoritmos de Ganho Constante

A compreensão do método da ODE envolve um conceito importante da teoria de equações diferenciais: o de pontos estacionários estáveis. Seja a equação diferencial

$$\frac{dx(t)}{dt} = f(x), \quad (4.20)$$

onde  $x(t)$  é um vetor em  $\mathbb{R}^l$ . Uma solução desta equação diferencial é um vetor de  $\mathbb{R}^l$  que varia no tempo. Ou seja, a solução determina uma trajetória em  $\mathbb{R}^l$ . Um ponto estacionário de 4.20 é um valor  $\bar{x} \in \mathbb{R}^l$  tal que  $f(\bar{x}) = 0$ . Note que se  $x(t) = \bar{x}$  então  $dx(t)/dt = 0$ . Ora, a anulação da derivada implica que o valor de  $x(t)$  é constante e é por isto que este ponto se chama ponto estacionário: nele, o movimento da trajetória cessa.

Um ponto estacionário estável é aquele que de certa forma atrai as trajetórias que passam próximas a ele. Mais precisamente,  $\bar{x}$  é um ponto estacionário estável de 4.20 se todas as trajetórias que começam em alguma vizinhança de  $\bar{x}$  tendem a  $\bar{x}$ . Matematicamente, isto equivale a dizer que existe um  $\epsilon > 0$  tal que

$$\|x(0) - \bar{x}\| < \epsilon \Rightarrow \lim_{t \rightarrow \infty} x(t) = \bar{x}. \quad (4.21)$$

A bola em  $\mathbb{R}^l$  descrita por  $\|x(0) - \bar{x}\| < \epsilon$  é o chamado domínio de atração de  $\bar{x}$ .

De posse destes conceitos, podemos passar à descrição do método da ODE. Considere então um filtro adaptativo com vetor de coeficientes  $X(n)$ , adaptado segundo:

$$X(n+1) = X(n) + \mu V_n(X(n)), \quad (4.22)$$

onde  $\mu$  é um passo de adaptação e  $V_n(X(n))$  é a direção de adaptação, que obviamente depende do instante de tempo  $n$  e do valor presente de  $X(n)$ . Considere também que o algoritmo parte de um valor inicial  $X(0) = X_0$ .

O método da ODE associa a seguinte equação diferencial a este algoritmo:

$$\frac{dx(t)}{dt} = \bar{V}(x(t)), \quad (4.23)$$

onde  $x(t)$  é um vetor com a mesma dimensão de  $X(n)$  e  $\bar{V}(x(t))$  é a esperança de  $V_n(X(n))$ , calculada como se o vetor aleatório  $X(n)$  fosse um vetor determinístico igual a  $x(t)$ . Matematicamente,

$$\bar{V}(x(t)) = E[V_n(X(n))], \text{ para } X(n) = x(t). \quad (4.24)$$

Note que este valor depende de  $x(t)$  e não é aleatório, pois dado  $x(t)$  podemos determinar inequivocamente  $\bar{V}(x(t))$ . Além disto, consideraremos  $V_n(X(n))$  estacionário, de forma que  $\bar{V}(x(t))$  não depende de  $n$ . Assim, a solução de 4.23 é uma função determinística de  $t$ . Indicamos este fato escrevendo  $x(t)$  em letras minúsculas.

Finalmente, considere que  $y(t)$  seja a solução desta equação diferencial dada a condição inicial  $y(0) = X_0$ . Os teoremas desenvolvidos em [24, 25] garantem que é pequena a probabilidade de que a distância entre  $y(n\mu)$  e  $X(n)$ , adaptado por 4.22, seja grande. Em outras palavras, estes teoremas dizem que, na média, a evolução de  $X(n)$  a partir do valor inicial  $x_0$  acompanha a evolução de  $y(n\mu)$ . Esta é a chamada convergência na média ou em probabilidade. Matematicamente, os teoremas afirmam que existem duas constantes positivas  $C_1$  e  $C_2$  e uma função  $\epsilon(\mu)$  que vai a zero quando  $\mu$  vai a zero tais que, para todo  $S$  finito temos

$$P \left\{ \sup_{0 \leq n, \mu \leq S} \|X(n) - y(n\mu)\| > C_1 \epsilon(\mu) \right\} < C_2 \epsilon(\mu). \quad (4.25)$$

Este é o resultado que nos permite associar o comportamento do algoritmo ao da ODE. Para entender como isto é feito, considere que  $y(t)$  tende para um valor  $\bar{x}$  quando  $t \rightarrow \infty$ . Como  $X(n)$  está na média sempre próximo de  $y(n\mu)$ , isto nos leva à conclusão que  $X(n)$  deve de alguma maneira convergir para o mesmo  $\bar{x}$ . O inverso também parece natural. De fato, podemos afirmar o seguinte:

**Teorema 4.3** *Considere um filtro com vetor de parâmetros  $X(n)$  adaptado por*

$$X(n+1) = X(n) + \mu V_n(X(n)) \quad (4.26)$$

*e considere que as seguintes condições são satisfeitas:*

- *A entrada e a saída desejada do filtro são conjuntamente estacionárias.*
- *Os filtros envolvidos na adaptação são exponencialmente estáveis.*

*Então, para um valor suficientemente pequeno de  $\mu$ , o algoritmo converge na média para um valor  $\bar{x}$  se e somente se  $\bar{x}$  for um ponto estacionário estável da equação diferencial*

$$\frac{dx(t)}{dt} = \bar{V}(x(t)), \quad (4.27)$$

onde

$$\bar{V}(x(t)) = E[V_n(X(n))], \text{ para } X(n) = x(t). \quad (4.28)$$

Na realidade, este teorema está escrito em uma linguagem pouco precisa matematicamente. Este enunciado serve mais para dar uma idéia da conexão entre o algoritmo de adaptação 4.26 e a equação diferencial 4.27. Seu enunciado preciso e sua demonstração (cf. [24, 25]) são complexos e fogem ao escopo deste trabalho. Entretanto, através de algumas aproximações, podemos ter uma idéia intuitiva de que este resultado é razoável. Desenvolveremos este raciocínio no próximo item.

### 4.2.2 Explicação Intuitiva para o Método da ODE

Para obter alguma intuição a respeito do resultado do teorema 4.3, vamos escrever as variações de  $X(n)$  em sucessivas adaptações, a partir de uma certa iteração  $n$  até a iteração  $n + N$ :

$$\begin{aligned} X(n+1) - X(n) &= \mu V_n(X(n)) \\ X(n+2) - X(n+1) &= \mu V_{n+1}(X(n+1)) \\ &\vdots \\ X(n+N) - X(n+N-1) &= \mu V_{n+N-1}(X(n+N-1)). \end{aligned}$$

Somando estes termos, temos que

$$X(n+N) - X(n) = \mu \sum_{i=n}^{n+N-1} V_i(X(i)). \quad (4.29)$$

Se o passo de adaptação  $\mu$  for suficientemente pequeno, a variação do vetor de coeficientes é lenta, de forma que podemos considerar  $X(n) \approx X(n+1) \approx \dots \approx X(n+N-1)$  e, portanto,

$$X(n+N) - X(n) \approx \mu \sum_{i=n}^{n+N-1} V_i(X(n)). \quad (4.30)$$

Além disto, se considerarmos que  $V_n(X(n))$  é um processo ergódico, sua média pode ser estimada por:

$$E[V_n(X(n))] = \lim_{m \rightarrow \infty} \frac{1}{m} \sum_{i=1}^m V_i(X(n)). \quad (4.31)$$

Ora,  $E[V_n(X(n))] = \bar{V}(X(n))$ . Assim, para  $n$  e  $N$  suficientemente grandes, podemos invocar a lei dos grandes números e afirmar que:

$$\bar{V}(X(n)) \approx \frac{1}{N} \sum_{i=n}^{n+N-1} V_i(X(n)). \quad (4.32)$$

Substituindo este resultado em 4.30, concluímos que:

$$X(n+N) - X(n) \approx N \mu \bar{V}(X(n)). \quad (4.33)$$

Agora, se fizermos  $n = t$  e  $\Delta t = N \mu$ , a equação 4.33 sugere que escrevamos

$$\frac{X(t + \Delta t) - X(t)}{\Delta t} \approx \bar{V}(X(t)). \quad (4.34)$$

Esta equação pode ser vista como uma discretização da equação diferencial 4.27. Desta forma, na média, a trajetória dos parâmetros  $X(n)$  pode ser vista como uma solução discretizada desta ODE, fornecendo valores da solução  $y(t)$  para instantes de tempo  $t = N\mu$ .

Assim, este raciocínio intuitivo e pouco formal nos leva a crer que o comportamento de  $X(n)$  adaptado pela equação 4.26 de fato se aproxima do comportamento da solução de 4.27. A demonstração formal do teorema 4.3 busca determinar as circunstâncias em que as aproximações apresentadas aqui são válidas, bem como estabelecer qual o erro cometido ao se fazê-las.

### 4.2.3 Funções de Lyapunov

A importância do método da ODE reside no fato de que ele indica para quais pontos o algoritmo 4.26 pode convergir. A determinação destes pontos, entretanto, exige a determinação dos pontos estacionários estáveis da equação 4.27, o que nem sempre é uma tarefa trivial. Uma ferramenta muito útil neste sentido são as funções de Lyapunov.

Seja uma equação diferencial como 4.27, tendo  $\bar{x}$  como um ponto estacionário, e uma função  $L(x) : \mathbb{R}^l \rightarrow \mathbb{R}$ . Se, para uma dada vizinhança  $\|x - \bar{x}\| < \epsilon$  do ponto  $\bar{x}$ , tivermos que:

- $L(x)$  assume apenas valores positivos e apresenta um mínimo em  $\bar{x}$ .
- Para qualquer solução de 4.27  $L(x(t))$  é uma função decrescente de  $t$ .

então  $\bar{x}$  é um ponto estacionário estável de 4.27 cujo domínio de atração é a região  $\|x - \bar{x}\| < \epsilon$ . A função  $L(x)$  é uma função de Lyapunov associada à equação diferencial.

Para entender porque a existência de uma função de Lyapunov garante que  $\bar{x}$  é um ponto estacionário estável, considere uma trajetória  $x(t)$  que se inicie em um ponto  $x(0)$  satisfazendo  $\|x(0) - \bar{x}\| < \epsilon$ . Como  $L(x(t))$  é uma função decrescente do tempo, a trajetória  $x(t)$  evolui sempre no sentido de diminuir o valor de  $L(x(t))$ . Entretanto,  $L(x)$  é limitado inferiormente e portanto não pode diminuir indefinidamente, de forma que  $L(x)$  vai ser forçosamente minimizado. Ora, este mínimo ocorre em  $\bar{x}$ , por construção. Assim, qualquer trajetória que comece em um  $x(0)$  tal que  $\|x(0) - \bar{x}\| < \epsilon$  tende para  $\bar{x}$ . Portanto,  $\bar{x}$  é um ponto estacionário estável de 4.27, cujo domínio de atração é a região  $\|x - \bar{x}\| < \epsilon$ . É interessante notar que se conseguirmos uma função que satisfaça as condições impostas às funções de Lyapunov para qualquer valor de  $\epsilon$ , então o ponto  $\bar{x}$  é um ponto estacionário estável cujo domínio de atração é o espaço todo.

É possível demonstrar que existe uma função de Lyapunov associada a todo ponto estacionário estável de 4.27. Infelizmente, não existe nenhuma regra geral para a obtenção de uma função de Lyapunov associada a uma dada equação diferencial. Entretanto, como veremos a seguir, para algoritmos baseados no gradiente de uma função custo a determinação de uma função de Lyapunov é imediata.

### 4.2.4 Análise do Algoritmo LMS

Por comodidade, reproduziremos aqui o algoritmo LMS apresentado na seção 2.5.2. A adaptação do vetor de coeficientes  $H(n)$  do equalizador de fase não-linear é dada por:

$$H(n+1) = H(n) - \kappa \Xi(n), \quad (4.35)$$

onde o vetor  $\Xi(n)$  é a estimativa instantânea do gradiente da função custo  $J_\varepsilon(H)$ . Assim,

$$\Xi(n) = \frac{\partial |\varepsilon(n)|^2}{\partial H^*(n)}. \quad (4.36)$$

Usando a notação apresentada no desenvolvimento do método da ODE, temos

$$\begin{aligned} X(n) &= H(n) \\ V_n(X(n)) &= -\Xi(n) \end{aligned}$$

e, portanto,

$$\bar{V}(x) = -\mathbb{E} \left[ \frac{\partial |\varepsilon(n)|^2}{\partial x^*} \right] = -\frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x^*}. \quad (4.37)$$

A última igualdade é possível pois a esperança estatística é um operador linear. Assim, a equação diferencial associada ao algoritmo LMS é

$$\frac{dx(t)}{dt} = \bar{V}(x) = -\frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x^*}. \quad (4.38)$$

Considere agora a função  $L(x) = J_\varepsilon(H) = \mathbb{E} (|\varepsilon(n)|^2)$ . Obviamente, seu ponto de mínimo corresponde a um ponto estacionário de 4.38, pois neste ponto o gradiente de  $L(x)$  é nulo, o que também anula  $\bar{V}(x)$ . Se mostrarmos que  $L(x)$  é decrescente ao longo de qualquer trajetória  $x(t)$ , teremos mostrado que  $L(x)$  é uma função de Lyapunov associada a 4.38. Consideremos então a derivada de  $L(x(t))$  em relação a  $t$ . Aplicando a regra da cadeia,

$$\frac{dL(x(t))}{dt} = \left[ \frac{\partial L(x(t))}{\partial x} \right]^H \frac{dx(t)}{dt} = \left[ \frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x} \right]^H \frac{dx(t)}{dt}. \quad (4.39)$$

De 4.38,

$$\frac{dL(x(t))}{dt} = - \left[ \frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x} \right]^H \frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x} \quad (4.40)$$

$$= - \left\| \frac{\partial \mathbb{E} (|\varepsilon(n)|^2)}{\partial x} \right\|^2 \leq 0. \quad (4.41)$$

Observe que, neste caso, a igualdade só é válida se o gradiente for nulo, ou seja, no ponto de mínimo de  $L(x)$ . Portanto,  $L(x)$  é uma função decrescente de  $t$  ao longo de trajetórias  $x(t)$ , o que implica que  $L(x)$  é uma função de Lyapunov associada a 4.38.

Assim, concluímos que os pontos que minimizam  $L(x)$  correspondem a pontos estacionários estáveis de 4.38. Portanto, o algoritmo 4.35 converge na média para pontos de mínimo de  $L(x) = J_\varepsilon(H)$ . Ora, já vimos que nestes pontos o canal está equalizado. Se notarmos que em momento algum fizemos alguma hipótese a respeito do domínio de atração dos pontos estacionários, concluiremos que este corresponde ao espaço todo. Desta forma, independente de sua inicialização, o algoritmo LMS levará o vetor de coeficientes  $H$  para um mínimo de  $J_\varepsilon(H)$  e, conseqüentemente, à equalização do canal.

### 4.2.5 O Método da ODE de Ljung

Considere um algoritmo GN para adaptação de um vetor de coeficientes  $H(n)$ . Basicamente, as recursões deste algoritmo são dadas por:

$$\begin{aligned}\varepsilon(n) &= y(n) - \hat{y}(n) \\ R(n) &= R(n-1) + \alpha [Y(n)Y(n)^T - R(n-1)] \\ H(n) &= H(n-1) + \alpha R^{-1}(n)Y(n)\varepsilon(n).\end{aligned}\tag{4.42}$$

Observe que este algoritmo apresenta algumas semelhanças com o algoritmo LMS. Conforme vimos, a principal diferença entre os dois é que no LMS a direção de adaptação é dada por  $Y(n)\varepsilon(n)$ , enquanto que no GN esta direção é alterada pela matriz  $R^{-1}(n)$ , que leva em conta as derivadas segundas da função custo. Desta forma, o LMS pode ser visto como um caso especial do GN, onde a matriz  $R(n)$  é tomada como sendo a identidade.

A aplicação do método da ODE para algoritmos de ganho constante na análise de convergência do algoritmo GN é muito difícil. A razão disto é que este algoritmo envolve duas adaptações: a do vetor de coeficientes  $H(n)$  e a da matriz  $R(n)$ . Isto dificulta a obtenção de uma função de Lyapunov associada à ODE 4.27. A equação diferencial a ser estudada na análise do GN é

$$\frac{dx(t)}{dt} = \bar{V}(x) = E [R^{-1}(n)Y(n)\varepsilon(n)]\tag{4.43}$$

Como a matriz  $R(n)$  e o vetor  $Y(n)$  não são independentes, não podemos chegar facilmente a nenhuma conclusão interessante a respeito da convergência deste algoritmo.

O método da ODE de Ljung, entretanto, permite a análise em separado das duas recursões de 4.42, possibilitando a análise do algoritmo GN. Sua desvantagem é que ele só é aplicável quando o ganho de adaptação  $\alpha$  é variável e  $\alpha(n) \rightarrow 0$  quando  $n \rightarrow \infty$ . Conforme vimos, isto “desliga” o algoritmo, de forma que este só pode ser usado na equalização de canais estacionários. Entretanto, a demonstração de que o algoritmo GN converge em ambientes estacionários pode ser indicativa de sua eficácia em ambientes mais gerais. Passemos, assim, à apresentação do método da ODE proposto em [22, 23]. O método é, na realidade, mais genérico do que o apresentado aqui. Entretanto, para simplificar a apresentação, ele será adaptado para o contexto de interesse.

Assim, considere um algoritmo como o da equação 4.42, onde o ganho de adaptação é feito variável ( $\alpha = \alpha(n)$ ). Dadas algumas condições discutidas em [22, 23], prova-se que os únicos possíveis pontos de convergência deste algoritmo são os pontos estacionários estáveis do seguinte sistema de equações diferenciais:

$$\frac{dx(t)}{dt} = R_D^{-1}(t)\bar{V}(x(t))\tag{4.44}$$

$$\frac{dR_D(t)}{dt} = G(x(t)) - R_D(t),\tag{4.45}$$

onde

$$\bar{V}(x(t)) = E[Y(n)\varepsilon(n)] \text{ e } G(x(t)) = E[Y(n)Y^T(n)].\tag{4.46}$$

Mais ainda, os teoremas de Ljung garantem que o algoritmo 4.42 converge com probabilidade 1 para um ponto estacionário estável da ODE 4.44.

Note que o método apresentado na seção 4.2.1 é conceitualmente parecido com o de Ljung. A diferença é que este último permite a análise desacoplada das duas recursões presentes em algoritmos como o GN.

#### 4.2.6 Análise do Algoritmo GN

Para analisar o algoritmo GN para o equalizador de fase usando o método da ODE de Ljung, vamos inicialmente escrever este algoritmo na forma dada pela equação 4.42. Assim, notando que no algoritmo GN a matriz  $P(n) = \hat{S}^{-1}(n)$ , podemos escrever:

$$\begin{aligned} \varepsilon(n) &= y(n) - \hat{a}(n) \\ \hat{S}(n) &= \hat{S}(n-1) + \alpha(n) [Y(n) Y(n)^T - \hat{S}(n-1)] \\ \mathcal{H}(n) &= \mathcal{H}(n-1) + \alpha(n) P(n) Y(n) \varepsilon(n). \end{aligned} \quad (4.47)$$

Na seqüência, suporemos que este algoritmo satisfaz as condições dos teoremas de Ljung. Com isto, podemos associar a ele as seguintes equações diferenciais:

$$\frac{dx(t)}{dt} = R_D^{-1}(t) \bar{V}(x(t)) \quad (4.48)$$

$$\frac{dR_D(t)}{dt} = G(x(t)) - R_D(t), \quad (4.49)$$

onde  $G(x(t)) = E [Y(n) Y^T(n)]$  e  $\bar{V}(x) = E [Y(n) \varepsilon(n)]$ .

Vamos analisar primeiramente o comportamento de  $\hat{S}(n)$ , que está associado ao comportamento de  $R_D(t)$ . Se observarmos que  $G$  é uma função do tempo, podemos afirmar que a solução da equação diferencial

$$\frac{dR_D(t)}{dt} = G(x(t)) - R_D(t) \quad (4.50)$$

é dada por (cf. [21])

$$R_D(t) = e^{-t} R_D(0) + \int_0^t e^{-(t-s)} G(s) ds. \quad (4.51)$$

Para verificar esta afirmação, analisemos a derivada de  $e^t R_D(t)$  em relação a  $t$ . Pela regra do produto,

$$\frac{de^t R_D(t)}{dt} = \frac{de^t}{dt} R_D(t) + e^t \frac{dR_D(t)}{dt}. \quad (4.52)$$

Agora, usando 4.50,

$$\begin{aligned} \frac{de^t R_D(t)}{dt} &= e^t R_D(t) + e^t (G(t) - R_D(t)) \\ &= e^t G(t). \end{aligned} \quad (4.53)$$

Integrando ambos os lados da equação acima e lembrando que  $G(x(t))$  pode ser vista como uma função do tempo, temos:

$$e^t R_D(t) - e^0 R_D(0) = \int_0^t e^s G(s) ds \quad (4.54)$$

Multiplicando 4.54 por  $e^{-t}$ , chegamos à equação 4.51.

Ora,  $G(t) = E [Y(n)Y^T(n)]$  é claramente uma matriz definida positiva. Além disto, observe que em geral a matriz  $P$  é inicializada como um múltiplo positivo da identidade, de forma que  $\hat{S}(0) = P^{-1}(0)$  também o é. Como  $R_D(0) = \hat{S}(0)$ , temos que  $R_D(0)$  é um múltiplo positivo da identidade e, portanto, definida positiva. Assim, como  $R_D(t)$  é a soma de duas matrizes positivas definidas, concluímos que  $R_D(t)$  também é uma matriz definida positiva.

Estamos agora prontos para mostrar que os pontos estacionários estáveis das equações 4.27 correspondem aos pontos de mínimo de  $J_\varepsilon(\mathcal{H})$ . O raciocínio é análogo ao desenvolvido para o algoritmo LMS. Inicialmente, observemos que, pela construção do algoritmo GN,

$$Y(n)\varepsilon(n) = \frac{\partial |\varepsilon(n)|^2}{\partial \mathcal{H}(n)}. \quad (4.55)$$

Assim, podemos escrever

$$\bar{V}(x) = -E \left[ \frac{\partial |\varepsilon(n)|^2}{\partial x} \right] = -\frac{\partial E(|\varepsilon(n)|^2)}{\partial x}. \quad (4.56)$$

Considere agora a função  $L(x) = J_\varepsilon(\mathcal{H}) = E(|\varepsilon(n)|^2)$ . Em seu ponto de mínimo, seu gradiente é nulo, o que significa que  $\bar{V}(x)$  também é nulo. Portanto este ponto corresponde a um ponto estacionário da equação diferencial

$$\frac{dx(t)}{dt} = R_D^{-1}(t)\bar{V}(x(t)). \quad (4.57)$$

É interessante notar que como  $R_D^{-1}(t)$  é inversível, os únicos pontos estacionários desta equação diferencial são aqueles em que  $\bar{V}(x(t))$  se anula. Concluímos assim que o mínimo de  $L(x)$  corresponde ao único ponto estacionário de 4.57.

Para mostrarmos que este ponto é estável, devemos mostrar que  $L(x)$  é decrescente ao longo de qualquer trajetória  $x(t)$  que seja solução de 4.57. Ou seja, devemos mostrar que  $L(x)$  é uma função de Lyapunov associada a esta equação diferencial. Consideremos então a derivada de  $L(x(t))$  em relação a  $t$ . Aplicando a regra da cadeia,

$$\frac{dL(x(t))}{dt} = \left[ \frac{\partial L(x(t))}{\partial x} \right]^T \frac{dx(t)}{dt} = \left[ \frac{\partial E(|\varepsilon(n)|^2)}{\partial x} \right]^T \frac{dx(t)}{dt}. \quad (4.58)$$

De 4.57,

$$\frac{dL(x(t))}{dt} = - \left[ \frac{\partial E(|\varepsilon(n)|^2)}{\partial x} \right]^T R_D^{-1}(t) \frac{\partial E(|\varepsilon(n)|^2)}{\partial x}. \quad (4.59)$$

Mas, como  $R_D(t)$  é definida positiva,  $R_D^{-1}(t)$  também é. Assim,

$$v^T R_D^{-1}(t)v \geq 0, \quad (4.60)$$

para qualquer vetor  $v$ . Além disto,  $R_D^{-1}(t)$  é inversível, e portanto só teremos a igualdade na equação acima se  $v = 0$ . Em particular, se  $v$  é o gradiente de  $L(x)$ ,

$$\left[ \frac{\partial E(|\varepsilon(n)|^2)}{\partial x} \right]^T R_D^{-1}(t) \frac{\partial E(|\varepsilon(n)|^2)}{\partial x} \geq 0. \quad (4.61)$$

Concluimos finalmente que

$$\frac{dL(x(t))}{dt} \leq 0, \quad (4.62)$$

para qualquer trajetória  $x(t)$  que seja solução de 4.57. A igualdade só é observada quando o gradiente de  $L(x)$  se anula. Assim,  $L(x(t))$  é uma função decrescente de  $t$  ao longo de trajetórias  $x(t)$  que sejam soluções de 4.57, o que implica que  $L(x)$  é uma função de Lyapunov associada a 4.57. Portanto, os pontos estacionários estáveis de 4.57 correspondem àqueles que minimizam o critério  $J_\varepsilon(\mathcal{H})$ .

Concluimos assim que o algoritmo GN com ganho desvanecente leva à minimização de  $J_\varepsilon(\mathcal{H})$ , com probabilidade 1. Como vimos, esta minimização leva à equalização do canal. Assim, provamos que o algoritmo GN com ganho desvanecente é de fato eficaz na equalização de um canal estacionário. Este resultado, apesar de não poder ser diretamente aplicado ao algoritmo de ganho constante, serve como um indicativo de sua eficiência na equalização de canais não-estacionários.

### 4.3 Conclusão

Neste capítulo nos propusemos a fazer uma análise teórica dos algoritmos LMS e GN para equalizadores cegos baseados na equalização de magnitude e fase. Concentramos nossos esforços na análise do equalizador de fase, visto que o equalizador de magnitude é composto de um preditor, estrutura bastante estudada na literatura.

Assim, inicialmente revisamos alguns resultados obtidos em [3]. Principalmente, vimos que o critério de minimização do erro DD leva de fato à equalização do canal. Em seguida, mostramos a unimodalidade deste critério.

Finalmente, passamos à análise dos algoritmos propriamente ditos. Esta análise foi baseada no método da ODE, que consiste em associar uma equação diferencial ao algoritmo em estudo, de forma que as propriedades de convergência deste algoritmo estejam associadas à estabilidade da ODE. Vimos que existem duas técnicas de ODE, uma para algoritmos de ganho desvanecente e outra para algoritmos de ganho constante. Os algoritmos de ganho desvanecente não são capazes de acompanhar as não-estacionariedades inerentes aos canais de comunicações. Assim, todos os algoritmos propostos neste trabalho são de ganho constante.

Desta forma, apresentamos o método da ODE para algoritmos de ganho constante, bem como alguns conceitos relacionados à estabilidade de equações diferenciais e algumas ferramentas para analisá-las. Em seguida constatamos, a partir deste método, que o algoritmo LMS de fato leva à minimização do critério de decisão direta e, conseqüentemente, à equalização do canal.

Entretanto, vimos que o algoritmo GN dificilmente pode ser analisado por este método. Assim, apresentamos o método da ODE proposto por Ljung para a análise de algoritmos de ganho desvanecente e constatamos, a partir deste método, que o algoritmo GN com ganho desvanecente leva à equalização do canal. Conforme vimos, algoritmos de ganho desvanecente só podem ser aplicados a canais estacionários. Assim, a análise feita para o algoritmo GN é apenas um indício de seu bom funcionamento quando seu passo de adaptação for constante.

---

No próximo capítulo, faremos um outro tipo de análise dos algoritmos propostos. Esta análise será baseada na simulação do desempenho dos algoritmos na equalização de alguns tipos de canais encontrados em sistemas de telecomunicações. Buscaremos, neste estudo, constatar a unimodalidade dos algoritmos demonstrada teoricamente no presente capítulo. Mais do que isto, procuraremos analisar o desempenho dos algoritmos em termos de velocidade de convergência, informação que não é fornecida pelo método da ODE.

## Capítulo 5

# Resultados de Simulações

NESTE capítulo analisaremos o desempenho dos algoritmos propostos através de simulações, abordando diversas situações encontradas na prática: equalização de um canal passa-tudo, equalização de um modelo estacionário de um canal de rádio-móvel, equalização de um canal de microondas e, finalmente, equalização de um modelo não-estacionário de um canal de rádio-móvel. Nestas simulações, faremos um estudo comparativo entre os algoritmos propostos e técnicas já existentes, principalmente o algoritmo de Godard (CMA) e o algoritmo LMS para técnicas preditivas.

As curvas apresentadas neste capítulo para ilustrar o desempenho dos algoritmos mostrarão a evolução do quadrado do erro de decisão direta em função do número de iterações. Esta evolução depende da seqüência particular de dados sendo transmitida, e portanto possui um caráter aleatório. Assim, as curvas de aprendizado (*Learning Curves*) mostradas aqui foram obtidas através da mediação de 100 experimentos independentes. Em cada um destes experimentos os coeficientes do equalizador eram inicializados em um valor fixo (zero no caso da estrutura preditiva; 1 no *tap* central do filtro FIR no caso do algoritmo de Godard) e uma seqüência aleatória de dados que era, então, gerada e transmitida pelo canal a ser equalizado. Finalmente, o algoritmo era posto em funcionamento e a evolução do erro de decisão direta era observada.

Em todos os casos, a escolha dos parâmetros dos algoritmos buscou minimizar o número de iterações necessário para a convergência dos mesmos.

### 5.1 Canal Passa-Tudo

Inicialmente, analisaremos o desempenho do equalizador de fase isoladamente. Conforme vimos, o equalizador de fase supõe que o sinal em sua entrada já apresenta um espectro plano. Desta forma, considerando que a seqüência transmitida seja i.i.d, esta estrutura pode ser utilizada na equalização de um canal passa-tudo cuja saída possui, nesse caso, um espectro plano. De fato, a função do equalizador de magnitude é justamente transformar a cascata canal + equalizador de magnitude em um sistema passa-tudo. Nesta seção, simularemos o desempenho dos algoritmos preditivos na equalização de um canal passa-tudo cuja função de transferência é dada por:

$$F(z) = e^{j\pi/4} \frac{0.7 - z^{-1}}{1 - 0.7z^{-1}}. \quad (5.1)$$

Este canal apresenta um erro de fase constante igual a  $\pi/4$ . Os dados serão modulados em  $\pi/4$ -PSK. A ordem do equalizador usado nessas simulações foi 15, o que significa 15 coeficientes em cada estágio do equalizador de fase. Os parâmetros usados nas simulações estão mostrados na tabela 5.1.

	LMS	GN	RLS
Equalizador de Fase	$\kappa$ : 0,005	$w$ : 0,985 $E0$ : 7	$w$ : 0,99 $E0$ : 1
Ajuste de Fase ( $\theta$ )	$\eta$ : 0,005	$w$ : 0,9	$w$ : 0,9

Tabela 5.1: Parâmetros Usados na Simulação do Canal Passa-Tudo

As curvas de aprendizado para os algoritmos LMS, RLS e GN estão apresentadas na figura 5.1.

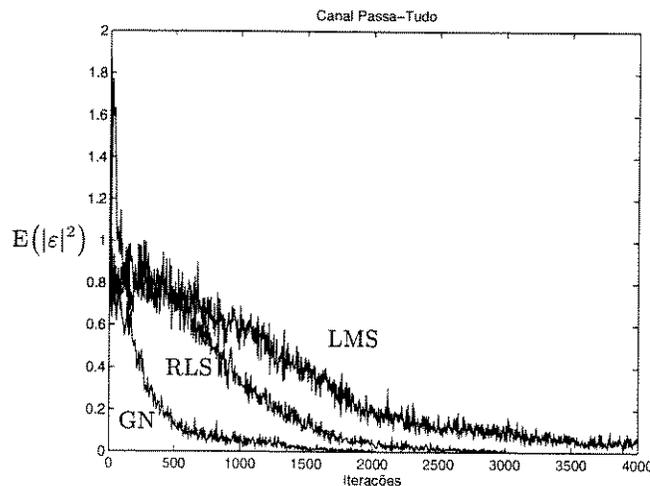


Figura 5.1: *Learning Curves* para o Canal Passa-Tudo

Observe que a velocidade de convergência do algoritmo GN é maior que a do RLS. Observe também que o algoritmo GN apresenta uma velocidade de convergência de três a quatro vezes maior que a do LMS. Este é um ponto intrigante pois, conforme veremos adiante, esta vantagem não se mantém quando a estrutura completa (equalizador de magnitude + equalizador de fase) é utilizada. Quando comparados a outras técnicas de equalização cega, observamos um desempenho significativamente melhor dos algoritmos preditivos. Em [26], por exemplo, o olho do canal é aberto somente após 10000 iterações.

## 5.2 Canal Rádio-Móvel: Ambiente Estacionário

Nesta seção, simularemos o desempenho dos algoritmos LMS, GN e RLS aplicados ao equalizador completo (equalizador de magnitude + equalizador de fase). O canal a ser equalizado é baseado em um modelo estacionário de um canal rádio-móvel. Este modelo segue as espe-

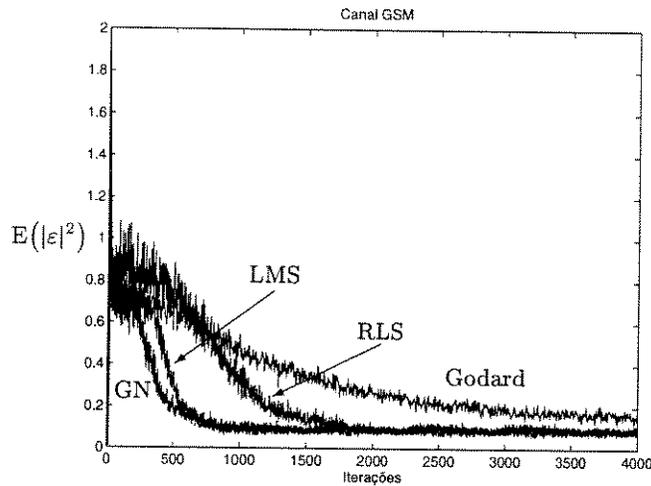


Figura 5.3: *Learning Curves* para o Canal Rádio-Móvel Estacionário

número de iterações. De fato, observamos que a velocidade de convergência dos coeficientes do preditor IIR é pouco menor do que a do erro DD, que está associada à convergência da estrutura completa (equalizador de magnitude + equalizador de fase). Desta forma, quando a estrutura completa é usada, é a velocidade de adaptação do equalizador de magnitude que dita a velocidade de convergência do equalizador. Como essa velocidade é praticamente igual para os algoritmos LMS e GN, vemos que não existe uma contradição entre a maior eficiência do algoritmo GN na adaptação do equalizador de fase e o desempenho semelhante dos dois equalizadores na adaptação da estrutura completa.

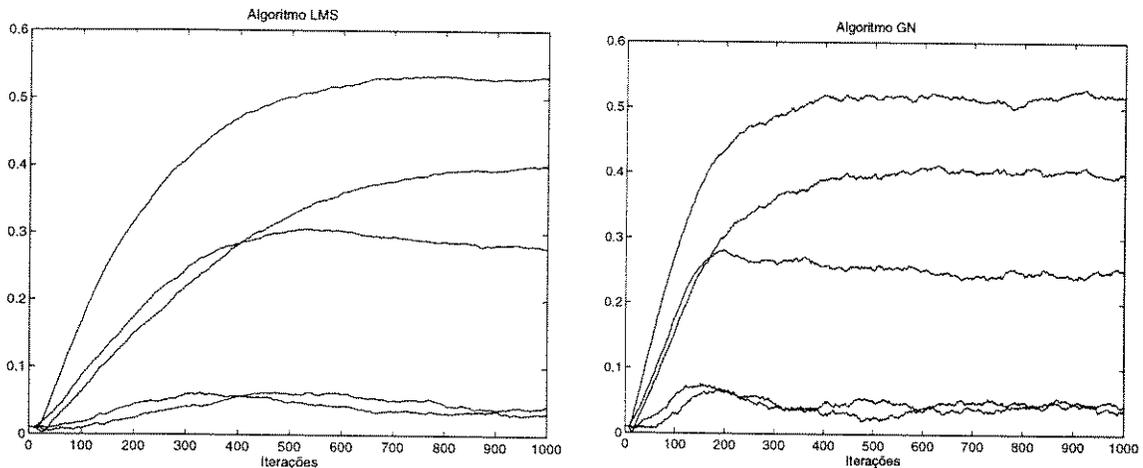


Figura 5.4: Evolução dos Valores Absolutos dos Coeficientes do Preditor IIR Adaptados pelos Algoritmos LMS (a) e GN (b).

Finalmente, é interessante notar que o desempenho do RLS manteve-se pior que o do

GN, tanto no presente caso quanto na equalização do canal passa-tudo. Conforme vimos, o RLS pode ser visto como um caso especial do GN, onde o passo de adaptação  $\alpha$  é tomado como 1. Assim, podemos imaginar que o RLS funciona como o GN com um arranjo de parâmetros fixo, onde não temos controle sobre o valor de  $\alpha$ . Nas simulações apresentadas, este arranjo de parâmetros se mostrou pouco favorável ao bom desempenho do equalizador. Por esta razão e tendo em vista que o RLS pode ser considerado um caso particular do GN, não mais consideraremos o algoritmo RLS nas simulações que se seguem.

### 5.3 Canal de Microondas

Nesta seção, analisaremos o desempenho do equalizador completo na equalização de um canal de microondas. O modelo deste canal foi apresentado em [20] e sua resposta ao impulso discreta está mostrada na tabela 5.3. É interessante notar que este canal apresenta um nulo espectral, conforme evidenciado na figura 5.5. Isto torna sua equalização uma tarefa particularmente difícil.

$h(0) = -0.1814E - 03 + i0.1085E - 02$	$h(1) = -0.2888E - 02 - i0.1310E - 04$
$h(2) = 0.1314E - 02 - i0.3950E - 02$	$h(3) = 0.2272E - 01 + i0.2702E - 02$
$h(4) = -0.1073E + 00 + i0.2183E - 01$	$h(5) = 0.6982E + 00$
$h(6) = -0.1387E + 00 - i0.6843E + 00$	$h(7) = -0.8670E - 04 + i0.1095E + 00$
$h(8) = -0.7162E - 02 - i0.2173E - 01$	$h(9) = 0.3610E - 02 - i0.2073E - 02$
$h(10) = 0.5867E - 03 + i0.2828E - 02$	

Tabela 5.3: Coeficientes da Resposta ao Impulso  $h(n)$  do Canal de Microondas

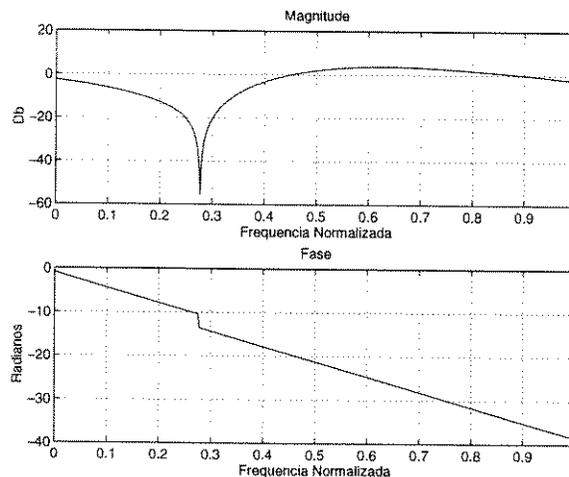


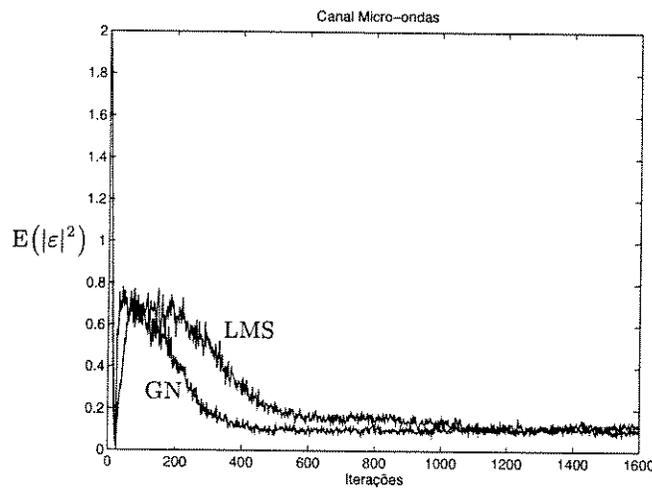
Figura 5.5: Resposta em Freqüência do Canal de Microondas

As simulações foram realizadas para os algoritmos GN e LMS. Os parâmetros usados estão mostrados na tabela 5.4. Os resultados estão mostrados na figura 5.6.

Notamos mais uma vez que os algoritmos LMS e GN apresentaram um desempenho

	LMS	GN
Equalizador de Fase	$\kappa$ : 0,004	$w$ : 0,983 $E0$ : 5
Ajuste de Fase ( $\theta$ )	$\eta$ : 0,0028	$w$ : 0,9
Equalizador de Magnitude	$\mu$ : 0,0028	$w$ : 0,987 $E0$ : 1
CAG	$\lambda$ : 0,004	$\lambda$ : 0,02

Tabela 5.4: Parâmetros Usados na Simulação do Canal de Microondas

Figura 5.6: *Learning Curves* para o Canal de Microondas

semelhante, convergindo em menos de 1000 iterações. Em nossas simulações, tentamos equalizar este canal de microondas com um filtro FIR adaptado pelo algoritmo de Godard. Neste caso, mesmo usando 55 *taps* no filtro, não conseguimos obter convergência em menos de 6000 iterações, o que nos levou a desistir deste algoritmo. Em [3], um equalizador baseado na técnica *Stop and Go* (cf. [15]) foi usado para equalizar este canal e a convergência foi obtida em aproximadamente 4500 iterações. Isto ilustra mais uma vez o desempenho superior das técnicas preditivas quando comparadas a outras técnicas de equalização cega.

## 5.4 Canal Rádio-Móvel: Ambiente Não-Estacionário

Na seção 5.2, mostramos o desempenho de equalizadores cegos na equalização de um modelo estacionário de um canal rádio-móvel. Este modelo, entretanto, está muito longe da realidade dos sistemas digitais de comunicações móveis. Os canais rádio-móvel reais apresentam uma característica fortemente não-estacionária (cf. [6]). Com o objetivo de testar os equalizadores cegos em situações mais próximas da realidade, usaremos nesta seção um simulador de canais rádio-móvel que leva em conta estas não-estacionariedades. A entrada do equalizador será conectada à saída deste simulador, como seria feito em uma situação

real.

Para simular a transmissão de um sinal digital por um canal D-AMPS (o sistema americano de telefonia celular), usamos o ambiente de simulação desenvolvido em [2]. Ele é baseado em um modelo de dois raios para o canal rádio-móvel, como o mostrado na figura 5.7. Nesta figura, vemos que o sinal modulado se propaga por dois percursos distintos até chegar ao terminal de recepção. Vemos também que o segundo percurso tem um atraso de  $\tau$  e uma perda de potência  $C$  em relação ao primeiro. Estes dois percursos podem modelar, por exemplo, a seguinte situação: o primeiro representa um sinal que chega à estação rádio-base (ERB) vindo direto do transmissor do telefone celular, enquanto o segundo representa um sinal que chega à ERB após ser refletido em alguma construção, como edifício, por exemplo. Na figura 5.7, estão ainda representados o ruído aditivo  $z(n)$ , bem como as grandezas  $\xi_1$  e  $\xi_2$ , que são duas variáveis aleatórias gaussianas correlacionadas e que simulam o efeito do deslocamento do móvel (efeito Doppler).

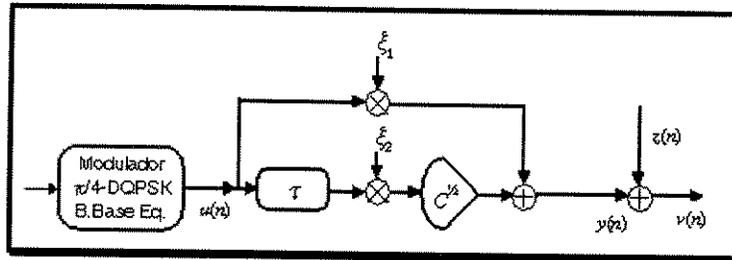


Figura 5.7: Modelo de Dois Raios do Canal Rádio-Móvel (Ref. [2]).

Nas simulações apresentadas aqui, supusemos um móvel que se desloca a uma velocidade de 4 km/h. Esta é aproximadamente a velocidade de uma pessoa caminhando e representa uma situação muito comum em telefonia celular. Supondo ainda a ausência de ruído e uma perda de potência de 15% no segundo percurso, simulamos a transmissão de 7520 símbolos de uma modulação  $\pi/4$ -DPSK pelo canal rádio-móvel. O perfil de potência do sinal na saída do canal está mostrado na figura 5.8

Finalmente, usamos a saída do canal D-AMPS para testar o desempenho dos algoritmos LMS e GN aplicados a um equalizador de magnitude de ordem 5 e um equalizador de fase de ordem 11. Os parâmetros usados nas simulações estão mostrados na tabela 5.5. Os resultados estão mostrados na figura 5.9 e 5.10, respectivamente. Observe que aqui, devido ao caráter não-estacionário do canal, não faz sentido falar em curvas de aprendizado.

Mais uma vez observamos que os dois algoritmos apresentaram um desempenho semelhante. Além disso, o desempenho é aparentemente não satisfatório, pois o erro DD assume muitas vezes valores altos, indicando situações em que o equalizador é incapaz de abrir o olho do canal. Mais ainda, a variação do erro DD, transitando entre períodos em que ele se mantém baixo para outros em que ele se eleva, indica que nem sempre os algoritmos são capazes de acompanhar as variações do canal rádio-móvel. Entretanto, podemos observar que, em vários períodos, o erro DD é baixo e, portanto, o equalizador consegue manter o olho do canal aberto. Além disto, observamos que em algumas situações o erro DD passa de um período com valores altos, indicando que o olho do canal está fechado, para outro em que seus valores são pequenos, indicando uma abertura do olho. Isto é uma propriedade

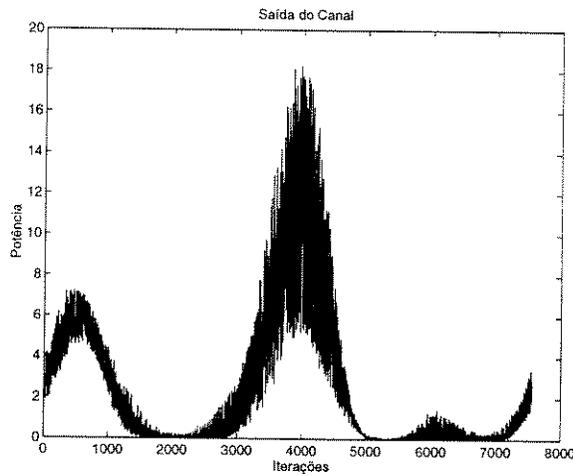


Figura 5.8: Potência da Saída do Canal Rádio-Móvel

	LMS	GN
Equalizador de Fase	$\kappa$ : 0,004	$w$ : 0,983 $E0$ : 5
Ajuste de Fase ( $\theta$ )	$\eta$ : 0,0028	$w$ : 0,9
Equalizador de Magnitude	$\mu$ : 0,0028	$w$ : 0,987 $E0$ : 1
CAG	$\lambda$ : 0,004	$\lambda$ : 0,02

Tabela 5.5: Parâmetros Usados na Simulação do Canal Rádio-Móvel Não-Estacionário

muito interessante dos algoritmos: eles conseguem se “recuperar”, abrindo o olho do canal mesmo após uma situação desfavorável em que este estava fechado. Este resultado indica um bom potencial de uso de técnicas preditivas de equalização em sistemas de comunicações móveis.

A capacidade de abertura do olho do canal está ilustrada nas figuras 5.11 e 5.12. Nestas figuras, mostramos as constelações na saída dos equalizadores adaptados pelos algoritmos LMS e GN, respectivamente. Estas constelações mostram apenas os períodos em que o olho do canal está aberto, o que corresponde aos seguintes intervalos de iterações:

- {[150, 1100]; [2000, 2300]; [5200, 5600]} para o algoritmo LMS
- {[60, 1250]; [3500, 4900]; [5200, 5550]} para o GN.

Note que o desempenho dos algoritmos é de fato semelhante, apesar de o GN manter o olho aberto durante mais tempo.

## 5.5 Conclusão

Através das simulações apresentadas aqui, podemos confirmar alguns resultados obtidos teoricamente. Para isto, observemos que as curvas apresentadas são uma média de 100 ex-

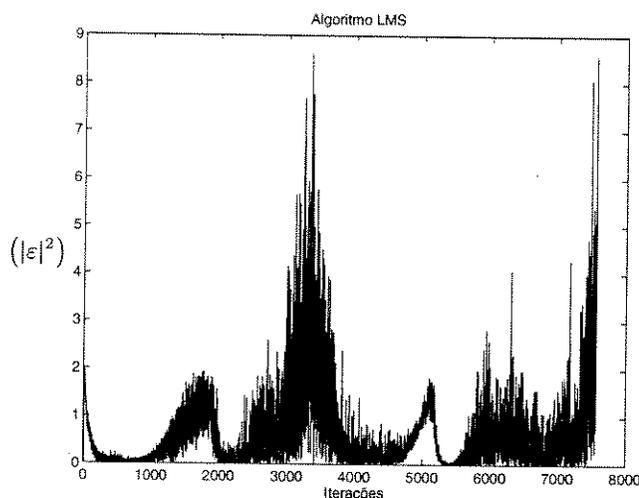


Figura 5.9: Evolução do Erro DD para o Canal Rádio-Móvel: Algoritmo LMS

perimentos independentes e que em nenhuma delas pudemos constatar a convergência para um mínimo local. Isto é um forte indicativo de que a eficiência dos algoritmos, constatada teoricamente no capítulo anterior, de fato se observa em situações mais práticas.

Os resultados de simulação obtidos neste capítulo nos permitem chegar a algumas conclusões interessantes a respeito da velocidade de convergência dos algoritmos de equalização cega.

Em primeiro lugar, as técnicas preditivas se revelaram consideravelmente mais velozes do que algumas técnicas mais tradicionais, tais como o algoritmo de Godard e o “Stop-and-Go”. Obviamente, não podemos tomar isto como uma conclusão geral, mas, ao menos nas simulações realizadas para testar os algoritmos, a vantagem das técnicas preditivas foi sempre muito clara.

Em segundo lugar, vale destacar o pior desempenho do algoritmo RLS quando comparado ao GN. Conforme havíamos observado na dedução destes algoritmos, eles apresentam muitas semelhanças e, de certa forma, o algoritmo RLS pode ser visto como uma caso especial do GN, onde o parâmetro  $\alpha$  presente neste último fica restrito a um único valor,  $\alpha = 1$ . Isto pode ser usado para explicar esta diferença de desempenho. Por permitir um controle sobre  $\alpha$ , o algoritmo de GN permite um melhor ajuste fino sobre seus parâmetros e, conseqüentemente, uma melhor velocidade de convergência.

Finalmente, é interessante comparar os desempenhos dos algoritmos LMS e GN. O GN se mostrou muito mais eficiente do que o LMS quando ambos foram usados para adaptar o equalizador de fase isoladamente. Entretanto, quando estes algoritmos foram usados com a estrutura completa de equalização, seus desempenhos tornaram-se muito próximos. Uma possível explicação para este fato é que o desempenho destes algoritmos na adaptação dos coeficientes do preditor IIR é muito semelhante, e consideravelmente lento. Assim, é a adaptação do preditor de magnitude que controla a velocidade de convergência da estrutura completa, de forma que o melhor desempenho do GN na adaptação do equalizador de fase perde importância neste contexto.

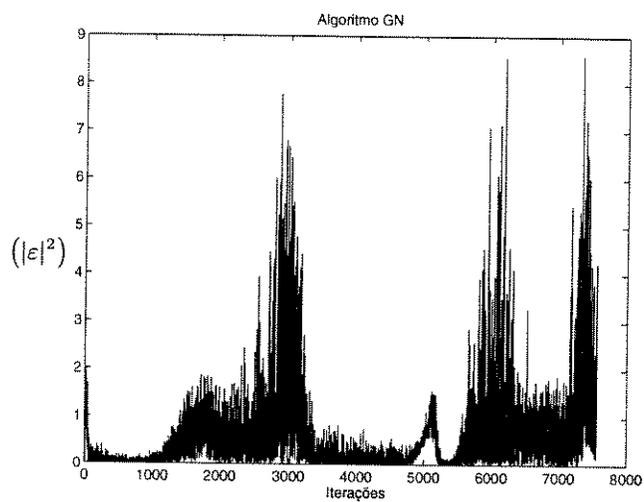


Figura 5.10: Evolução do Erro DD para o Canal Rádio-Móvel: Algoritmo GN

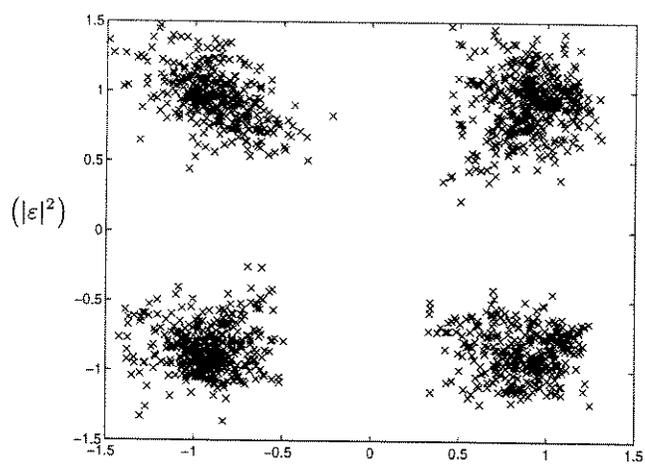


Figura 5.11: Constelação na Saída do Equalizador: Algoritmo LMS

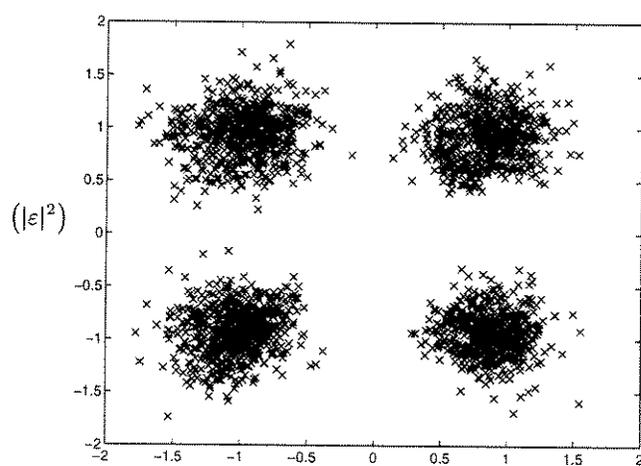


Figura 5.12: Constelação na Saída do Equalizador: Algoritmo GN

## Capítulo 6

# Conclusões

Neste trabalho, fizemos um estudo detalhado das técnicas preditivas de equalização cega baseadas na cascata de um equalizador de magnitude e de um equalizador de fase.

Inicialmente, apresentamos o problema da equalização de canais de telecomunicações e mostramos algumas soluções para este problema. As soluções clássicas envolvem a transmissão de uma seqüência de treinamento, o que diminui a eficiência do sistema. Surge daí o interesse em técnicas de equalização que dispensem o uso deste treinamento. Estas técnicas dão origem aos equalizadores cegos. Entre estas técnicas, destacam-se as técnicas preditivas de equalização cega e, entre estas, destaca-se a equalização de magnitude e fase. Esta estrutura possui a vantagem de ser não-linear, o que possibilita seu bom funcionamento em situações em que o canal introduz um nulo espectral. Neste ponto, esta estrutura se assemelha a um DFE.

Para esta estrutura, foram apresentados dois novos algoritmos: o RLS e o GN. Em seguida, fizemos uma análise teórica da estrutura do equalizador de fase e dos algoritmos RLS e GN associados. Provaram-se, sob algumas condições pouco restritivas, os seguintes fatos:

- O critério de decisão direta associado ao equalizador de fase é unimodal.
- Os algoritmos LMS e GN de fato minimizam o critério. Assim, devido à sua unimodalidade, estes algoritmos levam à equalização do canal independente de sua inicialização.

Não foi possível analisar o algoritmo GN supondo que seu ganho de adaptação seja constante. Ainda que isto seja uma característica fundamental dos algoritmos usados em sistemas de telecomunicações, o ganho constante dificulta muito a análise deste algoritmo. Assim, somente conseguimos demonstrar que este algoritmo leva à equalização do canal se o seu ganho for desvanecente. Este resultado mostra que o GN pode ser aplicado na equalização de canais estacionários, e é um indício da sua eficácia em canais mais gerais.

Os resultados de simulação nos levaram a algumas conclusões interessantes:

- O desempenho das técnicas preditivas se mostrou sempre superior ao das técnicas de Bussgang. Houve situações em que um algoritmo de Bussgang não foi de maneira alguma capaz de equalizar um canal para o qual a técnica preditiva obteve sucesso.
- O algoritmo GN se mostrou sempre mais eficiente do que o RLS. Vimos que isto pode ser explicado pelo fato de que o RLS é um caso particular do GN.

- Na equalização de um canal passa-tudo, onde apenas o equalizador de magnitude era testado, o GN se mostrou cerca de 4 vezes mais veloz do que o LMS. Entretanto, em outras situações, o desempenho destes algoritmos se mostrou semelhante. Este resultado parece indicar um problema no acoplamento entre as adaptações dos equalizadores de magnitude e o de fase.
- Os resultados teóricos a respeito da unimodalidade do critério de Decisão Direta e do comportamento dos algoritmos propostos foram confirmados. De fato, em momento algum observamos convergência dos algoritmos para um mínimo local.
- Quando as estruturas de equalização de magnitude e fase são usadas, a velocidade de convergência dos algoritmos propostos é controlada pela velocidade de convergência do preditor IIR. Para esta estrutura, ambos os algoritmos apresentam um desempenho semelhante. Isto pode ser uma explicação para a discrepância observada na velocidade de convergência dos algoritmos LMS e GN. Este último se revelou muito mais rápido que o LMS quando apenas o equalizador de fase era usado. Entretanto, em situações onde o equalizador completo foi usado, a velocidade de convergência dos dois algoritmos foi praticamente a mesma.

O estudo de equalizadores cegos é uma área do conhecimento relativamente nova, que vem experimentando um crescimento acelerado. Novas propostas e resultados surgem constantemente. Neste sentido, este trabalho se propôs a apresentar uma colaboração para o estudo dos equalizadores preditivos e, evidentemente, deixa várias questões em aberto.

Inicialmente, não foi estudado o comportamento da estrutura proposta em ambientes com Relação-Sinal-Ruído baixa. A demonstração da unimodalidade do critério DD, assim como as simulações apresentadas aqui, supuseram a ausência do ruído aditivo. Levar em conta este ruído é fundamental para uma eventual aplicação das técnicas preditivas em situações práticas, onde a SRN é sempre baixa.

Finalmente, seria interessante estudar a extensão das técnicas preditivas para a equalização de canais MIMO (do inglês *Multiple Input - Multiple Output*). O modelamento do canal de comunicações como um sistema MIMO é aplicável em situações onde o sinal recebido provém de múltiplos transmissores distintos, e a saída do canal é composta de múltiplas seqüências digitais. Canais de múltiplas entradas são encontrados, por exemplo, no sistema CDMA de telefonia celular, onde vários usuários compartilham uma mesma faixa de freqüência.

Quanto às múltiplas seqüências na saída do canal, elas são geradas basicamente de duas formas. Em uma delas, várias antenas (*Antenna Arrays*) são usadas para captar o sinal transmitido, de forma que cada uma delas gere uma seqüência digital correspondente ao sinal recebido. Temos, neste caso, um sistema com vários sensores. Também é possível obter múltiplas saídas do canal usando apenas um sensor. Neste caso, o sinal captado pelo receptor é amostrado várias vezes. Estas amostragens são feitas na taxa nominal de bits, mas com diferentes defasagens, gerando diferentes seqüências digitais. Obtém-se, neste caso, os chamados equalizadores fracionários.

No equalizador, as múltiplas seqüências digitais à saída do canal são processadas independentemente e, em seguida, combinadas de forma a gerar uma estimativa do sinal transmitido. As propriedades de equalizadores com esta característica têm sido intensamente

estudadas na literatura, e várias vantagens destes sistemas foram determinadas. Entre outras, prova-se que é possível equalizar um canal MIMO usando apenas as estatísticas de segunda ordem do sinal recebido, ou usando apenas preditores. Desta forma, a aplicação das técnicas preditivas a canais MIMO desperta imediato interesse e, possivelmente, levará à obtenção de algoritmos ainda mais eficientes para a solução do problema da equalização cega.

# Referências Bibliográficas

- [1] C. A. F. da Rocha and J. M. T. Romano, "A unimodal adaptation criterion for a nonlinear iir self-learning equalizer," in *Proc. 5<sup>th</sup> CMTC/GLOBECOM'96*, (London), Nov. 1996.
- [2] J. E. V. Bautista, *Performance Analysis of the  $\pi/4$ -DQPSK Modem Applied to the Mobile Radio Channel*. PhD thesis, CTA-ITA, Brazil, 1994. (in Portuguese).
- [3] C. A. F. Rocha, *Técnicas Preditivas para Equalização Autodidata*. PhD thesis, DECOM/FEEC/UNICAMP, 1996.
- [4] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1989.
- [5] S. Benedetto, E. Biglieri, and V. Castellani, *Digital Transmission Theory*. Prentice Hall International Inc., 1987.
- [6] M. D. Yacoub, *Foundations of Mobile Radio Engineering*. CRC Press, 1993.
- [7] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, third ed., 1991.
- [8] S. Haykin, *Adaptive Filter Theory*. Prentice-Hall International, 1991.
- [9] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*. Prentice-Hall International Editions, 1975.
- [10] A. Benveniste, M. Goursat, and G. Rouget, "Robust identification of a nonminimum phase system: Blind adjustment of a linear equalizer in data communications," *IEEE Trans. on Automatic Control*, vol. AC-25, pp. 385–398, June 1980.
- [11] O. Shalvi and E. Weinstein, "New criteria for blind deconvolution of nonminimum phase systems," *IEEE Trans. on Information Theory*, vol. 36, pp. 504–519, Mar. 1993.
- [12] S. Bellini, "Busgang techniques for blind deconvolution and equalization," in *Blind Deconvolution* (S. Haykin, ed.), pp. 8–54, Prentice-Hall Information and System Sciences Series, 1994.
- [13] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Trans. on Communications*, vol. COM-23, pp. 679–682, June 1975.

- [14] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Trans. on Communications*, vol. COM-28, pp. 1867–1875, Nov. 1980.
- [15] G. Picchi and G. Prati, "Blind equalization and carrier recovery using a 'stop-and-go' decision directed algorithm," *IEEE Trans. on Communications*, vol. COM-35, pp. 871–883, Aug. 1984.
- [16] O. Macchi and E. Eweda, "Convergence analysis of self-adaptive equalizers," *IEEE Trans. on Information Theory*, vol. IT-30, pp. 161–176, Mar. 1984.
- [17] O. Macchi, *Adaptive Processing: the Least Mean Square Approach with Applications in Transmission*. John Wiley & Sons Ltda., 1995.
- [18] C. R. Johnson, Jr., "Adaptive iir filtering: Current results and open issues," *IEEE Trans. on Information Theory*, vol. 30, pp. 237–250, Mar. 1984.
- [19] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*. Cambridge, MA: MIT Press, 1983.
- [20] J. C. M. Mota, *Equalização Adaptativa em Sistemas Rádio-Digitais: das Técnicas em Frequência à Teoria da Desconvolução Autodidata*. PhD thesis, DECOM/FEEC/UNICAMP, 1992.
- [21] R. Bellman, ed., *Stability Theory of Differential Equations*. New York: McGraw-Hill, 1953.
- [22] L. Ljung, "On positive real transfer functions and the convergence of some recursive schemes," *IEEE Trans. on Automatic Control*, vol. AC-22, pp. 539–551, Aug. 1977.
- [23] L. Ljung, "Analysis of recursive stochastic algorithms," *IEEE Trans. on Automatic Control*, vol. AC-22, pp. 551–575, Aug. 1977.
- [24] A. Benveniste, M. Goursat, and G. Ruget, "Analysis of stochastic approximation schemes with discontinuous and dependent forcing terms with application to data communication algorithms," *IEEE Trans. on Automatic Control*, vol. AC-25, pp. 1042–1058, Dec. 1980.
- [25] H. J. Kushner and A. Shwartz, "Weak convergence and asymptotic properties of adaptive filters with constant gains," *IEEE Trans. on Information Theory*, vol. IT-30, pp. 177–182, Mar. 1984.
- [26] Z. Ding and R. A. Kennedy, "Blind adaptive equalizers for quadrature amplitude modulated communication systems based on convex cost functions," *Optical Engineering*, vol. 31, pp. 1189–1199, June 1992.
- [27] K. Hilal, *Fast Algorithms for Self-Learning Equalization – Application to the Mobile Radio Channel*. PhD thesis, École Nationale Supérieure des Télécommunications, France, 1993. (in French).