

SUPRESSÃO CEGA DE INTERFERÊNCIAS PARA CDMA BASEADA EM  
ESTATÍSTICAS DE SEGUNDA ORDEM

Tadeu Nagashima Ferreira

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS  
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA  
UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS  
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE  
EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Aprovada por:

---

Prof. Paulo Sergio Ramirez Diniz, Ph.D.

---

Prof. Sergio Lima Netto, Ph.D.

---

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

---

Prof. Raimundo Sampaio Neto, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

JANEIRO DE 2005

FERREIRA, TADEU NAGASHIMA

Supressão Cega de Interferências  
para CDMA Baseada em Estatísticas de  
Segunda Ordem [Rio de Janeiro] 2005

XVIII,167 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2005)

Tese - Universidade Federal do Rio  
de Janeiro, COPPE

1.Equalização Cega 2.Equalização

Adaptativa 3.*Set-Membership* 4.CDMA

I.COPPE/UFRJ II.Título (série)

## Agradecimentos

Agradeço à minha namorada Luciana Fernandes pelo apoio dado em inúmeros momentos deste trabalho e por ser uma pessoa maravilhosa. Agradeço também a meus pais, Helio e Nair, pelo grande esforço logístico e financeiro que viabilizaram o mestrado.

Agradeço ao Prof. Paulo Diniz, o farol que iluminou os caminhos a serem trilhados. Agradeço ao Prof. Sergio Lima Netto, o guia do caminho trilhado, sem o qual simplesmente nada disso ter sido feito ou escrito. Agradeço também aos demais professores e funcionários da área de processamento de sinais da COPPE.

Agradeço ao Prof. Marcello Campos pela grande ajuda neste trabalho, tanto em seu decorrer como nas correções da tese. Agradeço ao Prof. Raimundo Sampaio Neto pelas grandes contribuições que melhoraram o conteúdo desta tese.

Agradeço aos colegas da turma 2003 do mestrado em processamento de sinais: Filipe Diniz, Leonardo Baltar, Ana Luisa Santos, Michel Tcheou, Maurício Quélhas, Marcello Artimos Neves, Carlo Marcello, Ygor Ururahy, Danillo Graziosi, Luís Guilherme Uzeda, Luciana Leite, Ricardo Ramos, José Marcio Faier. Tenho tantas coisas a agradecer a vocês, agradeço então simplesmente por vocês existirem. Agradeço também, e muito, aos demais amigos da comunidade CGTOTAL, pelos dias movidos à cafeína e cevada.

Agradeço também aos colegas Bruno Costa, Rodrigo Torres e novamente Luis Guilherme Uzeda por mais uma vez seguirem os mesmos caminhos que eu e por serem grandes companheiros.

Agradeço aos doutorandos João Batista, Fábio Freeland, Miguel Furtado, Lisandro Lovisolo e Augusto Dantas pelos inúmeros conselhos e ensinamentos. Agradeço também a Bernardo Costa e Wallace Martins, também pelos ensinamentos, mostrando que os mais jovens tem realmente muita coisa a ensinar.

Agradeço à Faperj, pelos 12 meses de bolsa e à Capes, pelos 2 meses.

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

SUPRESSÃO CEGA DE INTERFERÊNCIAS PARA CDMA BASEADA EM  
ESTATÍSTICAS DE SEGUNDA ORDEM

Tadeu Nagashima Ferreira

Janeiro/2005

Orientadores: Paulo Sergio Ramirez Diniz

Sergio Lima Netto

Programa: Engenharia Elétrica

Nos últimos anos, um conjunto significativo de algoritmos de equalização cega foram desenvolvidos. Esses novos algoritmos têm como principal característica o uso de apenas estatísticas de segunda ordem do sinal de entrada para eliminar a interferência intersimbólica. Com isso, os algoritmos se tornam mais rápidos e evitam a presença de mínimos locais durante o processo de minimização do erro.

Vários dos novos algoritmos de equalização cega são colocados para um ambiente multiusuário. Esses algoritmos são particularmente importantes para sistemas de comunicação sem fio, especialmente sistemas de telefonia celular.

Nesta tese é realizado um estudo comparativo entre novos métodos de equalização cega para ambiente multiusuário. São apresentados e implementados algoritmos representativos de três famílias de equalizadores: baseados em subespaço, variância mínima de saída e predição linear. Além disso, são projetadas e implementadas algumas modificações nos algoritmos de variância mínima e predição linear usando os conceitos de reuso de dados e atualização adaptativa seletiva.

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

BLIND SUPPRESSION OF INTERFERENCES FOR CDMA BASED ON  
SECOND-ORDER STATISTICS

Tadeu Nagashima Ferreira

January/2005

Advisors: Paulo Sergio Ramirez Diniz

Sergio Lima Netto

Department: Electrical Engineering

Recently, several important algorithms for blind equalization have been proposed. The main characteristic of these new algorithms is the strict use of second-order statistics of the input signal in order to avoid intersymbol interference. As a result, the algorithms become faster and avoid local minima in the process of error minimization.

In this work, new algorithms are presented which are suitable for a multiuser framework. These algorithms are particularly important for applications in wireless communications, specially for cell phone systems.

This thesis compares the performance of some new blind equalization methods for a multiuser framework. In this work, we present and implement algorithms belonging to three different classes of equalizers: the ones based on subspace, minimum output variance, and linear prediction. Besides that, we design and implement some modifications on the minimum variance algorithm and linear prediction algorithm, using the ideas of data-reusing and selective adaptive updates.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivações . . . . .	1
1.2	A Tese . . . . .	4
<b>2</b>	<b>Equalizadores</b>	<b>6</b>
2.1	Introdução . . . . .	6
2.2	Canais ideais . . . . .	7
2.3	Interferência Intersimbólica . . . . .	8
2.4	Algoritmos de Equalização . . . . .	9
2.4.1	MLSE - <i>Maximum Likelihood Sequence Estimator</i> . . . . .	10
2.4.2	Equalizador Linear <i>Zero-Forcing</i> . . . . .	12
2.4.3	Equalizador Linear MMSE . . . . .	14
2.4.4	DFE - <i>Decision Feedback Equalizer</i> . . . . .	16
2.4.5	FSE - <i>Fractionally Spaced Equalizer</i> . . . . .	18
2.5	Comparação dos Equalizadores . . . . .	19
2.6	Equalização Adaptativa . . . . .	19
2.6.1	<i>Zero-Forcing</i> Adaptativo . . . . .	20

2.6.2	Implementações Usando LMS . . . . .	20
2.6.3	MLSE Adaptativo . . . . .	22
2.6.4	DFE Adaptativo . . . . .	23
2.6.5	Implementações Baseadas em RLS . . . . .	23
2.7	Conclusões . . . . .	24
<b>3</b>	<b>Equalização Cega</b>	<b>25</b>
3.1	Introdução . . . . .	25
3.2	Tipos de Equalização Cega . . . . .	26
3.3	As Origens da Equalização Cega . . . . .	28
3.3.1	Algoritmo de Sato . . . . .	28
3.3.2	Algoritmo BGR . . . . .	29
3.3.3	Algoritmos <i>Stop-and-Go</i> . . . . .	29
3.3.4	Algoritmo de Bussgang . . . . .	30
3.3.5	Algoritmo de Godard . . . . .	30
3.4	Equalização Cega para Sistemas SIMO . . . . .	31
3.5	Equalização Cega Utilizando Cicloestacionaridade . . . . .	33
3.6	Conclusões . . . . .	35
<b>4</b>	<b>CDMA</b>	<b>36</b>
4.1	Introdução . . . . .	36
4.2	Espalhamento Espectral . . . . .	37
4.3	Códigos de Espalhamento . . . . .	38
4.3.1	Códigos de Walsh-Haddamard . . . . .	38

4.3.2	Códigos de Gold . . . . .	39
4.4	Receptor RAKE . . . . .	40
4.5	Codificação de Canal . . . . .	41
4.6	Deteção Multiusuário . . . . .	42
4.6.1	Detetor por Descorrelação . . . . .	44
4.6.2	Detetor MMSE . . . . .	45
4.6.3	Detetor por Cancelamento Sucessivo de Interferência . . . . .	45
4.7	Importância da Deteção Multiusuário . . . . .	46
4.8	Modelo de CDMA Implementado . . . . .	48
4.9	Conclusões . . . . .	50
<b>5</b>	<b>Equalização Cega Usando Subespaço</b>	<b>51</b>
5.1	Introdução . . . . .	51
5.2	Separação do Subespaço do Sinal e do Ruído . . . . .	52
5.3	Detetor Multiusuário Cego Utilizado . . . . .	53
5.4	Deteção Multiusuário Cega com ISI . . . . .	55
5.5	Questões sobre o Algoritmo . . . . .	60
5.6	Conclusões . . . . .	60
<b>6</b>	<b>Equalização Cega Usando Variância Mínima de Saída</b>	<b>63</b>
6.1	Introdução . . . . .	63
6.2	Equivalência entre MOV e MMSE . . . . .	64
6.3	Critério de Variância Mínima de Saída com Restrições . . . . .	65
6.4	Cancelador de Bandas Laterais Generalizado . . . . .	66



6.5	Atualização Adaptativa da Versão GSC . . . . .	68
6.6	Questões sobre o Algoritmo . . . . .	70
6.7	Conclusões . . . . .	71
<b>7</b>	<b>Equalização Cega Usando Predição Linear</b>	<b>74</b>
7.1	Introdução . . . . .	74
7.2	Predição Linear no Caso Monousuário . . . . .	75
7.3	MUD Cega Baseada em Kalman . . . . .	76
7.4	MUD Cega com ISI Baseada em Kalman Usando Predição . . . . .	78
7.5	Versão LMS da Atualização do Detector . . . . .	82
7.6	Conclusões . . . . .	82
<b>8</b>	<b>Análise Comparativa</b>	<b>88</b>
8.1	Introdução . . . . .	88
8.2	Visão Geral dos Algoritmos . . . . .	89
8.3	Funcionalidades Básicas . . . . .	91
8.4	Simulações . . . . .	92
8.4.1	Simulações Usando Transmissão Síncrona e Canais com Memória	94
8.4.2	Simulações de Canais sem Memória . . . . .	102
8.4.3	Simulação de Recepção de Sinais Assíncronos . . . . .	107
8.5	Conclusões . . . . .	111
<b>9</b>	<b>Implementações com Atualização Seletiva</b>	<b>114</b>
9.1	Introdução . . . . .	114

9.2	Filtragem <i>Set-Membership</i> . . . . .	115
9.3	Método de Subespaço Usando SM . . . . .	116
9.4	Método CMV Usando SM . . . . .	116
9.5	Método de Predição Linear usando SM . . . . .	117
9.6	Resultados de Simulações com SM . . . . .	117
9.6.1	Resultados para o Subespaço . . . . .	117
9.6.2	Resultados Usando CMV . . . . .	120
9.6.3	Resultados usando Predição Linear . . . . .	121
9.7	Conclusões . . . . .	125
<b>10</b>	<b>Implementações com Reuso de Dados</b>	<b>128</b>
10.1	Introdução . . . . .	128
10.2	Algoritmo de Projeção Afim . . . . .	129
10.3	Algoritmo CMV com Atualizações Feitas por APA . . . . .	130
10.4	SM com Atualizações por APA . . . . .	131
10.5	Algoritmo CMV com Atualizações por SMAP . . . . .	133
10.6	Algoritmo de Predição Linear numa Estrutura GSC . . . . .	133
10.7	Algoritmo de Predição Linear com GSC Usando APA . . . . .	135
10.8	Algoritmo de Predição Linear com Implementação GSC e Atualizações SMAP . . . . .	137
10.9	Simulações dos Algoritmos Apresentados . . . . .	138
10.9.1	Resultados para CMV com Atualizações APA . . . . .	138
10.9.2	Resultados para CMV com Atualizações por SMAP . . . . .	139

10.9.3 Resultados para a Implementação GSC do Algoritmo de Predição Linear . . . . .	143
10.9.4 Resultados para o Algoritmo de Predição Linear com GSC-SMAP . . . . .	146
10.10 Conclusões . . . . .	146
<b>11 Conclusões</b>	<b>155</b>
11.1 Sobre a Equalização Cega . . . . .	155
11.2 Sobre os Algoritmos Atuais . . . . .	157
11.3 Sobre Atualização Seletiva e Reuso de Dados . . . . .	158
11.4 Contribuições da Tese . . . . .	160
11.5 Propostas de Trabalhos Futuros . . . . .	161
<b>Referências Bibliográficas</b>	<b>163</b>

# Lista de Figuras

1.1	Sistema de comunicação sem fio. . . . .	2
2.1	Representação no domínio da frequência do efeito de um canal cuja resposta em frequência não é plana. No eixo vertical tem-se a resposta em magnitude e no horizontal a frequência normalizada. . . . .	8
2.2	Esquema de funcionamento do DFE. . . . .	17
2.3	Exemplo de implementação de equalizador utilizando LMS. . . . .	21
2.4	Estrutura básica de um MLSE adaptativo. . . . .	22
3.1	Sistema SIMO equivalente ao sistema SISO superamostrado. . . . .	32
4.1	<i>Simple Shift Register Generator</i> . . . . .	39
4.2	Sistema CDMA com MUD. . . . .	44
4.3	Modelo de CDMA implementado. . . . .	48
5.1	Esquema de funcionamento do algoritmo de subespaço. . . . .	59
6.1	Esquema de funcionamento do GSC. . . . .	67
6.2	Esquema de funcionamento do algoritmo de equalização cega por minimização da variância de saída. . . . .	70
7.1	Modelo de sistema SIMO. . . . .	75

7.2	Modelo GSC para o sistema sem ISI. . . . .	77
7.3	Esquema de funcionamento do algoritmo de equalização cega por predição linear baseado em Kalman. . . . .	81
8.1	Zeros dos canais das equações (8.1) até (8.5). Como os coeficientes dos canais são reais, os zeros são pares conjugados. Com isso, só se representa o semi-plano $Z$ superior, pois o semi-plano inferior é simétrico. . . . .	98
8.2	Comparação de desempenho dos algoritmos de equalização cega com 1 usuário no sistema. . . . .	100
8.3	Comparação de desempenho dos algoritmos de equalização cega com 3 usuários no sistema. . . . .	101
8.4	Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no sistema. . . . .	102
8.5	Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no sistema para canais sem memória. . . . .	107
8.6	Comparação de desempenho dos algoritmos de equalização cega com 2 usuários no caso assíncrono. . . . .	112
8.7	Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no caso assíncrono. . . . .	113
9.1	SER e taxa de atualização para $\text{SNR} = 0$ dB para o algoritmo de subespaço. . . . .	120
9.2	SER e taxa de atualização para $\text{SNR} = 5$ dB para o algoritmo de subespaço. . . . .	121
9.3	SER e taxa de atualização para $\text{SNR} = 0$ dB para o algoritmo CMV.	123
9.4	SER e taxa de atualização para $\text{SNR} = 5$ dB para o algoritmo CMV.	125

9.5	SER e taxa de atualização para SNR = 0 dB para o algoritmo de predição linear. . . . .	126
9.6	SER e taxa de atualização para SNR = 5 dB para o algoritmo de predição linear. . . . .	127
10.1	Esquema do algoritmo de predição linear usando GSC. . . . .	136
10.2	Comparação da convergência de coeficientes do filtro adaptativo na 1ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 5 dB. . . . .	140
10.3	Comparação da convergência de coeficientes do filtro adaptativo na 2ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 5 dB. . . . .	140
10.4	SER e taxa de atualização para SNR = 0dB para o algoritmo CMV.	141
10.5	SER e taxa de atualização para SNR = 5dB para o algoritmo CMV.	143
10.6	Comparação da convergência de coeficientes do filtro adaptativo na 1ª sub-banda do algoritmo CMV: (a) com atualização APA; (b) com atualização SMAP para $\gamma = 0, 2$ ; (c) com atualização SMAP para $\gamma = 0, 35$ ; ambos para SNR = 5 dB. . . . .	144
10.7	Comparação da convergência de coeficientes do filtro adaptativo na 2ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização SMAP para $\gamma = 0, 2$ ; (c) com atualização SMAP para $\gamma = 0, 35$ ; ambos para SNR = 5 dB. . . . .	145
10.8	Comparação da convergência de coeficientes do filtro adaptativo no algoritmo de Predição Linear, com implementação GSC e: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 0 dB.	148
10.9	SER e taxa de atualização para SNR = -5dB para o algoritmo de predição linear - GSC-SMAP. . . . .	150

10.10	SER e taxa de atualização para $SNR = 0\text{dB}$ para o algoritmo de predição linear - GSC-SMAP. . . . .	151
10.11	Comparação da convergência de coeficientes do filtro adaptativo no algoritmo de Predição Linear, com implementação GSC e: (a) com atualização LMS; (b) com atualização SMAP ( $\gamma = 0,45$ ); ambos para $SNR = 0\text{ dB}$ . . . . .	152

# Lista de Tabelas

8.1	Comparação das funcionalidades básicas dos três algoritmos analisados. . . . .	92
8.2	Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais com memória - Equação (5.43 - Modelagem do Artigo.) . . . . .	95
8.3	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais com memória - Modelagem do Artigo. . . . .	95
8.4	Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gunawan - Canais com memória. . . . .	96
8.5	Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais com Memória - Equação (4.12) - Modelagem da Tese. . . . .	96
8.6	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais com memória - Modelagem da Tese. . . . .	97
8.7	Resultados para o Algoritmo de Predição Linear de Fan e Li - Canais com memória. . . . .	97
8.8	Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais sem Memória. . . . .	104
8.9	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais sem Memória - Modelagem do Artigo. . . . .	104
8.10	Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gunawan - Canais sem Memória. . . . .	105



8.11	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais sem Memória - Modelagem da Tese. . . . .	105
8.12	Resultados para o Algoritmo de Predição Linear de Fan e Li - Canais sem Memória. . . . .	106
8.13	Resultados para o Algoritmo de Subespaço de Wang e Poor - Assíncrono - Equação (5.43 - Modelagem do Artigo). . . . .	108
8.14	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Assíncrono - Modelagem do Artigo. . . . .	108
8.15	Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gunawan - Assíncrono. . . . .	109
8.16	Resultados para o Algoritmo de Subespaço de Wang e Poor - Assíncrono - Equação (4.12 - Modelagem da Tese). . . . .	109
8.17	Resultados para o Algoritmo CMV de Xu e Tsatsanis - Assíncrono - Modelagem da Tese. . . . .	110
8.18	Resultados para o Algoritmo de Predição Linear de Fan e Li - Assíncrono.	110
9.1	Resultados para o Algoritmo de Subespaço de Wang e Poor - SM - Modelagem do Artigo. . . . .	119
9.2	Resultados para o Algoritmo de Subespaço de Wang e Poor - SM - Modelagem do Artigo. . . . .	119
9.3	Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM. . . . .	122
9.4	Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM. . . . .	122
9.5	Resultados para o Algoritmo de Predição Linear de Fan e Li - SM. . . . .	124
9.6	Resultados para o Algoritmo de Predição Linear de Fan e Li - SM. . . . .	124
10.1	Resultados para o Algoritmo CMV de Xu e Tsatsanis - APA. . . . .	139

10.2	Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM. . . . .	142
10.3	Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM. . . . .	142
10.4	Resultados para o Algoritmo de Predição Linear - GSC. . . . .	147
10.5	Resultados para o Algoritmo de Predição Linear - GSC - APA. . . . .	147
10.6	Resultados para o Algoritmo de Predição Linear - GSC - SMAP. . . . .	148
10.7	Resultados para o Algoritmo de Predição Linear - GSC - SMAP. . . . .	149
11.1	Resumo da comparação entre os algoritmos. . . . .	158

# Lista de Algoritmos

5.1	Algoritmo de Equalização Cega por Subespaço . . . . .	62
6.1	Algoritmo de Equalização Cega por Variância Mínima de Saída . . .	73
7.1	Algoritmo de Equalização Cega por Predição Linear, usando Kalman	84
7.2	Algoritmo de Equalização Cega por Predição Linear, com Atua- lizações LMS . . . . .	86
10.1	Algoritmo CMV com Atualizações por APA . . . . .	151
10.2	Algoritmo de Equalização Cega por Predição Linear, estrutura GSC, com Atualizações APA . . . . .	153

# Capítulo 1

## Introdução

### 1.1 Motivações

Para fins didáticos, um sistema de comunicações é tradicionalmente dividido em três partes: transmissor, canal e receptor. Essa divisão do sistema, apesar de simples, carrega muito do espírito que está presente em um projeto de engenharia de comunicações. Enquanto o transmissor e o receptor são projetados a fim de que informações de voz, imagens, vídeos ou de outra natureza possam ser compartilhados, o canal é visto como um obstáculo natural a esse compartilhamento.

Na verdade, o canal é o meio físico que efetivamente possibilita a comunicação. Nesse meio, no entanto, estão presentes outros elementos que interferem no sinal eletromagnético que se deseja transmitir. Alguns desses interferidores, que recebem a denominação genérica de ruído, têm um comportamento essencialmente estocástico. A existência de ruídos tem grande influência na maneira como é projetado o objeto de estudo desta tese, o equalizador. Um diagrama de um sistema de comunicação, bem como o modo como o equalizador participa desse esquema podem ser vistos na figura 1.1.

Num sistema de comunicações multiusuário, outras formas de onda que não são ruídos de ambiente também podem interferir no sinal que se deseja detectar no

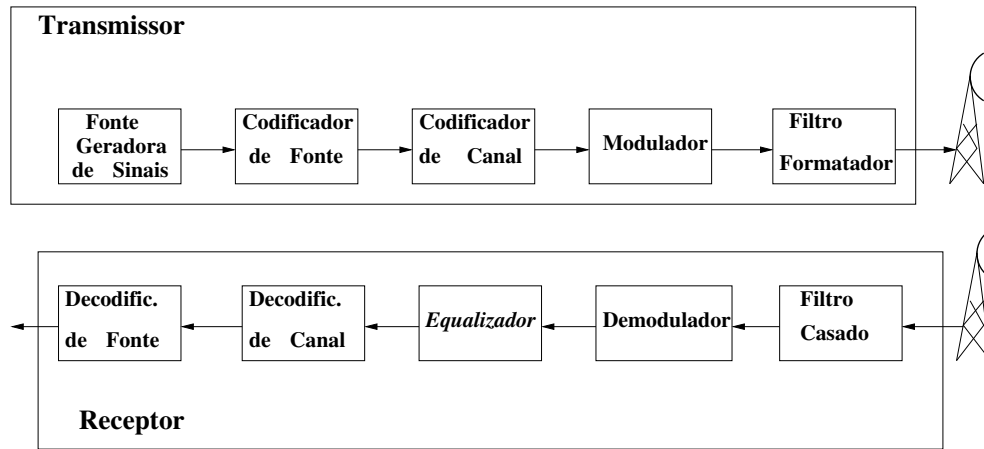


Figura 1.1: Sistema de comunicação sem fio.

receptor. Esses sinais são referentes aos outros usuários do sistema, cujas formas de onda por vezes se sobrepõem ao sinal do usuário desejado tanto no tempo quanto na frequência, como no caso do CDMA (*Code Division Multiple Access*).

Outra forma de distorção no sinal transmitido é gerada muitas vezes pela presença de múltiplos percursos na propagação do sinal. Como os percursos não têm necessariamente o mesmo comprimento, por vezes amostras já transmitidas influenciam na recepção da amostra atual do sinal. Essa interferência é conhecida como ISI (*Inter-Symbol Interference*), sendo também resultado da banda finita do canal utilizado. O papel principal do equalizador é exatamente o de anular o efeito causado pela ISI [1].

Quando não se tem conhecimento da resposta ao impulso do canal, normalmente se utiliza um equalizador adaptativo, no qual os coeficientes do equalizador são modificados durante o seu funcionamento de maneira a minimizar o erro entre sua saída e a sequência transmitida. Isso é feito tradicionalmente através do envio de uma sequência conhecida pelo receptor, denominada sequência de treinamento. A saída do equalizador é então comparada com essa sequência gerando mudanças nos coeficientes do equalizador. Os equalizadores que não utilizam sequência de treinamento na atualização de seus coeficientes são conhecidos como equalizadores cegos.

Nesta tese, são analisados algoritmos para realizar equalização cega em ambientes ruidosos com vários usuários. Conseqüentemente, esses algoritmos procuram lidar com os três obstáculos apresentados para comunicação à distância: ruído, presença de outros usuários e interferência intersimbólica.

No caso particular de sistemas multiusuários, a presença da interferência intersimbólica pode resultar em problemas de diferentes níveis de gravidade de acordo com a técnica de múltiplos acessos utilizada. No caso do TDMA (*Time Division Multiple Access*), por exemplo, a presença de ISI é muito grave, na medida em que o espalhamento dos símbolos no tempo resulta em interferência entre os usuários, pois os usuários são multiplexados no tempo. O CDMA, por sua vez, apresenta um certo grau de resistência ao problema de ISI pelo espalhamento espectral e pelo uso de códigos de espalhamento (como será mostrado nas seções 4.2 e 4.3) [1].

A maior resistência do CDMA ao problema de ISI poderia significar que um estudo de sistemas de equalização para multiusuários deveria dar ênfase ao estudo de ambientes que utilizassem outras técnicas de compartilhamento de acesso como TDMA e FDMA (*Frequency Division Multiple Access*). No entanto, o CDMA é o método de compartilhamento de acesso utilizado pelos principais padrões de telefonia celular de 3<sup>a</sup> geração, sendo o mais cotado também para fazer parte dos sistemas de 4<sup>a</sup> geração. Por isso, é de grande importância o estudo de uma melhoria no desempenho do CDMA. Essa é a principal razão da escolha do CDMA como o ambiente de comunicação a ser utilizado nesta tese.

Nos sistemas CDMA, o termo equalização é em várias situações contestado por se referir apenas à eliminação da interferência intersimbólica, deixando de lado o aspecto maior da eliminação de interferências referentes ao múltiplo acesso, denominadas geralmente de MAI (*Multiple Access Interference*). Os termos equalizador e equalização, no entanto, serão mantidos no decorrer da tese por traduzirem bem o enfoque proposto pela tese e por serem mais apropriados à visão de processamento de sinais que permeia este trabalho. O título da tese, no entanto, tem um sentido mais amplo e deve servir de porta de entrada para o trabalho para pesquisadores de várias áreas. Por isso, o termo supressão de interferências, cujo significado é

de melhor entendimento para profissionais de comunicações em geral, foi usado no título.

## 1.2 A Tese

No capítulo 2 são apresentados os principais aspectos referentes à equalização não-cega, tanto no caso *batch* quanto adaptativo. Os coeficientes do equalizador no caso *batch* são os mesmos durante a recepção de sinais, enquanto no caso adaptativo os valores dos coeficientes são atualizados de acordo com o sinal recebido.

O capítulo 3 mostra a evolução histórica dos algoritmos de equalização cega, partindo de algoritmos que utilizam momentos estatísticos de ordem superior até chegar a métodos que usam estatísticas de segunda ordem em conjunto com ciclo-estacionaridade ou diversidade espacial.

O capítulo 4 apresenta os principais aspectos do ambiente multiusuário utilizado, com compartilhamento de acesso dos usuários via diversidade de código.

Os capítulos 5, 6 e 7 são expostas três famílias de equalizadores cegos, respectivamente as que usam subespaço, variância mínima e predição linear. Em particular, são estudados e implementados quatro algoritmos, representativos dessas três classes.

O capítulo 8 complementa os capítulos anteriores ao explorar resultados experimentais na comparação entre os algoritmos.

No capítulo 9 é apresentada uma implementação dos algoritmos apresentados nos capítulos 5 e 6 utilizando o conceito de filtragem *Set-Membership*, a fim de se diminuir a complexidade computacional dos algoritmos.

No capítulo 10 são descritas implementações dos métodos de variância mínima e predição linear utilizando o algoritmo de projeção afim, que é um algoritmo baseado em LMS com reuso de dados, o que garante uma convergência mais rápida dos

coeficientes do filtro adaptativo. A fim de corrigir um efeito colateral dos algoritmos de reuso de dados, mais especificamente o aumento do desajuste do sistema, também são implementadas variações do algoritmo de projeção afim com filtragem *Set-Membership*.

No capítulo 11 pode ser visto um balanço das principais conclusões da tese, analisando-se as contribuições do trabalho desenvolvido e elaborando propostas para futuros trabalhos na área.



# Capítulo 2

## Equalizadores

### 2.1 Introdução

A interferência entre os símbolos transmitidos causa uma grande perda de eficiência na detecção das formas de onda transmitidas. Esse efeito está presente em canais com memória, apresentando maior distorção à medida que aumenta a ordem do filtro que modela o canal.

Neste capítulo, será apresentada a teoria básica a respeito do funcionamento dos equalizadores, que são sistemas usados para se cancelar o efeito causado pelo canal na seqüência transmitida, a fim de se restaurar o sinal original. Nas seções 2.2 e 2.3 serão vistos alguns conceitos teóricos relativos aos efeitos de um canal com banda passante limitada sobre o sinal transmitido. Na seção 2.4 serão vistos os principais algoritmos *batch*, ou seja, não-adaptativos, para equalização. Uma comparação entre esses algoritmos pode ser vista na seção 2.5. A seção 2.6, por sua vez, mostra os algoritmos para equalização adaptativa, importantes quando não se conhece a resposta ao impulso do canal utilizado.

## 2.2 Canais ideais

Considerando que um canal com banda limitada  $W$  tenha a resposta em frequência dada por [1]:

$$C(f) = \begin{cases} |C(f)|e^{j\theta(f)}, & \text{se } f \leq W \\ 0, & \text{se } f > W \end{cases} \quad (2.1)$$

sendo a característica de atraso da envoltória dada por [1]:

$$\tau(f) = -\frac{1}{2\pi} \frac{d\theta(f)}{df} \quad (2.2)$$

o canal é dito ideal quando  $|C(f)|$  e  $\tau(f)$  são constantes para todo  $f \leq W$ . Sua resposta em frequência é dita plana. Nessa situação, o efeito do canal sobre um sinal de entrada também com banda limitada a  $W$ , é apenas um ganho, com um possível atraso gerado por distorções lineares de fase, efeito que pode ser facilmente compensado no receptor. No caso de  $|C(f)|$  não ser constante, diz-se que o sistema apresenta distorção de amplitude. Um canal com distorção de amplitude é representado na Figura 2.1(b). No caso de  $\tau(f)$  variar com a frequência, o sistema apresenta distorção de fase [1].

No caso do canal ideal, a resposta em frequência do equalizador deve ser uma constante para todas as frequências, compensando a atenuação dada pelo canal. Se o canal não for ideal, a implementação do equalizador torna-se bem mais complexa.

A Figura 2.1 ilustra bem esta questão. Considere um sinal de entrada cujo espectro é dado por (a). Ao se passar o sinal pelo canal cuja resposta em frequência está mostrada em (b), na saída do sistema, obtém-se o espectro representado em (c). Pelas distorções causadas pelo canal, pode-se dizer, intuitivamente, que é bem mais difícil se voltar ao espectro original do sinal que no caso do canal ideal. Essa percepção intuitiva é confirmada na realidade.

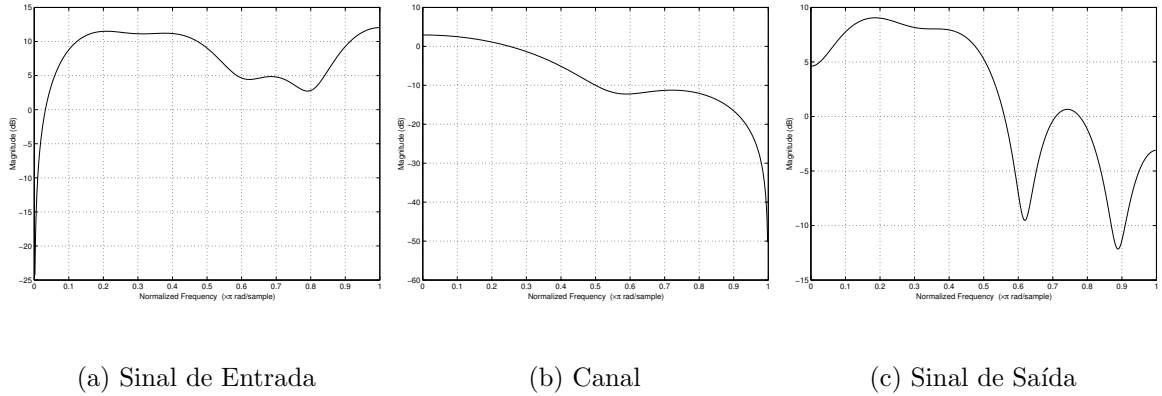


Figura 2.1: Representação no domínio da frequência do efeito de um canal cuja resposta em frequência não é plana. No eixo vertical tem-se a resposta em magnitude e no horizontal a frequência normalizada.

## 2.3 Interferência Intersimbólica

Uma outra forma de se visualizar a distorção provocada por um canal não-ideal pode ser feita no domínio do tempo. Como pode ser percebido pela equação (2.1), um canal que causa distorção de amplitude pode ser visto como um filtro, já que sua resposta varia de acordo com a frequência. A forma geral de um filtro digital com resposta ao impulso finita é dada por:

$$y(n) = a_0x(n) + a_1x(n-1) + \dots + a_{n-1}x(0) \quad (2.3)$$

Modelando-se o canal como um filtro FIR (*Finite Impulse Response*), percebe-se, pela equação (2.3), que diferentes amostras transmitidas interferem no símbolo atual recebido. Por isso, a distorção apresentada num canal não-ideal é conhecida como interferência intersimbólica (ISI). Outra forma de se verificar esse efeito é notando que, em geral, o canal tende a eliminar as componentes de alta frequência do sinal. Com isso, o espectro de frequência do sinal sofre uma redução. Conseqüentemente, no domínio do tempo ocorre uma dilatação dos símbolos transmitidos, que passam a interferir entre si.

Imagina-se que se tenha uma seqüência  $b(l)$  de símbolos para serem passados através de um canal contínuo com resposta em frequência  $C(f)$ , não nula

apenas para  $|f| \leq W$ . Assume-se que os símbolos sejam modulados por um pulso  $p(t)$  com resposta em frequência também restrita a uma banda  $W$ . Seja  $h(t)$ , a convolução contínua no tempo entre  $c(t)$  (resposta ao impulso contínua do canal) e  $p(t)$ . Considerando  $z(t)$  o ruído aditivo presente no canal, então o sinal recebido assume a seguinte forma [1]:

$$r(t) = \sum_{l=0}^{\infty} b(l)h(t - lT) + z(t) \quad (2.4)$$

Considera-se que, no receptor, este sinal seja filtrado (para fins de detecção) e depois amostrado a uma taxa  $1/T$ . Seja a resposta amostrada do filtro ao sinal  $h(t)$  dada por  $x(l)$  e a resposta amostrada do filtro ao ruído  $z(t)$  dada por  $n(l)$ , ao se normalizar o período para uma unidade de tempo, tem-se que [1]:

$$r(k) = \sum_{l=0}^{\infty} b(l)x(k - l) + n(k) \quad (2.5)$$

Normalizando-se  $x(0)$  para ser unitário, chega-se a uma expressão mais conveniente [1]:

$$r(k) = b(k) + \sum_{l \neq k} b(l)x(k - l) + n(k) \quad (2.6)$$

Nessa equação, percebe-se claramente a distorção provocada pela ISI na detecção do símbolo atual. O primeiro termo no lado direito da equação refere-se ao símbolo realmente transmitido. O segundo termo expressa o efeito de ISI, enquanto o terceiro é o ruído do sistema.

## 2.4 Algoritmos de Equalização

Nesta seção serão apresentados alguns algoritmos básicos de equalização, ou seja, que visam detectar a seqüência correta transmitida levando-se em conta as distorções do canal. O primeiro deles utiliza como critério a descoberta da seqüência transmitida que maximize uma figura de mérito muito importante em estatística: a razão de verossimilhança (*likelihood ratio*), recebendo por isso a denominação *maximum-likelihood sequence estimator* (MLSE). Ainda serão discutidos o equalizador linear e o DFE (*decision-feedback equalizer*), cujas concepções são baseadas

em estruturas básicas de filtros digitais. Outro tópico a ser discutido é o uso de equalizadores que funcionam a uma taxa de amostragem superior à taxa em que os símbolos foram transmitidos. Esses equalizadores recebem o nome de *fractionally spaced equalizers* (FSE).

Os algoritmos em geral apresentados aqui são colocados como obedecendo a critérios sub-ótimos de funcionamento. O critério ótimo é definido a partir da proposição dada a seguir.

Considera-se que uma seqüência  $\mathbf{b} = \{b(k)\}$  seja transmitida através de um canal com memória. Cada elemento dessa seqüência é escolhido dentre os  $M$  símbolos do alfabeto  $\{B_0, B_1, \dots, B_{M-1}\}$ . As possíveis seqüências de símbolos a serem transmitidas formam um conjunto muito grande. Por exemplo, se o alfabeto for binário  $\{0, 1\}$ , então, ao se transmitir dois símbolos, pode-se ter transmitido 00, 01, 10 ou 11, e tamanho desse conjunto cresce exponencialmente com o tamanho do alfabeto. A fim de designar apenas uma dessas possíveis seqüências, adota-se a notação  $\mathbf{b}_i$ .

Seja  $\mathbf{r} = \{r(k)\}$  a seqüência recebida na saída do canal. Então o critério ótimo de decisão é obtido a partir das probabilidades *a posteriori* [2]:

Decide-se que  $\mathbf{b}_i$  foi transmitida se:

$$p(\mathbf{b}_i|\mathbf{r}) > p(\mathbf{b}_j|\mathbf{r}), \forall i \neq j \quad (2.7)$$

### 2.4.1 MLSE - *Maximum Likelihood Sequence Estimator*

Considera-se de início uma distribuição probabilística qualquer  $f(x, \theta)$ . O problema de estimação colocado aqui consiste em descobrir o parâmetro  $\theta$  dessa distribuição com base na observação da variável aleatória  $x$ .

Utilizando a abordagem clássica da teoria da estimação,  $\theta$  é considerada uma constante desconhecida [3]. No entanto, esse valor não é por completo desconhecido. Ao se jogar para o alto uma moeda não viciada, por exemplo, espera-se que seja de 0,5 a chance de verificar-se que a parte de cima da moeda seja cara ao cair no

chão.

Uma nova abordagem foi trazida pelos estimadores bayesianos. O parâmetro  $\theta$  passa a ser visto como uma variável aleatória  $\theta_r$ . Já a distribuição de  $x$  é interpretada como  $f_x(x|\theta)$  quando  $\theta_r = \theta$ . As informações que já se possui sobre  $\theta$  são colocadas na distribuição  $f_\theta(\theta)$ . O problema de estimação de um parâmetro desconhecido é convertido em um problema de estimação do valor assumido por uma variável aleatória [3].

Pela teoria probabilística de Bayes [3]:

$$f_\theta(\theta|X) = \frac{f(X|\theta)}{f(X)} f_\theta(\theta) \quad (2.8)$$

Nessa equação,  $X = [x_1 \ x_2 \ \dots \ x_n]$  representa o vetor de valores observados. Considerando-se  $f(x|\theta)$  como a densidade de probabilidade da variável aleatória  $x$  dado que  $\theta_r = \theta$ . A função  $f(x|\theta)$  é chamada função de verossimilhança (*likelihood function*). A estimativa de  $\theta$  tal que  $f(x|\theta)$  é maximizada é chamada de estimativa de máxima verossimilhança [3] [4]. Por sua vez, o estimador que fizer uma estimativa de  $\theta$  baseada em  $f(x|\theta)$  é conhecido como estimador de máxima verossimilhança. Uma outra métrica bastante usada é a função de log-verossimilhança (*log-likelihood function*), definida como o logaritmo neperiano de  $f(x|\theta)$ .

Considera-se agora um sistema em que uma seqüência  $\{b(k)\}$  gerada por uma fonte qualquer passe por um canal  $h(k)$  de acordo com a seguinte equação:

$$y(k) = \sum_{l=-\infty}^{\infty} b(l)h(k-l) \quad (2.9)$$

À seqüência de saída do canal  $\{y(k)\}$  é acrescentado um ruído

$$r(k) = y(k) + n(k) \quad (2.10)$$

Considerando-se que  $h(k)$  seja uma seqüência de tamanho  $P$  finito, sua modelagem pode ser feita através de um *tapped-delay line*. Com isso, o canal pode ser visto como um sistema com  $M^{(P-1)}$  estados possíveis [5]. Define-se uma seqüência de estados  $\{s(k)\}$  contendo  $(P-1)$  amostras de  $\{b(k)\}$ .

Pode-se colocar o critério de máxima verossimilhança como a escolha da seqüência  $\{\hat{b}(k)\}$ , estimativa dos símbolos transmitidos, que maximize a seguinte métrica:

$$L[\hat{b}(k)] = p[r(k)|\hat{b}(k)] \quad (2.11)$$

Nessa equação,  $p[r(k)|\hat{b}(k)]$  representa a função densidade de probabilidade associada à observação  $r(k)$   $\hat{b}(k)$  é a seqüência transmitida.

Tanto  $\{b(k)\}$  como  $\{y(k)\}$  e  $\{s(k)\}$  são processos estocásticos. Dado que o canal  $h(k)$  é determinístico, o mapeamento entre as realizações de  $\{b(k)\}$  e  $\{y(k)\}$  é feito biunivocamente. O mesmo pode ser dito entre  $\{b(k)\}$  e  $\{s(k)\}$  [5]. Com isso, o critério de máxima verossimilhança, dado pela equação (2.11), pode ser reescrito da seguinte forma:

$$ML[\hat{b}(k)] = ML[\hat{s}(k)] = \max p[r(k)|\hat{s}(k)] \quad (2.12)$$

onde o circunflexo denota o valor estimado. No artigo [5], é descrita uma implementação para o critério de máxima verossimilhança baseada na detecção do conjunto de estados que satisfaçam (2.12). Isso é feito através de um filtro de branqueamento colocado em série a uma estrutura em treliça correspondente ao algoritmo de Viterbi (comumente usado também em outras aplicações como codificação de canal) [5]. Essa implementação tem a grande vantagem de ter complexidade computacional proporcional a  $m^K$ , onde  $m$  é o tamanho do alfabeto usado pela fonte e  $K$  é o comprimento do canal [5]. Isso é um grande avanço em relação às primeiras implementações do equalizador ótimo, cuja complexidade computacional crescia exponencialmente em relação ao tamanho da seqüência de entrada [5].

### 2.4.2 Equalizador Linear *Zero-Forcing*

A solução empregada nesta seção para o problema da equalização baseia-se no uso de um filtro linear com coeficientes  $\{w(j)\}$ , a partir de uma seqüência recebida  $\{r(j)\}$ , a fim de se obter  $\{\hat{b}(k)\}$ , a estimativa dos símbolos transmitidos

[1], da forma:

$$\hat{b}(k) = \sum_{j=-K}^K w(j)r(k-j) \quad (2.13)$$

Nesta subseção, inicialmente será explicado o critério de distorção de pico e depois será dada uma visão sobre o critério MSE (*mean-square error*) para se determinar os valores ótimos de  $\{w(j)\}$ .

Assume-se de início que o equalizador tenha um número infinito de coeficientes. Define-se uma seqüência  $\{q(l)\}$  formada pela convolução linear entre  $\{w(l)\}$  e  $\{h(l)\}$ . A saída do equalizador no  $k$ -ésimo instante é [1]:

$$\hat{b}(k) = q(0)b(k) + \sum_{l \neq k} b(l)q(l-k) + \sum_{j=-\infty}^{\infty} w(j)\eta(k-j) \quad (2.14)$$

O primeiro termo no lado direito dessa equação representa uma versão escalada do símbolo que se deseja detectar. O segundo termo representa a ISI, enquanto o terceiro está relacionado ao ruído. O valor máximo da ISI é conhecido como distorção de pico, sendo expresso por [1]:

$$D(\mathbf{w}) = \sum_{l=-\infty, l \neq 0}^{\infty} |q(l)| = \sum_{l=-\infty, l \neq 0}^{\infty} \left| \sum_{j=-\infty}^{\infty} w(j)h(l-j) \right| \quad (2.15)$$

A condição para eliminação de ISI é dada por:

$$q(l) = \sum_{j=-\infty}^{\infty} w(j)h(l-j) = \begin{cases} \xi, & l = 0 \\ 0, & l \neq 0 \end{cases} \quad (2.16)$$

onde  $\xi$  é uma constante qualquer. Fazendo-se  $\xi = 1$ , sem perda de generalidade, e achando-se a transformada Z da equação acima chega-se a:

$$Q(z) = W(z)H(z) = 1 \Rightarrow W(z) = \frac{1}{H(z)} \quad (2.17)$$

A resposta do equalizador no plano Z deve ser a inversa da resposta do canal, ou seja, anulam-se as distorções geradas pelo canal apenas colocando-o em série com um sistema que tenha o inverso de sua função de transferência, como era de se esperar. A equação (2.17) define o equalizador chamado *zero-forcing*.

Um dos problemas encontrados pelo equalizador *zero-forcing* é a amplificação do ruído nas frequências atenuadas pelo canal, ou seja, nas frequências que são mais amplificadas pelo equalizador.



Em certas situações, o equalizador deve funcionar em um ambiente contendo ruído colorido, ou seja, que apresenta correlação entre suas amostras. É colocado no receptor um filtro de branqueamento antes do equalizador, a fim de se facilitar, através de filtro casado, a detecção da forma de onda transmitida. A função de transferência do filtro de branqueamento tem a forma  $H^*(1/z)$ . Nesse caso, pode-se definir um equalizador equivalente englobando-se o que está definido na equação (2.17) e mais o branqueador, chegando-se a:

$$W'(z) = \frac{1}{H(z)H^*(1/z)} = \frac{1}{X(z)} \quad (2.18)$$

Analisa-se agora o caso de o equalizador ter tamanho finito  $(2K + 1)$ . Como  $w(j) = 0$  para  $|j| > K$ , então a convolução de  $h(l)$  e  $w(l)$  é zero para  $l < -K$  ou  $l > (K + L - 1)$ . Neste caso a métrica de distorção é dada por:

$$D(\mathbf{c}) = \sum_{l=-K, l \neq 0}^{K+L-1} |q(n)| = \sum_{l=-K, l \neq 0}^{K+L-1} \left| \sum_j w(j)h(l-j) \right| \quad (2.19)$$

Embora o equalizador tenha  $(2K + 1)$  parâmetros a serem ajustados, há  $(2K + L)$  valores não-nulos na resposta  $q(l)$ . Então, é quase sempre impossível se eliminar completamente a ISI no equalizador. Essa função de distorção tem um mínimo global e nenhum mínimo local [1]. Sua minimização pode ser feita utilizando-se algoritmos adaptativos.

### 2.4.3 Equalizador Linear MMSE

No caso de se usar a métrica MSE, procura-se minimizar a média do erro quadrático de estimação da seqüência  $b(k)$ , ou seja, a performance é medida de acordo com a métrica  $J = E[|b(k) - \hat{b}(k)|^2]$  [1].

A partir da teoria de filtros de Wiener, a solução para minimização do MSE pode ser obtida a partir da ortogonalização do erro em relação aos coeficientes da seqüência do sinal. Com isso, em presença de ruído branco gaussiano e aditivo,

com densidade espectral de potência  $N_0/2$ :

$$E \left[ \left( b(k) - \sum_{j=-\infty}^{\infty} w(j)h(k-j) \right) r^*(k-l) \right] = 0 \Rightarrow$$

$$\sum_{j=-\infty}^{\infty} w(j)E[r(k-j)r^*(k-l)] = E[b(k)r^*(k-l)] \quad (2.20)$$

onde  $h(k)$  representa o filtro que modela o canal e:

$$E[r(k-j)r^*(k-l)] = \sum_{m=0}^L h^*(m)h(m+l-j) + N_0\delta(l,j)$$

$$= \begin{cases} c(l-j) + N_0\delta(l,j), & |l-j| \leq L \\ 0, & |l-j| > L \end{cases} \quad (2.21)$$

$$E[b(k)r^*(k-l)] = \begin{cases} f^*(-l), & -L \leq l \leq 0 \\ 0, & \text{demais } l \end{cases} \quad (2.22)$$

Usando-se as equações (2.20), (2.21) e (2.22) e tirando-se a transformada Z dos termos, chega-se à expressão do filtro MMSE (*minimum mean-square error*) [1], tal que

$$W(z)[H(z)H^*(z^{-1}) + N_0] = H^*(z^{-1}) \Rightarrow W(z) = \frac{H^*(z^{-1})}{H(z)H^*(z^{-1}) + N_0} \quad (2.23)$$

Do mesmo modo que no caso do equalizador *zero-forcing* para o ruído colorido, pode-se incluir o filtro de branqueamento na expressão do equalizador, fazendo-se

$$W'(z) = \frac{1}{H(z)H^*(z^{-1}) + N_0} \quad (2.24)$$

Percebe-se que, para um valor pequeno de  $N_0$ , potência média do ruído, a expressão da equação (2.24) aproxima-se bastante da equação (2.18), que utilizou outra métrica na minimização dos coeficientes. No entanto, para um valor de  $N_0$  diferente de zero, tem-se um valor residual de ISI para a métrica MSE. Esse valor é dado por

$$J_{\min} = 1 - \sum_{j=-\infty}^{\infty} w(j)h(-j) \quad (2.25)$$

Mais uma vez estende-se o resultado para o caso de um equalizador finito com tamanho  $(2L + 1)$ . Denotando-se a métrica por  $J(K)$ , chega-se à expressão [1]

$$J(K) = E \left[ \left| b(k) - \sum_{j=-K}^K w(j)v(k-j) \right|^2 \right] \quad (2.26)$$

Definindo-se

$$A_{lj} = \begin{cases} r(l-j) + N_0\delta(l,j), & |l-j| \leq L \\ 0, & |l-j| > L \end{cases} \quad (2.27)$$

$$e_l = \begin{cases} h^*(-l), & -L \leq l \leq 0 \\ 0, & \text{demais } l \end{cases} \quad (2.28)$$

a ortogonalização do erro em relação aos sinais  $v^*(k-j)$  leva à equação [1]

$$\sum_{j=-K}^K w_j A_{lj} = e_l \quad (2.29)$$

que pode ser colocada na forma matricial, definindo-se:

$$\mathbf{A}\mathbf{w}_{\text{opt}} = \mathbf{e} \Rightarrow \mathbf{w}_{\text{opt}} = \mathbf{A}^{-1}\mathbf{e} \quad (2.30)$$

Assim como no caso do equalizador com número infinito de coeficientes, não se consegue anular o valor de  $J(K)$ . O valor mínimo ( $J_{\min}$ ) é dado por:

$$J_{\min}(K) = 1 - \mathbf{e}^H \mathbf{A}^{-1} \mathbf{e} \quad (2.31)$$

#### 2.4.4 DFE - *Decision Feedback Equalizer*

A idéia central na concepção do equalizador por realimentação de decisão (DFE) é se aproveitar das decisões sobre os símbolos já detectados e retirar a parte da ISI devida a eles na detecção do símbolo atual. Seu funcionamento é baseado em dois filtros. O filtro *feedforward* tem função idêntica aos filtros usados nas subseções anteriores. Já o filtro *feedback* melhora o desempenho do sistema ao retirar o efeito de ISI devido aos símbolos já detectados. O funcionamento do sistema pode ser visto na figura 2.2 [1].

A seguir serão formalizados alguns resultados sobre os coeficientes dos filtros *feedback* e *feedforward* do DFE para o caso em que os filtros tenham número finito de coeficientes.

Considera-se nesse caso que haja  $(K_1 + 1)$  coeficientes na parte *feedforward* do DFE e  $K_2$  na parte *feedback*, de maneira que os coeficientes  $\{w_k\}$  com  $-K_1 \leq$

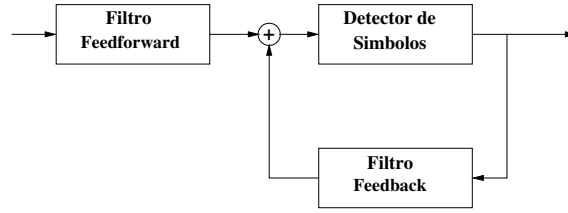


Figura 2.2: Esquema de funcionamento do DFE.

$k \leq 0$  correspondam à parte *feedforward*, enquanto os coeficientes  $\{w_k\}$  com  $0 < k \leq K_2$  estão relacionados à parte *feedback*. Considera-se ainda que a seqüência  $\{\tilde{b}(k-1), \dots, \tilde{b}(k-K_2)\}$  corresponda aos símbolos já detectados, enquanto  $\hat{b}(k)$  é o símbolo que se deseja detectar. A relação desses parâmetros pode ser melhor expressada através da seguinte equação [1]:

$$\hat{b}(k) = \sum_{j=-K_1}^0 w(j)r(k+j) + \sum_{j=1}^{K_2} w(j)\tilde{b}(k-j) \quad (2.32)$$

Utilizando-se o critério MSE ( $J = E[|b(k) - \hat{b}(k)|^2]$ ), essa métrica leva a uma série de equações lineares para se achar os coeficientes do filtro *feedforward* [1]:

$$\sum_{j=-K_1}^0 \psi_{lj}w(j) = h^*(-l) \quad l = -K_1, \dots, -1, 0 \quad (2.33)$$

onde

$$\psi_{lj} = \sum_{m=0}^{-l} h^*(m)h(m+l-j) + N_0\delta(l, j) \quad l, j = -K_1, \dots, -1, 0 \quad (2.34)$$

sendo  $N_0$  a variância do ruído.

Considerando que o filtro de *feedforward* tenha um número infinito de coeficientes, há um MSE residual no sistema dada por [1]:

$$J_{\min} = \exp \left\{ \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \ln \left[ \frac{N_0}{X(e^{j\omega T}) + N_0} \right] d\omega \right\} \quad (2.35)$$

onde  $\exp\{\cdot\}$  é a função exponencial e  $X(e^{j\omega T})$  é o espectro do sinal na saída do filtro casado. De modo geral, há um considerável ganho de performance num equalizador ao se eliminar os efeitos de ISI de termos já detectados através do DFE.

Há, no entanto, uma degradação significativa na performance do DFE devido à ISI residual, ou seja, devido à realimentação de detecções erradas, especialmente em canais com distorção severa [1]. Em comparação com o filtro linear tradicional, o DFE apresenta um desempenho bem melhor para canais com severas distorções, principalmente no caso de haver nulos no espectro ou de o canal não ter fase mínima [6].

### 2.4.5 FSE - *Fractionally Spaced Equalizer*

Nos equalizadores utilizados anteriormente, os multiplicadores dos filtros estavam espaçados por um período de símbolo. Isso torna o desempenho de todo o demodulador bastante dependente de uma perfeita sincronização com a portadora, em relação à frequência.

O equalizador fracionalmente espaçado (FSE - *Fractionally Spaced Equalizer*) tem seu funcionamento baseado numa amostragem do sinal com uma taxa equivalente a, no mínimo, o dobro da taxa de transmissão dos símbolos.

Uma das grandes vantagens do FSE em relação aos equalizadores lineares tradicionais é a robustez quanto a erros de sincronização com a portadora [7]. Um erro de sincronização da portadora pode gerar componentes de frequência maior que o permitido pelo teorema de Nyquist [7] para um sistema que funciona na taxa de símbolos, mas que não gera problemas para um FSE.

No caso do equalizador por realimentação de decisão, visto na subseção 2.4.4, a otimalidade da estrutura é baseada no fato do filtro de detecção no receptor estar perfeitamente casado com a resposta do canal com distorção. A fim de se eliminar esta dependência, é comum se usar um filtro *feedforward* fracionalmente espaçado. O filtro *feedback*, no entanto, continua espaçado pelo período de símbolo [1].

# 2.5 Comparação dos Equalizadores

De todos os equalizadores vistos, o que atinge o melhor desempenho quanto à minimização dos erros de detecção é o MLSE. Sua principal desvantagem, como visto na subseção 2.4.1, é a complexidade computacional do sistema.

Os algoritmos de equalização linear e DFE enfrentam ISI residuais na implementação, por terem um número finito de coeficientes. Algumas implementações alternativas do DFE, como o uso de um preditor linear como filtro de *feedback*, melhoram o desempenho do sistema.

Algumas tentativas foram feitas no sentido de diminuir a complexidade computacional do MLSE. Muitas dessas implementações foram feitas obrigando o sinal a passar por um equalizador linear ou DFE antes da equalização ser feita efetivamente por um MLSE. Com isso, o primeiro equalizador apenas diminui o efeito da ISI, de maneira a suavizar o trabalho do MLSE, que passaria a exigir menos complexidade computacional.

Os algoritmos utilizados até o momento são soluções para problema de equalização quando o canal é conhecido e invariante no tempo. No entanto, em muitas situações, é necessário que os coeficientes do equalizador se ajustem ao canal, cuja resposta não é conhecida ou tenham que ser ajustados a uma nova resposta em frequência de um canal variante no tempo. Nestas situações, o mais indicado é a utilização de algoritmos adaptativos, que serão descritos na próxima seção.

# 2.6 Equalização Adaptativa

Nesta seção serão tratados equalizadores cujos coeficientes variam no sentido de minimizar o erro de detecção do sistema. Nos algoritmos descritos aqui, os coeficientes do equalizador são treinados utilizando-se uma seqüência de símbolos previamente determinada e, após o período de treinamento, o sistema fica em funcionamento no chamado modo direcionado por decisão (*decision-directed*). Neste

modo, os coeficientes do equalizador são atualizados com base apenas na decisão sobre o último símbolo transmitido.

### 2.6.1 *Zero-Forcing* Adaptativo

Conforme visto na subseção 2.4.2, o algoritmo de *zero-forcing* para equalizadores com resposta ao impulso finita (FIR) tem solução não-tendenciosa para o caso de a distorção de pico da entrada ser menor que um.

Considera-se que seja transmitida a seqüência  $\{b(k)\}$  de símbolos de informação. Para símbolos de informação descorrelacionados,  $E[b(k)b^*(j)] = \delta(k, j)$ , ou seja, a correlação só existe para  $k = j$ . Supõe-se ainda que o ruído seja descorrelacionado com os símbolos gerados. Nesse caso, sendo  $e(k)$  o erro na estimação da seqüência transmitida, tem-se, a partir de (2.14), que:

$$E[e(k)b^*(k-j)] = \delta(j, 0) - q(j) \quad (2.36)$$

onde  $q(j)$  representa a resposta ao impulso da combinação com o equalizador. Para as condições de cancelamento total de ISI, ou seja,  $E[e(k)b^*(k-j)] = 0$ , devemos ter  $q(0) = 1$  e os demais  $q(k)$  devem ser nulos. Considerando-se que a métrica a ser minimizada é o *zero-forcing*, um algoritmo bastante simples de atualização é enunciado em [1]:

$$w_j(k+1) = w_j(k) + \mu e(k)b^*(k-j) \quad (2.37)$$

Nessa equação, o parâmetro  $\mu$  representa o passo de atualização, enquanto  $e(k)b^*(k-j)$  é uma estimativa da correlação cruzada entre o erro instantâneo de estimação  $e(k)$  e o símbolo transmitido.

### 2.6.2 Implementações Usando LMS

Considerando uma função de  $n$  variáveis (uma superfície no espaço  $(n+1)$ -dimensional) sem mínimos locais, pode-se partir de qualquer ponto desta superfície e chegar ao mínimo global seguindo-se, a cada instante, a direção do gradiente, mas

## 2.6. EQUALIZAÇÃO ADAPTATIVA

no sentido contrário a ele. O mínimo global corresponderia à chamada solução de Wiener do problema. Um algoritmo simples que utiliza essa idéia para a minimização do erro médio quadrático do sistema é o LMS (*Least Mean Squares*). Sua expressão geral é dada pela fórmula [8]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\mathbf{x}(k)e(k) \quad (2.38)$$

No caso,  $\mathbf{x}(k)$  é a seqüência de entrada,  $e(k)$  é o erro entre a saída e o sinal desejado,  $\mu$  é o passo de atualização e  $\mathbf{w}(k)$  é o vetor de coeficientes do filtro adaptativo. No caso do equalizador, o sinal desejado (ou seja, o símbolo efetivamente transmitido) é a seqüência de treinamento. A única restrição é que essa seqüência deve ter no mínimo tantos elementos quanto o tamanho do equalizador [1]. Um exemplo de uma implementação de equalizador utilizando LMS é mostrado na figura 2.3 [1].

Para aplicações em tempo real, a velocidade de convergência do LMS pode se tornar inadequada. Quando a razão entre os autovalores máximo e mínimo do matriz de covariância do sinal recebido for próxima de 1, a convergência ocorre de maneira relativamente rápida. Por outro lado, quando o autovalor máximo for muito maior que o mínimo, a convergência é bem mais lenta [1]. Em [8], são apresentados vários algoritmos baseados em LMS que se propõem a melhorar a velocidade de convergência do mesmo, como o LMS normalizado, o LMS de domínio transformado e o LMS com projeção afim (*affine projection*).

No caso de implementações adaptativas de equalizadores fracionalmente

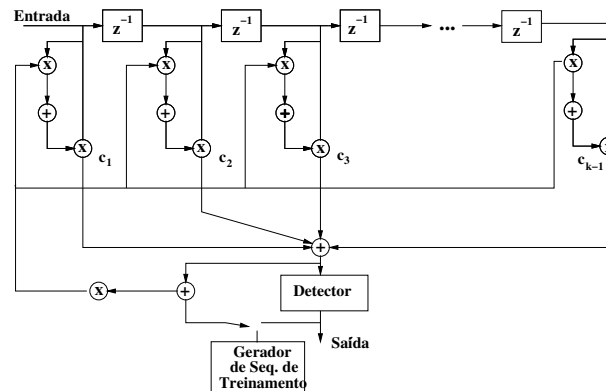


Figura 2.3: Exemplo de implementação de equalizador utilizando LMS.



espaçados, a escolha do parâmetro  $\mu$  deve ser cuidadosa. Nos FSE adaptativos, uma fração dos autovalores tem valores muito pequenos em relação aos demais [1]. Em implementações em máquinas de precisão finita, erros devido à quantização tendem a se acumular exatamente nas direções dos autovetores associados a esses pequenos autovalores. Esse problema do sistema, no entanto, não se reflete na métrica MSE [1].

### 2.6.3 MLSE Adaptativo

Conforme descrito na Subseção 2.4.1, a implementação do MLSE requer o conhecimento dos coeficientes do canal discreto no tempo  $\{h(k)\}$ , tanto no algoritmo de Viterbi quanto na parte de detecção [1]. No caso de o canal ser desconhecido ou variante no tempo, emprega-se uma solução adaptativa na qual um estimador do canal é colocado em paralelo ao algoritmo de Viterbi, como visto na figura 2.4.

Um estimador de canal tem um objetivo similar ao de um equalizador: descobrir os coeficientes do canal. A diferença básica é que o equalizador, em particular o *zero-forcing*, tem como objetivo inverter o canal, enquanto o estimador visa apenas disponibilizar estes coeficientes a fim de serem usados pelo algoritmo de Viterbi. A estrutura interna do estimador de canal é a mesma do equalizador *zero-forcing* visto na Subseção 2.4.2 [1].

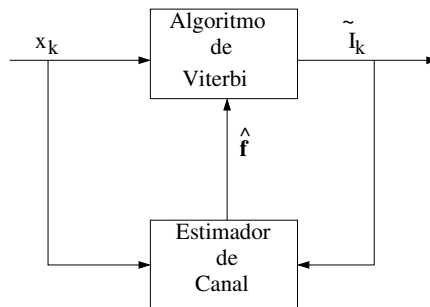


Figura 2.4: Estrutura básica de um MLSE adaptativo.

### 2.6.4 DFE Adaptativo

Do mesmo modo que o equalizador linear pode ter seus coeficientes definidos de maneira adaptativa, os coeficientes dos filtros do equalizador *feedforward* e o de *feedback* também podem ser definidos a partir de algoritmos adaptativos.

A equação (2.33), que relaciona os coeficientes do equalizador *feedforward* com os coeficientes do canal, inclui em sua expressão uma inversão de matriz, o que é bastante caro computacionalmente, principalmente com o aumento do número de coeficientes do canal. Ao se implementar um DFE adaptativo, evita-se a inversão de matriz, tornando o sistema mais ágil.

### 2.6.5 Implementações Baseadas em RLS

O uso do LMS, como visto na Subseção 2.6.2, tem a desvantagem da convergência lenta. Uma opção é o uso do algoritmo RLS (*recursive least squares*). Além de possuir uma convergência mais rápida que o LMS, a velocidade de convergência no RLS não depende da razão dos autovalores da matriz de correlação [8]. Uma desvantagem da abordagem do RLS é, no entanto, a maior complexidade computacional do sistema e possíveis instabilizações, problema que não afeta o LMS [8]. Outra vantagem do RLS é o seu bom funcionamento, em geral, em ambientes não estacionários, devido a sua capacidade de rápido rastreamento de mudanças nas estatísticas do meio. Essa capacidade é decorrente do relacionamento próximo entre seu funcionamento e a filtragem de Kalman.

A diferença básica do RLS em relação ao LMS diz respeito à função objetivo (ou seja, a métrica que se deseja minimizar). No caso do LMS, deve-se minimizar uma aproximação instantânea do MSE, que é dada [8] por  $|e(k)|^2$ . Já no caso do RLS, a métrica a ser minimizada são os mínimos quadrados, que leva em consideração o valor do erro em instantes anteriores como estimativa do erro médio quadrático. A atualização dos coeficientes no RLS é feita a partir de [8]:

$$\mathbf{w}(k) = \mathbf{w}(k-1) + e'(k)\mathbf{S}_D(k)\mathbf{x}(k) \quad (2.39)$$

$$\mathbf{S}_D(k+1) = \frac{1}{\lambda} \left[ \mathbf{S}_D(k) - \frac{\mathbf{S}_D(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{S}_D^T(k)}{\lambda + \mathbf{x}^T(k)\mathbf{S}_D^T(k)\mathbf{x}(k)} \right] \quad (2.40)$$

onde  $\mathbf{w}(k)$  representa os coeficientes do equalizador no instante  $k$ . Além disso,  $e'(k)$  é o erro *a priori* do sistema enquanto  $\mathbf{x}(k)$  representa o vetor de entrada e  $\mathbf{S}_D(k)$  é a inversa da matriz de correlação do sinal de entrada, calculada de maneira iterativa.

## 2.7 Conclusões

Neste capítulo foram apresentados algoritmos *batch* (não-adaptativos) e adaptativos para equalização não-cega, isto é, com algum conhecimento do canal utilizado, seja por conhecimento prévio ou pelo uso de seqüências de treinamento.

O objetivo principal deste capítulo é tentar dar uma visão geral do problema de equalização antes de se colocar as restrições próprias deste trabalho, ou seja, uso de algoritmos cegos em ambientes multiusuários. A importância deste capítulo é basicamente mostrar as técnicas de equalização consolidadas e que explicam até certo ponto a maneira como a questão da equalização cega é geralmente tratada.

# Capítulo 3

## Equalização Cega

### 3.1 Introdução

Nos algoritmos de equalização adaptativa descritos até agora, a atualização dos coeficientes é feita através do uso de seqüências de treinamento. No entanto, em algumas situações, não é interessante que essas seqüências sejam usadas. Cenários baseados em testes não-invasivos e de interceptação de sinais foram as principais motivações para equalização cega [9], exatamente por serem aplicações em que o transmissor não sabe da existência do receptor. Além disso, o uso de seqüências de treinamento demanda um certo gasto de banda, o que, em certas aplicações em tempo real, pode tornar a implementação inviável. No caso de sistemas *multicast*, o treinamento de cada novo cliente demandaria uma perda imensa da banda passante, o que poderia inviabilizar a transmissão. De fato, se a cada novo cliente que entrasse no sistema fosse necessário que esse solicitasse uma seqüência de treinamento, o sistema perderia muito em flexibilidade. Por fim, em aplicações em que um servidor provê um serviço em tempo real, como em sistemas de *broadcast*, a degradação do serviço a cada vez que surge um novo cliente causaria danos à qualidade de serviço do sistema.

Em resumo, os algoritmos cegos apresentam bastante utilidade quando o transmissor não tem conhecimento de alguns dos receptores a serem utilizados,

quando a perda de banda for crítica ao serviço e nas situações em que o número de clientes for muito grande.

Ao contrário do que se poderia esperar, porém, o uso comercial de algoritmos de equalização cega é restrito no momento. Um ponto crucial nesta discussão é a velocidade de convergência dos algoritmos adaptativos, que se revela muito importante para aplicações em tempo real. Algoritmos de equalização cega geralmente precisam de mais amostras que seus equivalentes não-cegos para atingir a convergência. A necessidade de um número maior de amostras tem origem numa dependência maior com relação às estatísticas do sinal recebido para se avaliar que sinal foi transmitido.

Por necessitarem que se manipulem estatísticas de ordem alta do sinal recebido ou propriedades mais complexas, como cicloestacionaridade, algoritmos cegos tendem a exigir um custo de implementação bem maior que os não-cegos para realizarem a equalização. Isso pode ter um impacto considerável na velocidade do sistema, especialmente quando não se dispõe de *hardwares* adequados à sua implementação, ou no custo do sistema, visto que implementações específicas exigem um maior custo financeiro.

Neste capítulo, é feita uma introdução ampla do conceito de equalização cega. Na seção 3.2 será mostrada uma classificação preliminar dos algoritmos de equalização cega. Uma visão geral dos primeiros algoritmos de equalização cega é apresentada na seção 3.3. Os algoritmos cegos que se utilizam de estatísticas de segunda ordem do sinal transmitido e recebido são mostrados nas seções 3.4 e 3.5.

## 3.2 Tipos de Equalização Cega

De várias maneiras o problema de equalização cega do canal está relacionado com o problema de identificação do canal. No entanto, o enfoque de cada um desses tópicos é bastante particular. No caso da identificação do canal deseja-se obter qual o modelo de filtro digital cuja resposta em frequência que mais se aproxima do canal.

### 3.2. TIPOS DE EQUALIZAÇÃO CEGA

---

Já a questão da equalização cega se baseia na descoberta da seqüência transmitida dado que o canal é desconhecido. A identificação do canal pode servir como uma poderosa ferramenta para se chegar a esta resposta. Ao se focar a questão de equalização de uma maneira mais pragmática, baseando-se na SER (*symbol error rate*) no receptor, as condições para equalização cega de maneira completa são menos rígidas que no caso de se desejar identificação cega exata de canal.

Devido à estreita ligação entre os problemas de equalização e identificação de canal, a equalização pode ser feita a partir do conhecimento dos coeficientes do modelo do canal. Nesse caso, a equalização é dita indireta. Quando a equalização cega é feita sem o uso da identificação de canal, dá-se o nome de equalização direta.

No caso de sistemas SISO (*single input single output*), os primeiros métodos de equalização exploraram estatísticas de alta ordem (HOS - *higher-order statistics*) do sinal desejado, já que estatísticas de segunda ordem não trazem informações suficientes sobre a fase do sinal. Ao se usar algoritmos adaptativos para equalização direta, a utilização estatísticas de alta ordem está normalmente implícita na função objetivo do algoritmo.

Mais recentemente, foram desenvolvidos métodos que utilizam certas propriedades estatísticas do sinal transmitido, como cicloestacionaridade, sem fazer uso de HOS. O problema de se identificar a resposta ao impulso do canal antes de se efetuar a equalização cega é muito semelhante ao problema de desconvolução cega, que tem sido estudado separadamente para a análise de séries temporais.

Para a equalização direta ou indireta de sistemas SIMO (*single input multiple output*), tenta-se explorar a diversidade de informações devido às múltiplas saídas a fim de se evitar o uso de estatísticas de ordem elevada. Com isso, a equalização cega é feita utilizando-se apenas estatísticas de segunda ordem do sinal de entrada. Essa abordagem evita alguns problemas que surgem ao se utilizar HOS como a presença nociva de mínimos locais, e a lenta convergência de algoritmos adaptativos, além do alto custo computacional da HOS.

## 3.3 As Origens da Equalização Cega

Os primeiros algoritmos de equalização cega se baseavam em um sistema com modulação PAM (*pulse amplitude modulation*) com  $M$  níveis de informações, simétricas em relação ao zero.

Supõe-se que a probabilidade de se enviar qualquer dos  $M$  símbolos seja a mesma nesses equalizadores. Sendo  $y(k) = \mathbf{w}^H \mathbf{r}$  a seqüência de símbolos na saída do equalizador e  $\mathbf{w}(k)$  os coeficientes do filtro adaptativo, a expressão da função objetivo é dada por [10]:

$$F(\mathbf{w}) = E\{\Psi(y(k))\} \quad (3.1)$$

onde  $\Psi(y(k))$  deve ser uma função par [10].

Utilizando um algoritmo do tipo gradiente estocástico, chega-se à seguinte expressão [10]

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \frac{\partial \Psi(y(k))}{\partial \mathbf{w}(k)} \quad (3.2)$$

Chamando-se  $\psi$  a derivada de  $\Psi$  em relação a  $\mathbf{w}$ , chega-se à expressão geral de atualização para os primeiros algoritmos adaptativos cegos [10]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \psi(\mathbf{w}^T(k)\mathbf{r}(k))\mathbf{r}(k) \quad (3.3)$$

A função  $\psi(\cdot)$  é chamada de função erro devido ao fato de substituir o erro na expressão de algoritmos de gradiente estocástico.

A escolha da melhor função objetivo, e, conseqüentemente da função erro, para a convergência do equalizador é a grande questão desses algoritmos, que em geral utilizam HOS implícitas na função objetivo.

### 3.3.1 Algoritmo de Sato

O algoritmo introduzido por Yoichi Sato em 1975 é considerado o primeiro trabalho de equalização cega, na época sob o nome de *self-recovering equalization*. Neste algoritmo, a função erro é dada por, com a restrição  $E[b(k)] \neq 0$ :

$$\psi(y(k)) = y(k) - \frac{E[(b(k))^2]}{E[b(k)]} \text{sgn}(y(k)) \quad (3.4)$$

onde  $b(k)$  representa o símbolo transmitido. A equação que atualiza os coeficientes do filtro adaptativo é dada por

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \left[ y(k) - \frac{E[(b(k))^2]}{E[b(k)]} \text{sgn}(y(k)) \right] \mathbf{r}(k) \quad (3.5)$$

A partir de (3.4), descobre-se que a função objetivo que está sendo minimizada tem a seguinte forma.

$$\Psi(y(k)) = \frac{1}{2} \left( |y(k)| - \frac{E[(b(k))^2]}{E[b(k)]} \right)^2 \quad (3.6)$$

#### 3.3.2 Algoritmo BGR

Em 1980, Benveniste, Goursat e Rouget descreveram um novo algoritmo, do qual o algoritmo de Sato tornou-se apenas um caso particular. Nesse algoritmo, a função erro é dada por [10]:

$$\psi(y(k)) = f(y(k)) - \frac{E[f(b(k))b(k)]}{E[b(k)]} \text{sgn}(y(k)) \quad (3.7)$$

A principal restrição do algoritmo é de que a função  $f(\cdot)$  deve ser ímpar e duas vezes diferenciável, com segunda derivada não-negativa em todo o seu domínio. A equação (3.4) do algoritmo de Sato representa o caso em que  $f(x) = x$  para a equação (3.7).

#### 3.3.3 Algoritmos *Stop-and-Go*

A convergência tanto do algoritmo de Sato quanto do BGR é dependente do fato de o sinal de sua função erro  $\psi(y(k))$  em cada instante coincidir com o sinal do erro  $[y(k) - b(k-d)]$  ( $d$  é o atraso introduzido pela operação de equalização) do algoritmo LMS. A fim de se eliminar essa dependência, foram desenvolvidos algoritmos que atualizam os coeficientes do filtro adaptativo apenas se o sinal dessas duas funções coincidirem. Esses algoritmos receberam o nome de *Stop-and-Go*. Define-se duas funções de erro  $\psi_1(y(k))$  e  $\psi_2(y(k))$ , a idéia do algoritmo *stop-and-*



$g_o$  é resumida pela expressão:

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) - \mu\psi_1(y(k))\mathbf{x}(k), & \text{se } \text{sgn}[\psi_1(y(k))] = \text{sgn}[\psi_2(y(k))] \\ \mathbf{w}(k) & \text{se } \text{sgn}[\psi_1(y(k))] \neq \text{sgn}[\psi_2(y(k))] \end{cases} \quad (3.8)$$

O uso destes algoritmos propiciam uma convergência mais rápida do equalizador cego, além de se evitar de maneira mais eficiente a convergência para mínimos locais.

#### 3.3.4 Algoritmo de Bussgang

A idéia do algoritmo de Bussgang é chegar à seqüência  $\hat{b}(k)$  que seja a melhor estimativa de  $b(k)$  a partir da seqüência  $y(k)$ , ou seja, da saída do equalizador. O julgamento sobre a melhor estimativa é feita a partir do critério MAP (*maximum a posteriori*), definido por [10]

$$\hat{b}(k-d) = \arg \max_b p(y(k)|b) \quad (3.9)$$

É mostrado em [10] que, dada a variância do ruído na saída do equalizador, o critério MAP leva ao estimador de mínima variância:

$$g(y(k)) = \hat{b}(k-d) = \min E\{b(k-d)y(k)\} \quad (3.10)$$

onde  $g(y(k))$  é um sistema não-linear sem memória que serve como estimador da seqüência transmitida. A chamada condição de Bussgang para o funcionamento do sistema é dada por [10]:

$$E\{y(k-m)y(k)\} = E\{y(k-m)g(y(k))\} \quad (3.11)$$

A atualização do sistema é então:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu[g(y(k)) - y(k)]\mathbf{r}(k) \quad (3.12)$$

#### 3.3.5 Algoritmo de Godard

A partir de uma simples modificação na fórmula (3.6), Godard chegou, no início da década de 1980, a uma classe mais geral de equalizadores. Sua nova função

custo tem a seguinte forma [10]:

$$\Psi(y(k)) = \frac{1}{2q} \left( |y(k)|^q - \frac{E[(b(k))^{2q}]}{E[(b(k))^q]} \right)^2 \quad (3.13)$$

onde o parâmetro  $q$  corresponde a uma constante positiva. A atualização dos coeficientes do equalizador adaptativo é feita de acordo com

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \left( |y(k)|^q - \frac{E[(b(k))^{2q}]}{E[(b(k))^q]} \right) |y(k)|^{q-2} y(k) \mathbf{r}^*(k) \quad (3.14)$$

Em 1982, Treichler desenvolveu, de maneira independente, o algoritmo de módulo constante (CMA - *constant modulus algorithm*) que corresponde ao caso de  $q = 2$ . Este algoritmo, no entanto, foi desenvolvido mais especificamente para aplicações em que o sinal de entrada é tal que

$$|b(k)|^2 = \frac{E[|b(k)|^4]}{E[|b(k)|^2]} \quad (3.15)$$

## 3.4 Equalização Cega para Sistemas SIMO

Como já foi abordado na seção 3.2, a utilização de HOS para se equalizar um canal traz algumas desvantagens ao sistema, dentre as quais pode-se destacar a presença de mínimos locais, a convergência lenta e um elevado custo computacional. Foram feitas várias tentativas de se evitar o uso de HOS na equalização. Uma delas consiste em se explorar propriedades estatísticas, como a cicloestacionaridade, do sinal transmitido. Em outras situações, procurou-se usar apenas estatísticas de segunda ordem do sinal transmitido. No entanto, a utilização de estatísticas de até segunda ordem prejudica a identificação do sistema, principalmente no que se refere à identificação de fase do sistema. Em sistemas SIMO, entretanto, há uma certa redundância implícita nas respostas dos múltiplos canais. Essa redundância se deve a todos os subcanais serem excitados pela mesma seqüência de entrada. A presença dessa redundância possibilita que os algoritmos de equalização utilizados se baseiem apenas em estatísticas de segunda ordem, evitando-se os problemas descritos para HOS.

A abordagem utilizada para sistemas SIMO é equivalente à utilização de um sistema SISO superamostrado no tempo, baseando-se na descrição feita em [10].

### 3.4. EQUALIZAÇÃO CEGA PARA SISTEMAS SIMO

Nesta seção, o sinal recebido a partir de um canal analógico obedece à seguinte relação:

$$r(t) = \sum_{u=1}^p \sum_{k=-\infty}^{\infty} b(k)h_u(t - kT) + n(t) \quad (3.16)$$

Faz-se uma superamostragem tal que  $h_u(k) \triangleq h_u(kT/q)$  [10], sendo  $q$  inteiro. Com isso,

$$\mathbf{r}(l) = \sum_{u=1}^p \sum_{k=-\infty}^{\infty} \mathbf{b}(k)h_u(l - kq) + \mathbf{n}(l) \quad (3.17)$$

O sistema SIMO equivalente a esta superamostragem no tempo é mostrado na figura 3.1 [10]. Nota-se que todo o desenvolvimento feito aqui para sistemas SIMO

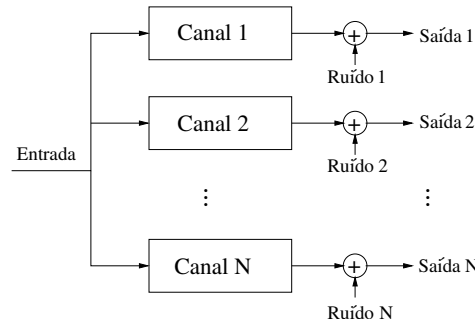


Figura 3.1: Sistema SIMO equivalente ao sistema SISO superamostrado.

pode ser estendido para um sistema MIMO (*multiple input multiple output*). A única condição que deve ser aceita para se fazer essa extensão é que o sistema MIMO possua o número de subcanais maior que o número de usuários.

Supondo que o sistema tenha  $p$  usuários e  $q$  canais, definem-se os seguintes vetores [10]:

$$\mathbf{b}(k) = \begin{bmatrix} b_1(k) \\ b_2(k) \\ \vdots \\ b_q(k) \end{bmatrix}, \quad \mathbf{r}(k) = \begin{bmatrix} r_1(k) \\ r_2(k) \\ \vdots \\ r_q(k) \end{bmatrix}, \quad \mathbf{n}(k) = \begin{bmatrix} n_1(k) \\ n_2(k) \\ \vdots \\ n_q(k) \end{bmatrix} \quad (3.18)$$

Além disso, a matriz  $\mathbf{H}(i)$  que contém os coeficientes de todos os subcanais no

tempo  $i$  é dada pela seguinte expressão [10]:

$$\mathbf{H}(i) \triangleq \begin{bmatrix} h_1(iq) & h_2(iq) & \cdots & h_p(iq) \\ h_1(iq+1) & h_2(iq+1) & \cdots & h_p(iq+1) \\ \vdots & \vdots & \ddots & \vdots \\ h_1(iq+q-1) & h_2(iq+q-1) & \cdots & h_p(iq+q-1) \end{bmatrix} \quad (3.19)$$

Forma-se então a matriz Toeplitz em bloco  $\tau_L$ : [10].

$$\tau_L(\mathbf{H}) = \begin{bmatrix} \mathbf{H}(0) & \mathbf{H}(1) & \cdots & \mathbf{H}(M) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}(0) & \mathbf{H}(1) & \cdots & \mathbf{H}(M) & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}(0) & \mathbf{H}(1) & \cdots & \mathbf{H}(M) \end{bmatrix} \quad (3.20)$$

e o sistema MIMO pode ser representado por

$$\mathbf{r}(k) = \tau_L(\mathbf{H})\mathbf{b}(k) + \mathbf{n}(k) \quad (3.21)$$

Os algoritmos de equalização baseiam-se na matriz de covariância da saída do canal conforme a seguinte equação [10]:

$$\mathbf{R}_b = \tau_L(\mathbf{H})\tau_L^H(\mathbf{H}) + \sigma_n^2\mathbf{I} \quad (3.22)$$

## 3.5 Equalização Cega Utilizando Cicloestaciona- ridade

De uma maneira geral, a principal dificuldade para o uso de estatísticas de segunda ordem para a identificação de canal é a falta de informações com respeito à fase. Tendo-se a densidade espectral de potência (PSD - *power spectrum density*) do sinal transmitido (indicada por  $S_b(j\omega)$ ), as informações com respeito à resposta em magnitude do canal podem ser obtidas a partir da densidade espectral de potência na saída do canal ( $S_r(j\omega)$ ), através da seguinte relação:

$$|H(j\omega)|^2 = \frac{S_r(j\omega)}{S_b(j\omega)} \quad (3.23)$$

No entanto, para uma certa classe de processos estocásticos é possível que se recuperem informações sobre a fase do canal. Isso é possível através da utilização da densidade espectral de correlação (SCD - *spectral correlation density*). Essa grandeza só está definida para processos cicloestacionários, sendo obtida pela transformada de Fourier da função autocorrelação cíclica do sinal. Com isso, a fim de se evitar o uso de HOS, basta que o sinal utilizado seja cicloestacionário tanto na entrada quanto na saída do canal, permitindo que se use a SCD.

Considera-se um sinal QAM (*quadrature amplitude modulation*)  $b(k)$  que passa por um canal analógico  $h(t)$ . A saída do canal pode ser expressa por

$$r(t) = \sum_{k=-\infty}^{\infty} b(k)h(t - kT + t_0) + n(t) \quad (3.24)$$

Supondo  $b(k)$  e  $n(t)$  independentes e estacionários, com as respectivas funções autocorrelação dadas por

$$R_b(k - l) = E[b(k)b^*(l)] \quad (3.25)$$

$$R_n(t_2 - t_1) = E[n(t_2)n^*(t_1)] \quad (3.26)$$

a autocorrelação do sinal na saída do canal é dada por [10]:

$$\begin{aligned} R_r(t_1, t_2) &\triangleq E[r(t_1)r^H(t_2)] \\ &= \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} E[b(k)b^*(l)]h(t_1 - kT + t_0)h^*(t_2 - lT + t_0) + R_n(t_2 - t_1) \\ &= \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} R_b(k - l)h(t_1 - kT + t_0)h^*(t_2 - lT + t_0) + R_n(t_2 - t_1) \end{aligned} \quad (3.27)$$

Este processo é cicloestacionário no sentido amplo, dado que:

$$E[\mathbf{r}(t + T)] = E[\mathbf{r}(t)] \quad (3.28)$$

$$R_r(t_1 + T, t_2 + T) = R_r(t_1, t_2) \quad (3.29)$$

Considera-se que o sinal analógico na saída do canal seja amostrado com uma taxa  $q$  vezes a taxa de símbolos original.

Pode ser mostrado que a SCD do sinal da saída do canal amostrado no tempo obedeça a seguinte relação com a função de transferência do canal [10]:

$$S_r(z) = H(z)H^*(e^{-j\beta}/z^*) + qN_0\sigma(l) \quad (3.30)$$

A partir dessa relação, juntamente com a equação (3.23), consegue-se recuperar informações sobre a resposta de fase do canal utilizado.

## 3.6 Conclusões

Neste capítulo são vistos os principais algoritmos que deram origem à equalização cega, mostrando a evolução dos primeiros algoritmos, específicos para PAM e que utilizavam manipulação de momentos de terceira e quarta ordens até algoritmos mais gerais que se valem apenas de momentos estatísticos de segunda ordem.

O objetivo deste capítulo é mostrar, através da evolução histórica algumas razões pelas quais os algoritmos de equalização cega assumiram as formas atuais. Alguns problemas encontrados nos primeiros algoritmos se refletem em restrições colocadas nos novos, como, por exemplo, o uso da suposição de cicloestacionaridade para combater o problema da identificação simultânea de módulo e fase da resposta do canal. O entendimento da origem das restrições e suposições presentes nos novos algoritmos facilita um entendimento mais profundo de seu funcionamento.

# Capítulo 4

## CDMA

### 4.1 Introdução

*Code Division Multiple Access* é atualmente a técnica mais promissora para o compartilhamento de acesso em sistemas de telefonia celular. Seu funcionamento se baseia no espalhamento do espectro do sinal de informação, através do uso de um código específico, diferente para cada usuário. A diversidade criada pelos códigos permite que os sinais de informação possam ser transmitidos usando-se o mesmo *slot* de tempo e a mesma faixa de frequência, o que não acontece nos sistemas baseados em TDMA (*time division multiple access*) e FDMA (*frequency division multiple access*).

O uso de espalhamento espectral (*spread spectrum*) em sistemas de comunicações é bem mais antigo que os sistemas comerciais modernos de telefonia celular. Sua utilização no meio militar data de meados do século XX, tendo sua pesquisa motivada para o combate de ruídos de *jamming* [11].

Com o surgimento de sistemas de telefonia móvel, a pesquisa na área de espalhamento espectral passou a ter ênfase no desenvolvimento de técnicas de compartilhamento de acesso ao meio de transmissão.

Neste capítulo serão apresentados os principais aspectos teóricos do CDMA.

O escopo do capítulo será, no entanto, limitado a tratar apenas do embasamento teórico ligado às partes do CDMA efetivamente implementadas. O sistema de controle de potência, por exemplo, tão necessário na implementação dos sistemas comerciais baseados em CDMA, não é apresentado aqui, pois a potência dos usuários foi, nas simulações, igualada e os canais implementados de modo a não colocar ganhos no sinal. A seção 4.2 tratará do espalhamento espectral, uma das técnicas utilizadas pelo CDMA, que é na verdade anterior aos próprios sistemas de telefonia móvel. A seção 4.3 apresentará os códigos de espalhamento, que são os elementos do sistema que permitem a separação, no receptor, dos dados de informação referentes a cada usuário. A seguir, na seção 4.4, será mostrado como a questão do multipercurso é tratada atualmente nos sistemas CDMA através do uso de receptores RAKE. A seção 4.6 apresentará a questão da detecção multiusuário, que foi uma das principais motivações para o desenvolvimento de equalizadores cegos para sistemas multiusuário, como mostrado na seção 4.7. Por fim, na seção 4.8, será detalhada a implementação utilizada para o CDMA, inclusive com o modelo de dados utilizado no restante desta documentação.

## 4.2 Espalhamento Espectral

A idéia básica do espalhamento espectral é a de, a partir de um sinal de informação qualquer, gerar um sinal com estatísticas bem próximas às de um ruído branco, com banda muito maior que o sinal original [11]. Há duas técnicas principais para se fazer o espalhamento espectral: *frequency hopping* (FH) ou *direct sequence* (DS).

No espalhamento espectral do tipo FH, a frequência no centro da banda do sinal transmitido é mudada progressivamente [11]. A fim de que o sinal gerado seja semelhante a um ruído, a mudança na frequência é feita de acordo com uma seqüência pseudo-aleatória.

No caso do espalhamento espectral por *direct sequence*, cada símbolo de informação é multiplicado por uma seqüência pseudo-aleatória conhecida como



código de espalhamento. É gerada então uma seqüência com período mais curto que a original. Cada elemento dessa seqüência recebe o nome de *chip*.

O uso de espalhamento espectral apresenta várias vantagens em relação a outros métodos de compartilhamento de acesso. Primeiramente, há uma resistência em relação a ruídos de banda estreita, na medida em que o sinal é espalhado por uma faixa mais larga de frequências [6]. Além disso, a presença de multipercursos tem seus efeitos atenuados pelo uso de um número maior de elementos para se detectar o símbolo. Por fim, o uso de códigos individuais para os usuários garante uma certa privacidade na troca de informações [6].

O método a ser focado nesta documentação é o espalhamento espectral do tipo DS, pois é a técnica empregada na grande maioria dos sistemas existentes com espalhamento espectral. Nas seções e capítulos a seguir, quando se mencionar CDMA está se referindo ao DS-CDMA, a menos que dito o contrário.

## 4.3 Códigos de Espalhamento

A fim de que a faixa de frequências seja a maior possível para o sinal espalhado, esse deve se aproximar o máximo possível de um ruído branco. No entanto, a fim de se recuperar o sinal de informação original, o mesmo código deve estar presente também no receptor. Com isso, lança-se mão de seqüências pseudo-aleatórias (PN - *pseudo-noise*) [12].

### 4.3.1 Códigos de Walsh-Hadamard

Um dos tipos de seqüências pseudo-aleatórias mais usadas corresponde ao código de Walsh-Hadamard. Essas seqüências têm a propriedade de serem ortogonais entre si, mas não são necessariamente ortogonais a versões atrasadas de si

### 4.3. CÓDIGOS DE ESPALHAMENTO

mesmas. O algoritmo para gerá-las se baseia na matriz  $2 \times 2$  de Haddamard:

$$A_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (4.1)$$

Se a seqüência PN que se deseja tem tamanho 2, então basta se usar cada linha de  $A_2$  como código de espalhamento. Para gerar seqüências de tamanho maior basta aplicar o seguinte algoritmo recursivo.

$$A_{2K} = \begin{bmatrix} A_K & A_K \\ A_K & -A_K \end{bmatrix}, \text{ onde } K \text{ é potência de } 2 \quad (4.2)$$

As seqüências de tamanho  $K$  geradas são as linhas da matriz  $A_K$ .

#### 4.3.2 Códigos de Gold

Considera-se uma estrutura *simple shift register generator* (SSRG) como a representada na figura 4.1 [12]. A saída de cada atrasador pode ser vista como um estado do sistema. Esses estados se sucedem até voltar ciclicamente à seqüência de estados original. As seqüências de estados com maior comprimento são denominadas seqüências de comprimento máximo [12]. Essas seqüências têm comprimento  $2^m - 1$  para sistemas com  $m$  estados [12] [1]. Os SSRGs são casos particulares de estruturas mais gerais conhecidas como *linear feedback shift registers* (LFSR).

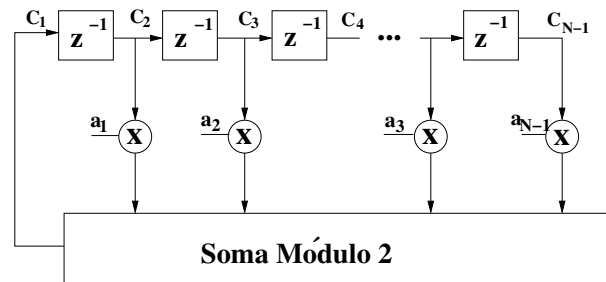


Figura 4.1: *Simple Shift Register Generator*.

Essas seqüências de comprimento máximo, no entanto, não são completamente ortogonais entre si. Com isso, o desespalhamento não garante a separação

total entre as informações de cada usuário. Além disso, quanto maior o comprimento das seqüências máximas, maior é o valor para a correlação entre seqüências diferentes [1].

Considera-se um sistema semelhante ao da figura 4.1 com  $m$  estados gerando seqüências de comprimento máximo com tamanho  $n = (2^m - 1)$ . Para alguns pares dessas seqüências de comprimento máximo, a correlação entre as seqüências pode assumir apenas três valores:  $\{-1, -t(m), t(m) - 2\}$ , onde  $t(m)$  é dado pela expressão a seguir [1]:

$$t(m) = \begin{cases} 2^{(m+1)/2} + 1, m \text{ ímpar} \\ 2^{(m+2)/2} + 1, m \text{ par} \end{cases} \quad (4.3)$$

Esses pares de seqüências são conhecidos como seqüências preferidas. A partir de um par de seqüências preferidas  $\mathbf{a}$  e  $\mathbf{b}$  com comprimento  $n$ , constroem-se  $n$  seqüências através da soma módulo 2 entre  $\mathbf{a}$  e versões deslocadas de  $\mathbf{b}$ . Ao se acrescentarem as seqüências  $\mathbf{a}$  e  $\mathbf{b}$  a esse conjunto, tem-se um novo conjunto de  $(n + 2)$  seqüências que corresponde a códigos de Gold. Além disso, para essas  $(n + 2)$  seqüências, a correlação entre essas seqüências e versões atrasadas das mesmas também só assume os valores  $\{-1, -t(m), t(m) - 2\}$ .

Nas implementações utilizadas nesta tese foram usados códigos de Gold de tamanho 31, por sua popularidade e por apresentarem uma baixa correlação entre os códigos, como visto nesta subseção.

## 4.4 Receptor RAKE

Como foi visto na seção 2.3, o uso de canais limitados em banda geram, no domínio do tempo, um espalhamento dos chips transmitidos. Esse efeito é semelhante ao produzido fisicamente por multipercursos.

Em certas situações, a interferência entre os símbolos acontece na ordem de um chip. O efeito do multipercurso então se traduz em reforço ou atenuação do símbolo transmitido, sem interferir com outros símbolos. A atenuação causada

pelo multipercurso se traduz no efeito de desvanecimento (*fading*) do sinal recebido. Nesses casos, um receptor RAKE permite usar a diversidade causada pelo multipercurso para minimizar os efeitos de desvanecimento do sistema.

Cada correlacionador representado na figura ?? procura por um multipercurso significativo dentro de uma janela de tempo. Na prática, ao se ter  $M$  correlacionadores no RAKE, consegue-se detectar os  $M$  multipercursos mais fortes [6]. A presença dos diversos correlacionares tem como função o combate aos efeitos de desvanecimento. Assumindo que o desvanecimento não seja uniforme em todos os multipercursos, os pesos  $a_i$  dos ramos podem ser ajustados de maneira a reforçar os ramos com maior SNR, ou seja, reforçando os que menos sofrem com desvanecimento [6].

A estrutura de um receptor RAKE implica em um custo computacional bem maior que a estrutura linear de um equalizador. Outro problema encontrado em relação ao receptor RAKE é uma degradação progressiva em sua resposta ao se aumentar o número de usuários, o que é resolvido pela adição de outras técnicas, como a detecção multiusuário (técnica vista na seção 4.6) [13].

## 4.5 Codificação de Canal

Uma estratégia usada para se evitar o efeito de ICI (*interchip interference*), e conseqüentemente da ISI nos sistemas comerciais *cdmaOne*, *cdma2000* e *3GSM* (antes conhecido como *UTRAN-WCDMA* é o uso de codificação de canal, mais particularmente, de códigos convolucionais [13] [14].

A codificação de canal consiste em um conjunto de técnicas baseadas nos limites de capacidade de canal dados a partir da teoria originalmente desenvolvida por Shannon na década de 1940 [14] [15]. Basicamente, a codificação de canal tem a função de gerar redundâncias nas informações transmitidas de modo que, mesmo ao se distorcer o sinal passando-o por um canal com memória, essas informações possam ser corretamente recebidas.

A abordagem feita pela codificação de canal para se eliminar a ISI é completamente diferente à apresentada pela equalização. Na codificação, busca-se, através de um certo conhecimento do canal utilizado, um código que gere redundância no sinal de modo que a saída do decodificador seja o sinal o mais próximo possível do original. Na equalização, busca-se reverter o efeito do canal filtrando-se certas componentes de frequência do sinal corrompido pela ISI. A vantagem da equalização sobre a codificação de canal é a economia de banda de transmissão, na medida em que é transmitido apenas o sinal original, sem a redundância e a flexibilidade apresentadas pelos equalizadores adaptativos, que exigem um menor conhecimento do meio de transmissão e que se adaptam a ambientes com estatísticas variantes no tempo.

## 4.6 Detecção Multiusuário

Uma das funções mais importantes de um sistema multiusuário é a separação das informações provenientes de cada um dos usuários do sistema. Tradicionalmente, isso é feito através de um conjunto de filtros casados com as seqüências de espalhamento dos usuários. Esse método é conhecido como detecção monousuário, na medida em que cada usuário é detetado individualmente sem se tirar nenhum proveito da detecção dos demais usuários [16].

A fim de que a detecção das informações do usuário em questão tenha uma alta eficiência, deve-se garantir que a correlação cruzada entre as seqüências de espalhamento seja desprezível se comparada com o valor da autocorrelação dessas seqüências.

Outra medida que deve ser adotada a fim de se garantir que a detecção monousuário seja eficiente é a presença do controle de potência. O controle de potência é importante, na medida em que, se um usuário transmitir seus dados com potência bem maior que os demais, certas correlações terão um valor alto. Isso não acontece pela similaridade entre os dados, mas pela amplitude dos valores de entrada [16].

#### 4.6. DETEÇÃO MULTIUSUÁRIO

---

No caso da detecção multiusuário (MUD - *multi-user detection*), informações referentes à detecção de um usuário são aproveitadas na detecção dos demais.

Supõe-se uma transmissão CDMA em que os canais referentes a cada usuário não tenham memória, ou seja, apenas gerem um ganho nos símbolos transmitidos. Considera-se que [16]:

- O ganho do canal referente ao usuário  $i$  seja dado por  $A_i$ .
- Seja  $\rho_{i,j}$  a correlação cruzada entre o usuário  $i$  e o usuário  $j$ .
- Os chips recebidos do canal sejam passados pelos filtros casados referentes à detecção monousuário.
- O sistema tem  $K$  usuários.
- O ruído é branco, aditivo e gaussiano.
- $z_i$  é a componente do ruído no sinal recebido ao passar pelo filtro casado referente ao usuário  $i$ .
- $d_i$  a componente do sinal (sem ruído) recebido ao passar pelo filtro casado referente ao usuário  $i$ .

Sendo:

$$\mathbf{R} = \begin{bmatrix} 1 & \rho_{2,1} & \rho_{3,1} & \cdots & \rho_{K,1} \\ \rho_{1,2} & 1 & \rho_{3,2} & \cdots & \rho_{K,2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{1,K} & \rho_{2,K} & \rho_{3,K} & \cdots & 1 \end{bmatrix} \quad (4.4)$$

$$\mathbf{A} = \begin{bmatrix} A_1 & 0 & 0 & \cdots & 0 \\ 0 & A_2 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & A_K \end{bmatrix} \quad (4.5)$$

$$\mathbf{d} = [d_1 \ d_2 \ \cdots \ d_K]^T \quad (4.6)$$

$$\mathbf{z} = [z_1 \ z_2 \ \cdots \ z_K]^T \quad (4.7)$$

tem-se que o vetor recebido na saída do detetor monousuário pode ser modelado matricialmente por

$$\mathbf{y} = \mathbf{R}\mathbf{A}\mathbf{d} + \mathbf{z} \quad (4.8)$$

Os detetores multiusuário são aplicados no vetor  $\mathbf{y}$ . Nota-se que a pré-multiplicação de  $\mathbf{A}\mathbf{d}$  pela matriz  $\mathbf{R}$  gera interferência entre os símbolos de diferentes usuários, o que dificulta a estimação do símbolo transmitido. Essa interferência é conhecida como MUI (*multi-user interference*). Nas próximas subseções serão mostrados os principais tipos de detecção multiusuário: por decorrelação, MMSE e por cancelamento sucessivo de interferência.

O esquema de funcionamento da detecção multiusuário é dado pela figura 4.2.

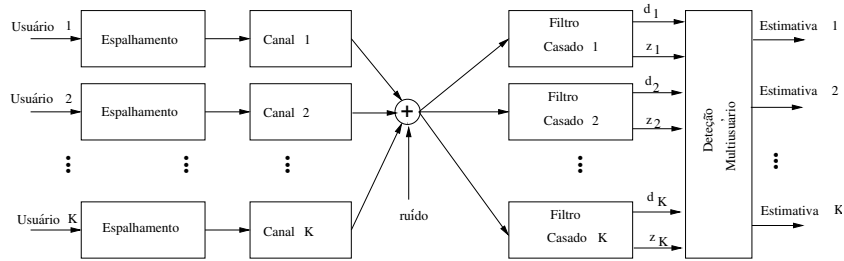


Figura 4.2: Sistema CDMA com MUD.

### 4.6.1 Detetor por Decorrelação

Neste tipo de detecção é aplicada a inversa da matriz de  $\mathbf{R}$  de correlação entre as seqüências de espalhamento sobre o vetor  $\mathbf{y}$  da saída do filtro casado. A seqüência detetada assume a forma [16];

$$\mathbf{y}_{\text{det}} = \mathbf{R}^{-1}\mathbf{y} = \mathbf{A}\mathbf{d} + \mathbf{R}^{-1}\mathbf{z} \quad (4.9)$$

Nota-se que a seqüência detectada  $\mathbf{y}_{\text{det}}$  não sofre mais influência da matriz de acoplamento  $\mathbf{R}$  que gera interferência entre seqüências de espalhamento de usuários diferentes.

De uma maneira geral, pode-se dizer que o detetor por descorrelação tem uma performance melhor que a do detetor monousuário. No entanto, a parcela  $\mathbf{R}^{-1}\mathbf{z}$  pode significar um reforço no efeito do ruído sobre a seqüência detetada. Para ambientes em que a potência do ruído seja relativamente alta, a perda de desempenho devida ao ruído pode ser bem mais grave que a MUI. Nesses casos, o uso do detetor por descorrelação pode deixar o sistema com um desempenho pior em relação à utilização do detetor monousuário [16].

### 4.6.2 Detetor MMSE

A fim de se evitar os problemas causados pelo reforço do ruído (*noise enhancement*) gerados pela detecção por descorrelação, foi desenvolvido um detetor  $\mathbf{M}$  que procura minimizar a métrica  $E[|\mathbf{d} - \mathbf{M}\mathbf{y}|^2]$ , onde  $E[\cdot]$  representa o operador valor esperado.

Para ruídos com baixa potência, o detetor por descorrelação tem um desempenho satisfatório. A idéia principal da métrica MMSE é evitar que o sistema tenha um desempenho bastante degradado para potências altas de ruído. Com isso, foi desenvolvido um detetor que segue a seguinte expressão [16]:

$$\mathbf{M} = [\mathbf{R} + \sigma^2\mathbf{I}]^{-1} \quad (4.10)$$

onde  $\sigma^2$  representa a variância do ruído transmitido. Percebe-se que, para valores pequenos de  $\sigma^2$ , o comportamento do detetor é semelhante ao do detetor por descorrelação. Para valores maiores da variância do ruído, a expressão matemática do detetor se afasta mais da simples inversão da matriz de correlação, evitando cair no problema de reforço do ruído.

### 4.6.3 Detetor por Cancelamento Sucessivo de Interferência

A idéia do detetor por cancelamento sucessivo de interferência (SIC - *Successive Interference Cancellation*) é reduzir a MUI causada por cada um dos usuários sucessivamente.



O método em geral precisa de um módulo antes do detetor monousuário que estime a potência transmitida por cada usuário e ordene os sinais dos usuários baseados nessa estimativa [16]. Faz-se então uma detecção monousuário apenas no usuário com a maior potência. A seguir, retira-se o sinal detetado deste usuário da mistura recebida, passando-se à detecção monousuário apenas do usuário com a segunda maior potência. Este procedimento continua sucessivamente até que todos os usuários sejam detetados.

A carga computacional extra proporcionada pelo SIC sobre o detetor monousuário tradicional é mínima se comparada aos detetores por descorrelação e MMSE. No entanto, o SIC apresenta alguns outros problemas como o atraso de resposta gerado pela presença dos vários estágios em cascata [16].

Ao se comparar o desempenho do SIC com o do detetor monousuário tradicional, nota-se que apenas os sinais de usuários com menor potência têm sua detecção melhorada pelo novo método, já que esses se beneficiam mais da subtração dos demais sinais.

## 4.7 Importância da Detecção Multiusuário

O uso de detetores multiusuário em sistemas CDMA resulta numa série de vantagens em relação aos detetores tradicionais. Essas vantagens são consequência de uma eliminação mais eficiente da interferência entre usuários.

Devido a problemas relacionados principalmente ao custo e às dimensões do móvel celular, a detecção multiusuário tem sua implementação restrita à estação base, e, por isso, o sistema com MUD pode ser utilizado apenas no *uplink*.

Além disso, o uso de detetores multiusuários permite que os requisitos de controle de potência sejam amenizados, mais uma vez por diminuir a interferência entre sinais de usuários diferentes. A atenuação desses requisitos traz uma menor complexidade na implementação do controle de potência. O controle da MUI também pode ser traduzido na redução da potência necessária para a transmissão

nos móveis, ou aumentar o alcance das células utilizadas [16], mantendo-se a mesma potência.

Apesar dos ganhos de performance apresentados ao se usar a MUD, essas melhorias ainda são limitadas por outras questões como a interferência entre usuários de células diferentes, influenciada pelo reuso de frequência, ou seja, a reutilização da mesma banda de transmissão em células diferentes.

Na detecção multiusuário apresentada, não foi considerado o efeito do multipercurso, que gera uma perda considerável de performance para esses detetores. Um conjunto de novos sistemas foi desenvolvido que leva em consideração o caso do multipercurso. Esses sistemas têm estruturas básicas equivalentes aos detetores mostradas nas subseções 4.6.1, 4.6.2 e 4.6.3. Eles apresentam como grande modificação a utilização da assinatura do usuário (ou seja, a convolução da seqüência de espalhamento com a resposta ao impulso do canal desse usuário) em lugar da seqüência de espalhamento para se fazer a detecção. Para isso, deve-se fazer uma estimação da resposta ao impulso dos canais associado a cada usuário. Isso é, de certa forma, associado ao problema de equalização do sistema, na medida em que se busca o conhecimento do efeito do canal com memória sobre a seqüência de dados original.

Assim como no caso da equalização, em muitas situações não se consegue estimar a resposta ao impulso dos canais através de seqüências de treinamento. Nesses casos, são feitas estimações cegas de canal a fim de se retirar a interferência intersimbólica e a interferência entre usuários conjuntamente no detetor. Como nessa técnica tem-se a eliminação dos efeitos do canal com memória no receptor, esses métodos são também abordados nessa tese como equalização cega. Na verdade, o desenvolvimento de detetores multiusuário que levem em conta o canal com memória foi uma das principais motivações para o desenvolvimento de técnicas de equalização cega para ambiente multiusuário.

## 4.8 Modelo de CDMA Implementado

Foi desenvolvida uma implementação com um modelo simplificado de CDMA de acordo com os seguintes elementos: espalhamento por código de canalização, passagem pelo canal digital e equalização. Como o caso mais crítico ocorre no *uplink*, onde os dados de cada usuário passam por multipercursos diferentes, foi implementada apenas essa parte do sistema. O modelo do sistema utilizado pode ser visto na figura 4.3.

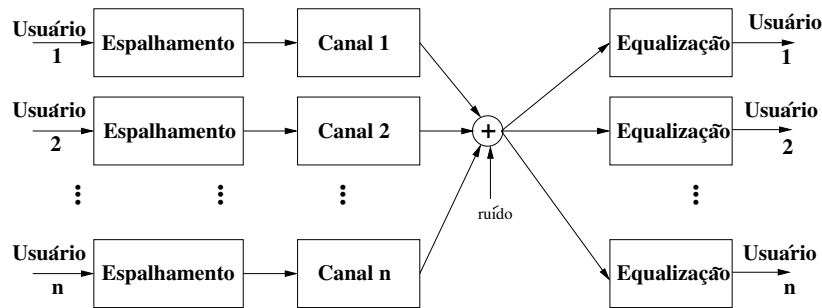


Figura 4.3: Modelo de CDMA implementado.

Resolveu-se não implementar as partes do sistema que não influenciavam na equalização, como a parte de modulação analógica, os códigos de embaralhamento (*scrambling*) e *interleaving*. Alguns outros elementos do sistema que teriam influência na equalização, como desvanecimento e a interferência entre os usuários, foram deixados de lado para se analisar melhor o combate à ISI.

No sistema implementado, fez-se uma série de suposições sobre o seu funcionamento:

- Sistema com  $K$  usuários.
- São transmitidos  $F$  símbolos.
- Cada usuário gera uma seqüência  $\{b_k(l)\}$ , ou  $\mathbf{b}_k$  com tamanho igual ao número de símbolos transmitidos, onde o índice  $k$  representa o usuário em questão e o parâmetro  $l$  é a evolução no tempo da seqüência.

- A transmissão utilizada é BPSK (*Binary Phase Shift-Keying*), respeitando-se as restrições do algoritmo de subespaço [17], descrito no capítulo 5.
- $\{c_k(l)\}$  ou  $\mathbf{c}_k$  com tamanho  $P$  denota o código de espalhamento de cada usuário. Essa seqüência é normalizada, de maneira que cada  $\mathbf{c}_k$  tenha norma-2 igual a 1.
- $\{h_k(l)\}$  ou  $\mathbf{h}_k$  com tamanho  $Q$  modela o canal com memória em questão, com os multipercursos relativo a cada usuário. O parâmetro  $l$  é relativo a cada multipercurso. Como não foi utilizado nenhum mecanismo de controle de potência, só foram definidos canais normalizados, ou seja,  $\|\mathbf{h}_k\|_2 = 1$ , onde *DFT* se refere à Transformada Discreta de Fourier. Essa medida é importante para se evitar que algum usuário tenha sua potência de transmissão aumentada em função do canal utilizado, melhorando seu desempenho e piorando o dos demais. O valor de  $h_k(l)$  quando  $0 \leq l \leq Q - 1$  é nulo.
- $\{s_k(l)\}$  ou  $\mathbf{s}_k$  é a assinatura de cada usuário, ou seja, a convolução do código de espalhamento com a resposta ao impulso do canal.
- O ruído  $\mathbf{n}(j)$ , com tamanho igual ao da seqüência de espalhamento, é aditivo, gaussiano e branco (AWGN - *additive white Gaussian noise*), com variância  $\sigma_n^2$ . Também pode ser expresso por  $\{n(j, l)\}, 0 \leq l \leq P$

Os sinais recebidos obedecem à seguinte expressão [17]

$$s_k(j) = \sum_{l=0}^{P+Q-1} c_k(j)h_k(l-j) \quad (4.11)$$

$$r(j, x) = \sum_{k=0}^{K-1} \sum_{l=0}^{F+P+Q-2} s_k(l-j-Px)b_k(j) + n(j, x) \quad (4.12)$$

O vetor  $\mathbf{r}(j)$ , correspondente à seqüência  $\{r(j, x)\}, 0 \leq x \leq P$  tem tamanho igual ao da seqüência de espalhamento, correspondendo, para cada  $l$ , ao vetor de chips recebidos que correspondem ao mesmo símbolo transmitido.

### 4.9 Conclusões

O objetivo deste capítulo foi apresentar o ambiente de comunicações utilizado nesta tese: o CDMA. São analisados os principais aspectos teóricos referentes ao CDMA, em particular, as questões relevantes para o sistema implementado. É também explicado como o problema da ISI é resolvido nos sistemas atuais.

Nas seções 4.2 até 4.5 foram descritos aspectos teóricos ligados às partes do sistema implementadas. Em particular, as seções 4.4 e 4.5 mostram como a questão dos multipercursos e ISI são tratados nos sistemas comerciais atuais.

As seções 4.6 e 4.7 mostram como surgiu a motivação de se pesquisar a questão da equalização cega para sistemas CDMA, que, por sua estrutura interna, já possui uma certa resistência à ISI. Como visto, a pesquisa em equalização para CDMA surgiu a partir da necessidade de se melhorar o desempenho de detetores multiusuário em sistemas com multipercursos. Esses detetores multiusuário, por sua vez, já são uma evolução dos detetores atualmente usados em sistemas comerciais, monousuário, baseados em filtros casados.

Na seção 4.8, o modelo do sistema implementado também foi apresentado, assim como a estrutura de dados utilizada. Esse modelo será usado na descrição dos algoritmos de equalização cega apresentados nos capítulos de 5 a 7.

# Capítulo 5

## Equalização Cega Usando Subespaço

### 5.1 Introdução

Como mostrado no capítulo anterior, a motivação inicial do uso de equalização cega para sistemas CDMA vem da tentativa de se melhorar o desempenho de detetores multiusuários. Neste capítulo será mostrado um dos primeiros métodos desenvolvidos para equalização cega em ambiente multiusuário, como descrito em [17]. No artigo em questão, a equalização cega é mostrada através de um caso particular de MUD para sistemas com presença de multipercursos.

Na seção 5.2 será mostrado como o algoritmo realiza a função de eliminação do ruído aditivo do sistema. A seção 5.3 apresenta o modelo de detecção multiusuário do sistema, para o caso em que não há ISI. A extensão vista no artigo [17] para a detecção multiusuário com ISI é explicada na seção 5.4. São também apresentadas algumas formas de se visualizar melhor o funcionamento do algoritmo, com um esquema gráfico e a apresentação em pseudo-código.

## 5.2 Separação do Subespaço do Sinal e do Ruído

Supõe-se, a princípio, que o canal de transmissão seja sem memória e que não dê ganho nos sinais transmitidos, ou seja, que o sinal recebido seja a soma dos sinais transmitidos e mais o ruído. Considera-se  $\mathbf{R}$  a matriz de correlação do sinal recebido  $\mathbf{r}$ , ou seja,

$$\mathbf{R} = E[\mathbf{r}\mathbf{r}^H] = E \left[ \left( \sum_{k=1}^K b_k \mathbf{c}_k + \mathbf{n} \right) \left( \sum_{l=1}^K b_l^* \mathbf{c}_l^H + \mathbf{n}^H \right) \right] \quad (5.1)$$

onde  $E[\cdot]$  representa o valor esperado e  $\mathbf{I}$  é a matriz identidade com as dimensões adequadas. Supondo que  $E[b_k b_l^*] = \delta(k, l)$  [17]:

$$\mathbf{R} = \sum_{k=1}^K \mathbf{c}_k \mathbf{c}_k^H + \sigma^2 \mathbf{I} = \mathbf{C}\mathbf{C}^H + \sigma^2 \mathbf{I} \quad (5.2)$$

usando a notação apresentada na seção 4.8,  $\mathbf{c}_k$  é a seqüência de espalhamento do usuário  $k$  e  $\mathbf{C}$  representa a matriz contendo em cada coluna a seqüência de espalhamento de um dos usuários. Faz-se então uma decomposição em autovalores e autovetores da matriz de correlação. Nota-se que alguns autovalores da decomposição da matriz de correlação estão relacionados ao espaço gerado pelas seqüências de espalhamento (recebendo o nome de subespaço do sinal, para seguir a nomenclatura dada por Wang e Poor em [17]). Já os demais autovalores estão relacionados ao subespaço gerado pelos vetores de ruído (recebendo o nome de subespaço do ruído). A separação entre esses autovalores será feita no decorrer desta seção, correspondendo à técnica conhecida como análise dos componentes principais (PCA - *principal component analysis*). A decomposição de  $\mathbf{R}$  em autovalores (matriz  $\mathbf{\Lambda}$ ) e autovetores (matriz  $\mathbf{U}$ ) é dada então por [17]:

$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H = \begin{bmatrix} \mathbf{U}_s & \mathbf{U}_n \end{bmatrix} \begin{bmatrix} \mathbf{\Lambda}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_n \end{bmatrix} \begin{bmatrix} \mathbf{U}_s^H \\ \mathbf{U}_n^H \end{bmatrix} \quad (5.3)$$

onde  $\mathbf{\Lambda}_s$  e  $\mathbf{U}_s$  representam respectivamente os autovalores e autovetores relacionados ao subespaço do sinal enquanto  $\mathbf{\Lambda}_n = \sigma^2 \mathbf{I}$  e  $\mathbf{U}_n$  representam os autovalores e autovetores relacionados ao subespaço do ruído. Como  $\mathbf{R}$  é uma matriz hermitiana,

seus autovetores são ortogonais entre si [18]. A partir de (5.2) pode-se desenvolver:

$$\mathbf{C}\mathbf{C}^H = \mathbf{R} - \sigma^2\mathbf{I} \quad (5.4)$$

$$= \mathbf{U} \begin{bmatrix} (\boldsymbol{\Lambda}_s - \sigma^2\mathbf{I}) & \mathbf{0} \\ \mathbf{0} & (\boldsymbol{\Lambda}_n - \sigma^2\mathbf{I}) \end{bmatrix} \mathbf{U}^H \quad (5.5)$$

$$= \mathbf{U} \begin{bmatrix} (\boldsymbol{\Lambda}_s - \sigma^2\mathbf{I}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{U}^H \quad (5.6)$$

e definindo-se  $(\boldsymbol{\Lambda}_0 = \boldsymbol{\Lambda}_s - \sigma^2\mathbf{I})$ , tem-se que

$$\mathbf{C}\mathbf{C}^H = \mathbf{U}_s(\boldsymbol{\Lambda}_s - \sigma^2\mathbf{I})\mathbf{U}_s^H = \mathbf{U}_s\boldsymbol{\Lambda}_0\mathbf{U}_s^H \quad (5.7)$$

Voltando-se à equação (5.2), multiplica-se essa equação à direita por  $\mathbf{v}$ , um autovetor de  $\mathbf{R}$ . Como a matriz referente ao ruído é a identidade escalada,  $\mathbf{v}$  também é seu autovetor, com autovalor  $\sigma^2$ . Com isso,

$$\lambda_R\mathbf{v} = \mathbf{C}\mathbf{C}^H\mathbf{v} + \sigma^2\mathbf{v} \quad (5.8)$$

sendo  $\lambda_R$  o autovalor de  $\mathbf{R}$  associado ao autovetor  $\mathbf{v}$ . Percebe-se que, como  $\mathbf{v}$  é autovetor de  $\mathbf{R}$  e da matriz do ruído, também será autovetor de  $\mathbf{C}\mathbf{C}^H$ . Como  $\mathbf{C}\mathbf{C}^H$  é positiva definida, então seus autovalores ( $\lambda_C$ ) são positivos. Então,

$$\lambda_R\mathbf{v} = \lambda_C\mathbf{v} + \sigma^2\mathbf{v} \quad (5.9)$$

$$\Rightarrow \lambda_R = \lambda_C + \sigma^2 \quad (5.10)$$

$$\Rightarrow \lambda_R > \sigma^2 \quad (5.11)$$

Com isso, os autovalores de  $\mathbf{R}$  associados ao subespaço do sinal são os  $K$  maiores autovalores, enquanto os demais, com valor  $\sigma^2$ , são associados ao subespaço do ruído, ou seja, os subespaços associados ao sinal e ruído podem ser separados através da decomposição em autovalores e autovetores da matriz de correlação do sinal recebido. Isso corresponde à PCA do vetor  $\mathbf{r}$ .

## 5.3 Detetor Multiusuário Cego Utilizado

Mantendo-se a suposição de que não haja ISI no sistema, o detetor multiusuário utilizado é definido como um filtro  $\mathbf{f}_j(k)$  por onde o sinal que vem dos



correlatores passa. Pelo fato de se estar usando transmissão BPSK a decisão baseada é no sinal algébrico da saída do filtro. O índice  $j$  do filtro está associado ao usuário alvo da detecção, enquanto o argumento  $k$  indica uma variação no tempo. Com isso, pode-se definir a estimativa do símbolo transmitido como [17]

$$\hat{b}_j(k) = \text{sgn}(\mathbf{f}_j^H(k)\mathbf{r}(k)) \quad (5.12)$$

onde  $\text{sgn}(\cdot)$  representa a função que dá o sinal algébrico de seu argumento.

O detetor utilizado obedece ao critério MMSE, assim como o detetor definido na seção 4.6.2. A métrica a ser minimizada por esse detetor assume a seguinte forma:

$$MSE(\mathbf{f}_j(k)) \triangleq E[|b_j(k) - \mathbf{f}_j^H(k)\mathbf{r}(k)|^2] \quad (5.13)$$

essa métrica é submetida à restrição  $\mathbf{f}_j^H \mathbf{c}_j = 1$  [17], para garantir a unicidade da solução.

A fim de se minimizar (5.13), utiliza-se o método de multiplicadores de Lagrange para se transformar um problema de minimização com restrições em outro equivalente, sem restrições. Com isso, usando o fato de que  $E[b_j^2(k)] = 1$  e  $E[\mathbf{r}(k)\mathbf{r}^H(k)] = \mathbf{R}$ , tem-se [17],

$$L(\mathbf{f}_j(k)) \triangleq MSE(\mathbf{f}_j(k)) - 2\lambda(\mathbf{f}_j^H(k)\mathbf{c}_j - 1) \quad (5.14)$$

$$= \mathbf{f}_j^H(k)E[\mathbf{r}(k)\mathbf{r}^H(k)]\mathbf{f}_j(k) - 2\mathbf{f}_j^H(k)E[b_j(k)\mathbf{r}(k)] + 1 - 2\lambda(\mathbf{f}_j^H(k)\mathbf{c}_j - 1) \quad (5.15)$$

$$= \mathbf{f}_j^H(k)\mathbf{R}\mathbf{f}_j(k) - (2 + 2\lambda)\mathbf{f}_j^H(k)\mathbf{c}_j + (1 + 2\lambda) \quad (5.16)$$

Derivando-se (5.16) em relação a  $\mathbf{f}_j(k)$  e igualando a zero, tem-se que

$$\mathbf{f}_j(k) = (1 + \lambda)\mathbf{R}^{-1}\mathbf{c}_j \quad (5.17)$$

fazendo-se mais uma vez a decomposição de  $\mathbf{R}$  em autovalores e autovetores e usando-se o fato de que os autovalores da inversa de uma matriz são o inverso dos autovalores da matriz original, chega-se a uma nova expressão para  $\mathbf{f}_j(k)$

$$\mathbf{f}_j(k) = (1 + \lambda)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H)\mathbf{c}_j + (1 + \lambda)\sigma^{-2}(\mathbf{U}_n\mathbf{U}_n^H)\mathbf{c}_j \quad (5.18)$$

Como  $\mathbf{c}_j$  pertence ao subespaço do sinal e é ortogonal ao subespaço do ruído [17],

$$\mathbf{f}_j(k) = (1 + \lambda)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^H)\mathbf{c}_j \quad (5.19)$$

Usando-se agora a restrição  $\mathbf{f}_j^H(k)\mathbf{c}_j = 1$  têm-se,

$$\mathbf{c}_j^H \mathbf{f}_j(k) = (1 + \lambda)(\mathbf{c}_j^H \mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H) \mathbf{c}_j \quad (5.20)$$

$$\Rightarrow 1 = (1 + \lambda)(\mathbf{c}_j^H \mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H) \mathbf{c}_j \quad (5.21)$$

$$\Rightarrow (1 + \lambda) = \frac{1}{\mathbf{c}_j^H \mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H} \mathbf{c}_j \quad (5.22)$$

substituindo esse resultado na equação (5.19), tem-se [17]

$$\mathbf{f}_j(k) = \frac{\mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{c}_j}{\mathbf{c}_j^H \mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{c}_j} \quad (5.23)$$

## 5.4 Detecção Multiusuário Cega com ISI

Nesta seção, o ambiente a ser considerado tem a questão de multipercorso resultando em ISI. O raciocínio matemático a ser utilizado é basicamente o mesmo das seções 5.2 e 5.3, com a modificação fundamental de que será usada a assinatura  $\mathbf{s}_j$  de cada usuário nas expressões que utilizavam a seqüência de espalhamento  $\mathbf{c}_j$ . A expressão do novo detetor multiusuário MMSE passa a ser então

$$\mathbf{f}_j(k) = \frac{\mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{s}_j}{\mathbf{s}_j^H \mathbf{U}_s \mathbf{\Lambda}_s^{-1} \mathbf{U}_s^H \mathbf{s}_j} \quad (5.24)$$

No entanto, ao contrário da expressão (5.23), em que todos as variáveis são conhecidas, na equação (5.24) não se conhece a assinatura  $\mathbf{s}_j$ , ou seja, a convolução entre a seqüência de espalhamento  $\mathbf{c}_j$  e a resposta ao impulso do canal  $\mathbf{h}_j$ . Como se conhecem as seqüências de espalhamento  $\mathbf{c}_j$ , falta apenas se descobrir qual a resposta ao impulso do canal de cada usuário  $\mathbf{h}_j$ . Para isso, considere a matriz  $\mathbf{S}_j$ , de dimensões  $(L - 1 + P \times L)$ , sendo  $L$  o tamanho da resposta ao impulso do canal e  $P$  o tamanho da seqüência de espalhamento.  $\mathbf{S}_j$  está associada ao usuário  $j$ , definida como:

$$\mathbf{S}_j = \begin{bmatrix} 0 & 0 & \dots & \mathbf{c}_j \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & \mathbf{c}_j & \dots & 0 \\ \mathbf{c}_j & 0 & \dots & 0 \end{bmatrix} \quad (5.25)$$

ou seja,  $\mathbf{S}_j$  tem uma estrutura de Hankel (constante nas anti-diagonais). Para cada um dos usuários tem-se que

$$\mathbf{s}_j = \mathbf{S}_j \mathbf{h}_j \quad (5.26)$$

Considere que  $\tilde{\mathbf{U}}_s$ , de dimensões  $(P + L - 1) \times K$  seja a estimativa da matriz de autovetores do subespaço do sinal. Então existe um vetor  $\mathbf{g}_j$  tal que  $\mathbf{s}_j = \tilde{\mathbf{U}}_s \mathbf{g}_j$ . Então  $\mathbf{h}_j$  pode ser determinado como a solução para a equação [17]

$$\mathbf{S}_j \mathbf{h}_j = \tilde{\mathbf{U}}_s \mathbf{g}_j \quad (5.27)$$

Observa-se que  $\mathbf{h}_j$  é unicamente determinado se e somente se a interseção entre o espaço das colunas de  $\mathbf{S}_j$  e o espaço das colunas de  $\tilde{\mathbf{U}}_s$  tiver dimensão 1. Se essa interseção de espaços tiver dimensão zero, o único vetor que satisfaz à equação (5.27) é o vetor nulo. Caso a dimensão dessa interseção for maior que 1, não se garante unicidade da solução. Define-se então a matriz  $\mathbf{D} \triangleq \begin{bmatrix} \tilde{\mathbf{U}}_s & \mathbf{S}_j \end{bmatrix}$  e o vetor  $\mathbf{w} \triangleq \begin{bmatrix} \mathbf{g}_j^H & -\mathbf{h}_j^H \end{bmatrix}^H$ . Como não se pode garantir que  $\tilde{\mathbf{U}}_s$  seja uma estimativa exata do subespaço do sinal, então não se pode garantir que a equação (5.27) tenha solução. Por isso define-se o problema de mínimos quadrados:

$$\min_{\mathbf{w}} \|\mathbf{D}\mathbf{w}\|^2 = \min_{\mathbf{w}} \mathbf{w}^H \mathbf{D}^H \mathbf{D} \mathbf{w} \quad (5.28)$$

Colocando-se a restrição  $\|\mathbf{w}\|^2 = 1$ , nota-se que a equação (5.28) torna-se equivalente à expressão do quociente de Rayleigh para a matriz  $\mathbf{D}^H \mathbf{D}$ . Sabe-se que, nesse caso, o valor de  $\mathbf{w}$  que minimiza a equação (5.28) é o autovetor associado ao menor autovalor de  $\mathbf{D}^H \mathbf{D}$ .

A partir do tratamento dado por [19] para transformar problemas de minimização com restrições em seus equivalentes sem restrições, chega-se a uma nova expressão para a função custo  $\mathcal{F}$  representada pela equação (5.28) com a restrição  $\|\mathbf{w}\|^2$  [17]:

$$\mathcal{F}(\mathbf{w}, \alpha) \triangleq \frac{1}{2} \mathbf{w}^H \mathbf{D}^H \mathbf{D} \mathbf{w} + \frac{\alpha}{4} (\mathbf{w} \mathbf{w}^H - 1)^2 \quad (5.29)$$

onde  $\alpha$  é uma constante positiva. Define-se  $\mathbf{B} = \mathbf{D}^H \mathbf{D}$ . Como  $\mathbf{B}$  é hermitiana, então a parte real de  $\mathbf{B}$  (denotada  $\mathbf{B}_R$ ) é simétrica e a parte imaginária de  $\mathbf{B}$  (denotada  $\mathbf{B}_I$ ) é anti-simétrica. Sendo a parte real de  $\mathbf{w}$  também denotada como

$\mathbf{w}_R$  e a parte imaginária de  $\mathbf{w}$  referenciada como  $\mathbf{w}_I$ , definem-se então

$$\mathbf{w}_m \triangleq \begin{bmatrix} \mathbf{w}_R \\ \mathbf{w}_I \end{bmatrix} \quad \mathbf{B}_m \triangleq \begin{bmatrix} \mathbf{B}_R & -\mathbf{B}_I \\ \mathbf{B}_I & \mathbf{B}_R \end{bmatrix} \quad (5.30)$$

e a equação (5.29) pode ser então reescrita como [17]

$$\mathcal{F}(\mathbf{w}_m, \alpha) \triangleq \frac{1}{2} \mathbf{w}_m^T \mathbf{B}_m \mathbf{w}_m + \frac{\alpha}{4} (\mathbf{w}_m \mathbf{w}_m^T - 1)^2 \quad (5.31)$$

Obtêm-se então o gradiente e o laplaciano de  $\mathcal{F}(\mathbf{w}_m, \alpha)$ :

$$\nabla \mathcal{F}(\mathbf{w}_m, \alpha) = \mathbf{B}_m \mathbf{w}_m + \alpha (\mathbf{w}_m^T \mathbf{w}_m - 1) \mathbf{w}_m \quad (5.32)$$

$$\nabla^2 \mathcal{F}(\mathbf{w}_m, \alpha) = \mathbf{B}_m + \alpha (\mathbf{w}_m^T \mathbf{w}_m - 1) \mathbf{I} + 2\alpha \mathbf{w}_m \mathbf{w}_m^T \quad (5.33)$$

Como a expressão  $\alpha (\mathbf{w}_m^T \mathbf{w}_m - 1)$  é um escalar, então o gradiente de  $\mathcal{F}(\mathbf{w}_m, \alpha)$  somente será zero quando o efeito de multiplicar  $\mathbf{B}_m$  por  $\mathbf{w}_m$  for o mesmo de multiplicar  $\mathbf{w}_m$  por  $-\alpha (\mathbf{w}_m^T \mathbf{w}_m - 1)$ , ou seja, se resultar num escalar. Com isso, o gradiente só será zero no caso em que  $\mathbf{w}_m$  for uma versão escalada de um autovetor da matriz  $\mathbf{B}_m$ , ou seja,  $\mathbf{w} = \|\mathbf{w}\| \mathbf{v}_B$ , onde  $\mathbf{v}_B$  representa um autovetor de  $\mathbf{B}$  de módulo unitário. Definindo-se  $\mathbf{v}_m$  contendo as partes real e imaginária de  $\mathbf{v}_B$ , tem-se que [17]:

$$\mathbf{0} = \nabla \mathcal{F}(\|\mathbf{w}\| \mathbf{v}_B, \alpha) = \mathbf{B}_m \|\mathbf{w}\| \mathbf{v}_m + \alpha (\|\mathbf{w}\|^2 - 1) \|\mathbf{w}\| \mathbf{v}_m \quad (5.34)$$

$$= \lambda_B \|\mathbf{w}\| \mathbf{v}_m + \alpha (\|\mathbf{w}\|^2 - 1) \|\mathbf{w}\| \mathbf{v}_m \quad (5.35)$$

onde  $\lambda_B$  corresponde a um autovalor de  $\mathbf{B}$ . Rearrumando-se a equação conclui-se que os pontos de derivada zero de  $\mathcal{F}(\|\mathbf{w}\| \mathbf{v}_B, \alpha)$  são os pontos  $\mathbf{w}_m$  tais que

$$\|\mathbf{w}_m\| = \sqrt{1 - \frac{\lambda_B}{\alpha}} \Rightarrow \quad \mathbf{w}_m = \sqrt{1 - \frac{\lambda_B}{\alpha}} \mathbf{v}_B \quad (5.36)$$

Com isso, o valor do laplaciano nesse ponto é dado por (5.36)

$$\nabla^2 \mathcal{F} \left( \sqrt{1 - \frac{\lambda_B}{\alpha}} \mathbf{v}_B, \alpha \right) = \mathbf{B}_m - \lambda_B \mathbf{I} + 2(\alpha - \lambda_B) \mathbf{v}_m \mathbf{v}_m^T \quad (5.37)$$

É mostrado em [17] que a função  $\nabla^2 \mathcal{F}(\cdot)$  só é positiva semi-definida no ponto  $(\sqrt{1 - \frac{\lambda_B}{\alpha}} \mathbf{v}_B, \alpha)$  se e somente se  $\lambda_B$  for o menor autovalor de  $\mathbf{B}$ . Isso garante que esse ponto de gradiente zero é ponto de mínimo. Caso o autovalor não seja o menor,

o laplaciano nesse ponto é indefinido, correspondendo então a um ponto de sela. Com isso, o ponto que minimiza a função  $\mathcal{F}(\mathbf{w}, \alpha)$  é dado por [17]:

$$\mathbf{w}_{opt} = \sqrt{1 - \frac{\lambda_{B \min}}{\alpha}} \mathbf{v}_B \quad (5.38)$$

É garantido então que um algoritmo baseado em LMS convirja para o ponto dado pela equação (5.38). Normalmente, escolhe-se o valor da constante  $\alpha$  bem maior que o menor autovalor de  $\mathbf{B}$ , a fim de que a norma da solução seja próxima de 1.

A fim de se obter o valor de  $\mathbf{w}(k)$  adaptativamente, utiliza-se um algoritmo do tipo gradiente. Esses algoritmos usam a direção do gradiente para se achar um valor ótimo para  $\mathbf{w}(k)$ :

$$\mathbf{w}(k) = \mathbf{w}(k-1) - \mu \nabla \mathcal{F}(\mathbf{w}(k), \alpha) \quad (5.39)$$

onde  $\mu$  representa o passo de atualização. Já o gradiente da função custo é obtido através da equação (5.32). Com isso, é escolhido um algoritmo adaptativo do tipo LMS com restrições com a seguinte atualização dos coeficientes:

$$\mathbf{w}(k) = \mathbf{w}(k-1) - \mu [\mathbf{B}(k-1)\mathbf{w}(k-1) + \alpha(\|\mathbf{w}(k-1)\|^2 - 1)\mathbf{w}(k-1)] \quad (5.40)$$

Como  $\mathbf{w} \triangleq [\mathbf{g}_j^H - \mathbf{h}_j^H]^H$ , então, progressivamente, têm-se estimativas cada vez melhores da resposta ao impulso do canal utilizado para o usuário em questão ( $j$ -ésimo usuário). Essa estimativa é usada conjuntamente com a seqüência de espalhamento  $\mathbf{c}_j$  para se obter a assinatura  $\mathbf{s}_j$  do usuário em questão, chegando-se ao detetor multiusuário para o caso de multipercurso com ISI através da equação (5.24).

A matriz  $\mathbf{B}(k) = [\tilde{\mathbf{U}}_s^H(k) \quad \mathbf{S}_j^H]^H [\tilde{\mathbf{U}}_s(k)\mathbf{S}_j]$  precisa, em cada instante de tempo, da estimativa  $\tilde{\mathbf{U}}_s(k)$  do subespaço do sinal. Essa estimativa pode ser obtida a partir da decomposição em autovalores e autovetores da estimativa da matriz de correlação  $\mathbf{R}(k)$  do sinal recebido.  $\mathbf{R}(k)$ , por sua vez, é calculada recursivamente através da seguinte expressão:

$$\mathbf{R}(k) = \frac{1}{k} [(k-1)\mathbf{R}(k-1) + \mathbf{r}(k)\mathbf{r}^H(k)] \quad (5.41)$$

Essa expressão é útil para o caso de ambientes estacionários. No caso de ambientes não-estacionários, deve-se usar um fator de esquecimento  $\lambda_f$  fixo (independente de qual iteração está), de maneira que a expressão se torna:

$$\mathbf{R}(k) = \lambda_f \mathbf{R}(k-1) + (1 - \lambda_f) \mathbf{r}(k) \mathbf{r}^H(k) \quad (5.42)$$

Outros métodos de se encontrar o subespaço do sinal podem ser utilizados, com complexidade computacional menor que o da decomposição em autovalores e autovetores. Por exemplo, ainda em [17], Wang e Poor descrevem o método PASTd (*Projection Approximation Subspace Tracking*), primeiramente apresentado por Yang em [20]. Na implementação utilizada nesta tese tentou-se evitar esses métodos, a fim de que suas aproximações não afetassem os resultados quanto a taxa de erro de símbolo em relação a razão sinal-ruído, que são o foco nesta tese. Além disso, o próprio método revelou-se custoso computacionalmente, sendo que seu ganho de velocidade em relação ao uso de SVD no ambiente de simulação utilizado se revelou bem pequeno. O cálculo do subespaço do sinal foi feito através de estimativa de  $\mathbf{U}_s$  através da equação (5.41).

Um esquema com o funcionamento do algoritmo pode ser visto na figura 5.1. Um resumo do sistema descrito neste capítulo pode ser visto no quadro Algoritmo 5.1.

A separação entre sinal e ruído é feita através de análise dos componentes principais. A seguir é feita uma estimativa adaptativa do canal do usuário em

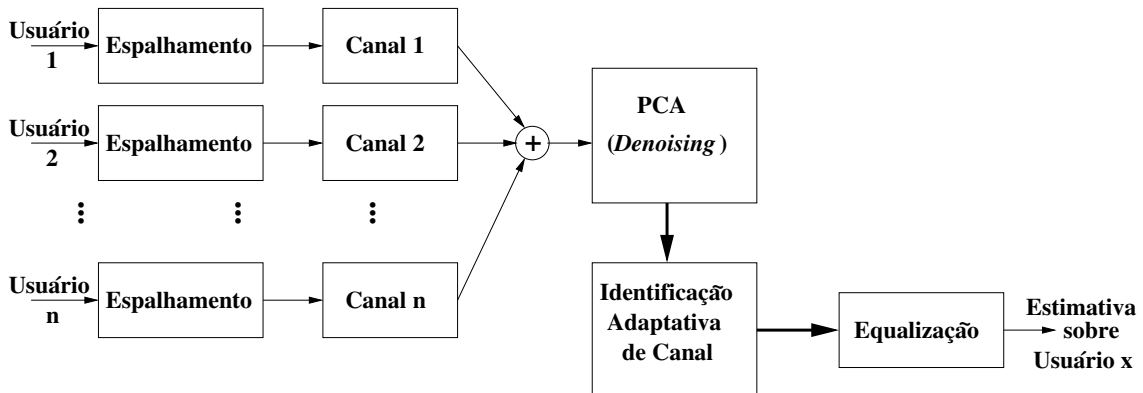


Figura 5.1: Esquema de funcionamento do algoritmo de subespaço.

questão, através da atualização contida na equação (5.40). Depois de os coeficientes do filtro  $\mathbf{w}(k)$  convergirem, usa-se a parte desses coeficientes que contêm a estimativa do canal, como visto na equação  $\mathbf{w} \triangleq [\mathbf{g}_j^H - \mathbf{h}_j^H]^H$ . Por fim, usa-se essa estimativa de  $\mathbf{h}_j$  para se achar a assinatura  $\mathbf{s}_j$  do usuário desejado, substituindo esse valor em (5.24) a fim de se chegar ao detetor multiusuário com ISI. Nota-se que o esquema da figura 5.1 está representando o esquema de detecção com equalização para um dos usuários apenas. Para se identificar a seqüência transmitida para todos os usuários, deve-se usar  $K$  blocos de identificação de canal e detecção multiusuário.

## 5.5 Questões sobre o Algoritmo

No modelo de CDMA colocado no artigo de Wang e Poor [17], a transmissão no *uplink* é feita de acordo com a seguinte equação:

$$r(j, x) = \sum_{k=1}^K b_k(j) \sum_{l=1}^L c_k(x-l)h_k(l) + \mathbf{n}(j) \quad (5.43)$$

Isso é diferente do modelo de CDMA colocado pela equação (4.12). Em (5.43), não há consideração de ISI pois cada conjunto de chips recebido depende de apenas um símbolo transmitido para cada usuário. Assim, o modelo contido no artigo [17] não corresponde exatamente ao problema proposto, que é o de desenvolver um detetor multiusuário com eliminação de ISI. Ao se simular os algoritmos, foram usadas as duas situações, tanto o modelo da equação (4.12) quanto o da equação (5.43).

Como será visto no capítulo 8, a implementação usando a equação (5.43) tem resultados bastante diferentes da implementação usando a equação (4.12). Serão discriminadas as duas implementações ao se realizar as comparações.

## 5.6 Conclusões

O objetivo deste capítulo é apresentar a teoria básica que envolve a família de métodos de equalização cega usando subespaço através do método descrito em [17].

O método de subespaço é o primeiro dos três tipos de algoritmos de equalização cega usando estatísticas de segunda ordem que serão apresentados nesta tese. Como foi visto, a principal motivação de desenvolvimento da equalização cega com subespaço são modificações em algoritmos de detecção multiusuário cega, como o da seção 5.3.

A idéia principal do algoritmo apresentado é descobrir o vetor que é a solução de um sistema linear baseado no subespaço do sinal encontrado. Essa equação, como foi vista, é expressa como  $\|\mathbf{D}\mathbf{w}\|^2 = 0$ . Tenta-se chegar à solução de mínimos quadrados através de um algoritmo adaptativo.

A seção 5.5 questiona alguns pontos de modelagem do CDMA apresentados no artigo [17], em particular as equações referentes ao sinal recebido. É proposta uma nova modelagem, mais fiel ao sistema encontrado na realidade, levando-se em conta a ISI.

A importância deste capítulo na tese está em descrever um dos algoritmos implementados na análise comparativa, apresentando-o tanto numa descrição textual quanto sob forma gráfica e pseudo-código, mais voltados para implementação.



## 5.6. CONCLUSÕES

---

### Algoritmo 5.1 Algoritmo de Equalização Cega por Subespaço

---

```
Para j=1:NúmeroDeUsuários,
    cj = GeraEspalhamento('CódigoDeEspalhamento',j);
    Cmatj = GeraMatrizEspalhamento(cj);
    bj = GeraSinal(NúmeroDeSímbolos);
    sj = Convolução(cj,hj);
    hj = GeraCanal(ComprimentoDoCanal);
    r = r + Convolução(bj,sj);
    Potencia_Sinal_dB(j) = MedePotenciaSinalEmDB(Convolução(bj,sj));
    Potencia_Ruido_dB(j) = SNR_Escolhida_dB - Potencia_Sinal_dB;
    r = r + GeraRuído(Potencia_Ruido_dB(j));
fim Para

Para j=1:NúmeroDeUsuários,
    Para k = 1:NúmeroDeSímbolos,
        autocorrelação_est(k) = [(k-1)*autocorrelação_est(k-1) + rrH]/k;
        Us = DecomposiçãoEmAutovetores(autocorrelação_est(k));
        λs = DecomposiçãoEmAutovalores(autocorrelação_est(k));
        D(k) = [Us Cmatj];
        w(k,:) = w(k-1,:) - μ[DH(k)D(k) + α(||w(k-1,:)||2 - 1)w(k-1,:)]
    fim Para

    CanalEstimado(:,k) = -w(k,NúmeroDeUsuários+1:end)
    mj = 
$$\frac{\mathbf{U}_s \Lambda_s^{-1} \mathbf{U}_s^H \mathbf{s}_j}{\mathbf{s}_j^H \mathbf{U}_s \Lambda_s^{-1} \mathbf{U}_s^H \mathbf{s}_j}$$

fim Para

Para j=1:NúmeroDeUsuários,
    Para k=1:NúmeroDeSímbolos,
        símbolo_estimado(j,k) = mjH r;
    fim Para
fim Para
```

---

# Capítulo 6

## Equalização Cega Usando Variância Mínima de Saída

### 6.1 Introdução

Na modelagem do sistema CDMA feita nesta tese, os canais não produzem ganhos ao sinal de entrada, o que evita disparidades nos ganhos dos diferentes usuários. Na prática, detetores multiusuário MMSE, visto na subseção 4.6.2, dependem do conhecimento da amplitude de cada sinal na saída dos canais. Isso dificulta o desenvolvimento de um detetor multiusuário cego baseado no conceito MMSE.

Neste capítulo é apresentado um método para detecção multiusuário cega para modelo de transmissão assíncrona, em sistemas com interferência intersimbólica, o que é equivalente ao problema da equalização cega multiusuário. O problema apresentado é equivalente ao MMSE, usando como função objetivo a variância mínima de saída (MOV - *minimum output variance*). A implementação utilizada está baseada no método desenvolvido por Xu e Tsatsanis apresentado em [21].

A seção 6.2 mostra a relação existente entre o detetor cego MMSE e o equalizador cego que usa MOV. Na seção 6.3 será apresentado o critério de mini-

mização de variância com restrições utilizado neste algoritmo. A estrutura adaptativa conhecida como cancelador de bandas laterais generalizado (GSC - *generalized sidelobe canceller*), que serve exatamente a um problema de minimização com restrições, é introduzida na seção 6.4. Por fim, na seção 6.5, são mostrados os algoritmos adaptativos utilizados para a atualização dos coeficientes dos filtros em cada sub-banda da estrutura GSC.

## 6.2 Equivalência entre MOV e MMSE

Considere um detetor multiusuário cego  $\mathbf{f}_j$  para sistemas com ISI. O índice  $j$  associa o detetor ao usuário  $j$ . Considera-se que esse detetor seja tal que  $\mathbf{f}_j^H \mathbf{c}_j = 1$ , ou seja, seu produto escalar com a seqüência de espalhamento  $\mathbf{c}_j$  seja 1. Considere a representação de  $\mathbf{f}_j$ :

$$\mathbf{f}_j = \mathbf{c}_j + \mathbf{q}_j \quad (6.1)$$

Como visto em [22], essa representação é canônica, ou seja, que utiliza bases ortogonais. Considera-se  $\mathbf{r}$  o vetor recebido, o detetor MOV é aquele que minimiza a métrica:

$$MOV(\mathbf{f}_j) = E[|\mathbf{f}_j^H \mathbf{r}|^2] \quad (6.2)$$

A métrica MSE para esse detetor é dada por

$$\begin{aligned} E[|b_j(k) - \mathbf{f}_j^H \mathbf{r}|^2] &= E[|b_j(k) - (\mathbf{c}_j^H + \mathbf{q}_j^H) \mathbf{r}|^2] \\ &= E[|b_j(k)|^2] + E[|(\mathbf{c}_j^H + \mathbf{q}_j^H) \mathbf{r}|^2] - 2E[|b_j(k)(\mathbf{c}_j^H + \mathbf{q}_j^H) \mathbf{r}|] \\ &= 1 + MOV(\mathbf{f}_j) - 2E[|b_j(k)(\mathbf{c}_j^H + \mathbf{q}_j^H) \mathbf{r}|] \end{aligned} \quad (6.3)$$

Usando-se o fato de  $E[\mathbf{f}_j^H b_j(k) \mathbf{r}] = \mathbf{f}_j^H E[b_j(k) \mathbf{r}] = \mathbf{f}_j^H \mathbf{c}_j = 1$ :

$$E[|b_j(k) - \mathbf{f}_j^H \mathbf{r}|^2] = MOV(\mathbf{f}_j) - 1 \quad (6.4)$$

No caso específico de se usar canais com ganhos unitários, esse resultado coincide com o mostrado em [22]. Com isso, verifica-se que o critério de minimização da variância de saída é equivalente à minimização do erro médio quadrático, sob a condição  $\mathbf{f}_j^H \mathbf{c}_j = 1$ .

A grande vantagem do uso da métrica da variância de saída é que sua expressão não depende dos ganhos dos canais, possibilitando com isso uma implementação cega no caso de canais com ganhos desconhecidos.

## 6.3 Critério de Variância Mínima de Saída com Restrições

Considerando a matriz de espalhamento  $\mathbf{C}_j$ , com tamanho  $(P + Q - 1) \times Q$  (onde  $P$  é o tamanho da seqüência de espalhamento e  $Q$  é o comprimento do canal) definida por

$$\mathbf{C}_j = \begin{bmatrix} \mathbf{c}_j & 0 & \dots & 0 \\ 0 & \mathbf{c}_j & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{c}_j \end{bmatrix} \quad (6.5)$$

A matriz de correlação do sinal recebido é estimada por  $\mathbf{R} = \mathbf{r}\mathbf{r}^H$ . Então a função custo a ser minimizada é definida por [21]:

$$\mathcal{F}(\mathbf{f}_j) = \mathbf{f}_j^H \mathbf{R} \mathbf{f}_j \quad \text{restrita a} \quad \mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j \quad (6.6)$$

onde  $\mathbf{g}_j$  é um vetor de restrições de tamanho  $Q$ . O uso de restrições é necessário, pois, caso contrário, a minimização da variância levaria à solução nula [23]. Na ausência de multipercursos, a restrição envolve apenas um produto escalar, sendo expressa por  $\mathbf{f}_j^H(k)\mathbf{c}_j = \alpha$ . A presença de multipercursos gera um conjunto de restrições na equação (6.6). Utilizando multiplicadores de Lagrange, a função descrita em (6.6) se torna:

$$\mathcal{F}(\mathbf{f}_j) = \mathbf{f}_j^H \mathbf{R} \mathbf{f}_j + \boldsymbol{\lambda}^H (\mathbf{C}_j^H \mathbf{f}_j - \mathbf{g}_j) \quad (6.7)$$

Derivando-se em relação a  $\mathbf{f}_j$  e igualando ao vetor nulo:

$$2\mathbf{R}\mathbf{f}_{j,opt} + \mathbf{C}_j \boldsymbol{\lambda} = \mathbf{0} \quad (6.8)$$

$$\Rightarrow 2\mathbf{C}_j^H \mathbf{R}^{-1} \mathbf{R} \mathbf{f}_{j,opt} + \mathbf{C}_j^H \mathbf{R}^{-1} \mathbf{C}_j \boldsymbol{\lambda} = \mathbf{0} \quad (6.9)$$

$$\Rightarrow 2\mathbf{g}_j + \mathbf{C}_j^H \mathbf{R}^{-1} \mathbf{C}_j \boldsymbol{\lambda} = \mathbf{0} \quad (6.10)$$

$$\Rightarrow \boldsymbol{\lambda} = -2(\mathbf{C}_j^H \mathbf{R}^{-1} \mathbf{C}_j)^{-1} \mathbf{g}_j \quad (6.11)$$

Substituindo o valor em (6.8), chega-se a:

$$2\mathbf{R}\mathbf{f}_{j,opt} - 2\mathbf{C}_j(\mathbf{C}_j^H\mathbf{R}^{-1}\mathbf{C}_j)^{-1}\mathbf{g}_j = \mathbf{0} \quad (6.12)$$

$$\Rightarrow \mathbf{f}_{j,opt} = \mathbf{R}^{-1}\mathbf{C}_j(\mathbf{C}_j^H\mathbf{R}^{-1}\mathbf{C}_j)^{-1}\mathbf{g}_j \quad (6.13)$$

Com isso, o valor da variância mínima encontrada para o sistema é de [21]:

$$\mathcal{F}_{opt} = \mathbf{f}_{j,opt}^H\mathbf{R}\mathbf{f}_{j,opt} = \mathbf{g}_j^H(\mathbf{C}_j^H\mathbf{R}^{-1}\mathbf{C}_j)^{-1}\mathbf{g}_j \quad (6.14)$$

Ao detector multiusuário baseado na métrica MOV com restrições ligadas aos multipercursos dá-se o nome de CMV (*constrained minimum variance*).

## 6.4 Cancelador de Bandas Laterais Generalizado

O problema de minimização de uma função custo com restrições nos coeficientes do filtro adaptativo pode ser tratado quebrando-se o problema em duas otimizações sem restrições. Essa é a idéia básica do algoritmo cancelador de bandas laterais generalizado (GSC - *generalized sideband canceller*) [8] [24]. Considere o problema de minimização cuja função custo  $\mathcal{F}$  e a restrição dos coeficientes do filtro  $\mathbf{f}_j$  seja dada pela equação (6.6).

A estrutura GSC que faz a atualização dos coeficientes de  $\mathbf{f}_j$  é mostrada na figura 6.1. A matriz  $\mathbf{C}_{j,n}$  é definida de maneira que  $\mathbf{C}_{j,n}^H\mathbf{C}_j = \mathbf{0}$ , ou seja, as colunas de  $\mathbf{C}_{j,n}$  pertencem ao espaço nulo da matriz de restrições  $\mathbf{C}_j$ . Com isso, cada vetor recebido passa por sub-bandas ortogonais entre si, a sub-banda superior trata do problema de otimização principal sem as restrições, enquanto a sub-banda inferior trata exclusivamente das restrições do sistema. Define-se então um vetor  $\mathbf{z}_j$  que tem parte de seus coeficientes em cada sub-banda, de maneira que

$$\mathbf{z}_j = \begin{bmatrix} \mathbf{w}_{j,u} \\ \mathbf{w}_{j,l} \end{bmatrix} \quad (6.15)$$

Como pode ser observado a partir da figura 6.1, a estimativa do símbolo

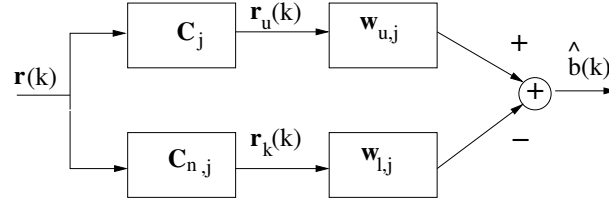


Figura 6.1: Esquema de funcionamento do GSC.

transmitido pelo  $j$ -ésimo usuário é dada por [8]:

$$\hat{b}_j(k) = \mathbf{w}_{j,u}^H \mathbf{C}_j^H \mathbf{r}(k) - \mathbf{w}_{j,l}^H \mathbf{C}_{j,n}^H \mathbf{r}(k) \quad (6.16)$$

$$= (\mathbf{C}_j \mathbf{w}_{j,u} - \mathbf{C}_{j,n} \mathbf{w}_{j,l})^H \mathbf{r}(k) \quad (6.17)$$

$$= (\mathbf{Tz}_j)^H \mathbf{r}(k) \quad (6.18)$$

onde  $\mathbf{T} = [\mathbf{C}_j \quad -\mathbf{C}_{j,n}]$ . A partir da equação (6.6) percebe-se que a restrição na otimização do sistema é dada por

$$\mathbf{C}_j^H \mathbf{Tz}_j = \mathbf{g}_j \quad (6.19)$$

Como, pela equação (6.6),  $\mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j$ , então (6.19) pode ser reescrita como [8]

$$\mathbf{f}_j = \mathbf{Tz}_j \quad (6.20)$$

ou seja,  $\mathbf{T}$  representa uma transformação linear entre o filtro de coeficientes  $\mathbf{z}_j$  que efetivamente é atualizado na estrutura GSC e o filtro de coeficientes  $\mathbf{f}_j$  do problema original, sobre o qual estão aplicadas as restrições. Expande-se a equação (6.19) a fim de se verificar de maneira mais detalhada sua influência no sistema. Usando a relação  $\mathbf{C}_j^H \mathbf{C}_{j,n} = \mathbf{0}$ , tem-se que:

$$\begin{aligned} \mathbf{C}_j^H \mathbf{Tz}_j &= \mathbf{g}_j \\ \Rightarrow \mathbf{C}_j^H \mathbf{f}_j &= \mathbf{g}_j \\ \Rightarrow \mathbf{C}_j^H \mathbf{C}_j \mathbf{w}_{j,u} - \mathbf{C}_j^H \mathbf{C}_{j,n} \mathbf{w}_{j,l} &= \mathbf{g}_j \\ \Rightarrow \mathbf{C}_j^H \mathbf{C}_j \mathbf{w}_{j,u} &= \mathbf{g}_j \end{aligned} \quad (6.21)$$

Com isso,  $\mathbf{w}_{j,u}$  deve ser inicializado de forma que [8]

$$\mathbf{w}_{j,u} = (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j \quad (6.22)$$

## 6.5 Atualização Adaptativa da Versão GSC

A partir da figura 6.1, observa-se que a expressão do equalizador  $\mathbf{f}_j$  referente ao  $j$ -ésimo usuário é dada por [21]:

$$\mathbf{f}_j = \mathbf{C}_j \mathbf{w}_{j,u} - \mathbf{C}_{j,n} \mathbf{w}_{j,l} \quad (6.23)$$

Como visto na equação (6.6), a restrição do sistema é expressa por  $\mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j$ . Multiplicando-se a equação (6.23) à esquerda por  $\mathbf{C}_j^H$ , e lembrando que  $\mathbf{C}_j^H \mathbf{C}_{j,n} = \mathbf{0}$ , obtém-se

$$\mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j = \mathbf{C}_j^H \mathbf{C}_j \mathbf{w}_{j,u} - \mathbf{C}_j^H \mathbf{C}_{j,n} \mathbf{w}_{j,l} \quad (6.24)$$

$$\Rightarrow \mathbf{g}_j = \mathbf{C}_j^H \mathbf{C}_j \mathbf{w}_{j,u} \quad (6.25)$$

$$\Rightarrow \mathbf{w}_{j,u} = (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j \quad (6.26)$$

Com isso, a equação (6.23) pode ser reescrita como:

$$\mathbf{f}_j = \mathbf{C}_j^H (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j - \mathbf{C}_{j,n} \mathbf{w}_{j,l} \quad (6.27)$$

Essa parametrização do equalizador em função de  $\mathbf{w}_{j,l}$  e  $\mathbf{g}_j$  permitirá que se tenha uma implementação adaptativa em que se atualizam os coeficientes dos filtros representados por esses dois vetores.

O conjunto de restrições  $\mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j$  já está contido na equação (6.27). Basta então encontrar uma nova função custo  $\mathcal{F}(\mathbf{f}_j)$  de acordo com a função custo original  $\mathcal{F}(\mathbf{f}_j) = \mathbf{f}_j^H \mathbf{R} \mathbf{f}_j$ , substituindo-se  $\mathbf{f}_j$  pela expressão encontrada em (6.27). Assim, [21]

$$\begin{aligned} \mathcal{F}(\mathbf{f}_j) &= \mathbf{g}_j^H (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_j (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j + \mathbf{w}_{j,l}^H \mathbf{C}_{j,n}^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l} \\ &\quad - \mathbf{g}_j^H (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l} - \mathbf{w}_{j,l}^H \mathbf{C}_{j,n}^H \mathbf{R} (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j \end{aligned} \quad (6.28)$$

De uma maneira geral, pode-se dizer que a minimização da variância tem como objetivo a supressão das perturbações do sinal desejado. No entanto, após essa supressão ser feita, é interessante que se maximize a potência do sinal de interesse.

Como pode ser visto na Figura 6.1, o algoritmo GSC divide o sistema em duas sub-bandas, operando em subespaços ortogonais entre si, pois  $\mathbf{C}_j^H \mathbf{C}_{j,n} = \mathbf{0}$ .

O subespaço formado pelas colunas de  $\mathbf{C}_j$  é conhecido como subespaço do sinal, já que contém versões atrasadas da seqüência de espalhamento do sinal para o usuário em questão. Deve-se então minimizar a contribuição da sub-banda ortogonal ao subespaço do sinal e maximizar a contribuição da sub-banda relativa ao sinal. Com isso, a atualização do filtro relativo a  $\mathbf{w}_{j,l}$  será feita a partir de um vetor com mesma direção que o gradiente da função custo, mas com sentido contrário ao do gradiente, buscando assim a minimização da função custo. Contrariamente, a atualização de  $\mathbf{g}_j$  deve ser feita a partir de um vetor com direção e sentido iguais ao do vetor gradiente da função custo. Então, as equações de atualização adaptativa são dadas por:

$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) - \mu_w \nabla_{w_{j,l}} \mathcal{F}(\mathbf{f}_j) \quad (6.29)$$

$$\mathbf{g}_j(k+1) = \mathbf{g}_j(k) + \mu_g \nabla_{g_j} \mathcal{F}(\mathbf{f}_j) \quad (6.30)$$

onde  $\mu_w$  e  $\mu_g$  representam os passos de atualização e  $\nabla_w$  e  $\nabla_g$  são os operadores gradientes em relação a  $\mathbf{w}_{j,l}$  e  $\mathbf{g}_j$ , respectivamente. O vetor  $\mathbf{w}_{j,l}$  tem tamanho  $P-1$ .

Calculando-se então o valor dos gradientes da função  $\mathcal{F}(\mathbf{f}_j)$  [21]:

$$\nabla_w \mathcal{F}(\mathbf{f}_j) = \mathbf{C}_{j,n}^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l} - \mathbf{C}_{j,n}^H \mathbf{R} \mathbf{C}_j (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j \quad (6.31)$$

$$\nabla_g \mathcal{F}(\mathbf{f}_j) = (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_j (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j - (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l} \quad (6.32)$$

Substituindo então a expressão do gradiente (6.31) na equação (6.29) e a expressão (6.32) na equação (6.30), chega-se às expressões finais de atualizações dos filtros.

$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) - \mu_w [\mathbf{C}_{j,n}^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l}(k) - \mathbf{C}_{j,n}^H \mathbf{R} \mathbf{C}_j (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j(k)] \quad (6.33)$$

$$\mathbf{g}_j(k+1) = \mathbf{g}_j(k)$$

$$+ \mu_g [(\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_j (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{g}_j(k) - (\mathbf{C}_j^H \mathbf{C}_j)^{-1} \mathbf{C}_j^H \mathbf{R} \mathbf{C}_{j,n} \mathbf{w}_{j,l}(k)] \quad (6.34)$$

A fim de que o sistema não convirja para a solução trivial  $\mathbf{g}_j = \mathbf{0}$ , é aplicada a seguinte normalização sobre o vetor  $\mathbf{g}_j(k+1)$  após cada iteração [21]:

$$\text{Para } \mathbf{g}_j(k+1) \text{ é atribuído } \frac{\mathbf{g}_j(k+1)}{\|\mathbf{g}_j(k+1)\|} \quad (6.35)$$

Um esquema geral de funcionamento do sistema pode ser visto na figura 6.2. Observa-se que, diferentemente do algoritmo descrito no capítulo 5, o MOV



não tem uma etapa específica de eliminação de ruído, outra de supressão de ISI e uma final de separação dos usuários. A eliminação do ruído é realizada pela própria estrutura do GSC em duas sub-bandas, ortogonais entre si, e minimizando a função custo na banda que não contém o subespaço do sinal. A supressão de ISI é realizada pelas restrições  $\mathbf{C}_j^H \mathbf{f}_j = \mathbf{g}_j$ , enquanto a separação dos usuários é feita pela maximização da função custo na sub-banda referente ao subespaço gerado por versões atrasadas da seqüência de espalhamento do usuário em questão. Uma visão geral do algoritmo é dada a partir do quadro Algoritmo 6.1.

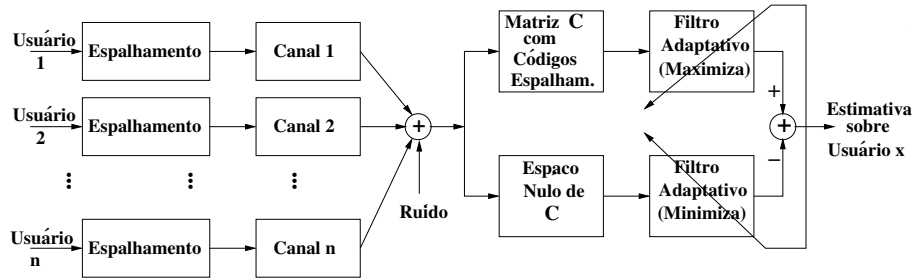


Figura 6.2: Esquema de funcionamento do algoritmo de equalização cega por minimização da variância de saída.

## 6.6 Questões sobre o Algoritmo

Na modelagem do sinal recebido contida no artigo [21] de Xu e Tsatsanis que descreve o algoritmo CMV implementado neste capítulo, considera-se que o sinal recebido devido ao usuário desejado  $j$  modelado da seguinte maneira:

$$\mathbf{r}_j(l) = \mathbf{s}_j b_j(l) + \tilde{\mathbf{s}}_j b_j(l-1) + \tilde{\tilde{\mathbf{s}}}_j b_j(l+1) \quad (6.36)$$

onde  $\tilde{\mathbf{s}}_j$  e  $\tilde{\tilde{\mathbf{s}}}_j$  referem-se às assinaturas atrasadas do usuário em questão, de maneira a incluir a ISI no símbolo atual. Mais especificamente,

$$\tilde{\mathbf{s}}_j = \begin{bmatrix} s_j(P-d_j) \\ \vdots \\ s_j(P+Q-1) \\ \mathbf{0} \end{bmatrix} \quad \tilde{\tilde{\mathbf{s}}}_j = \begin{bmatrix} \mathbf{0} \\ s_j(0) \\ \vdots \\ s_j(P+Q-1-d_j) \end{bmatrix} \quad (6.37)$$

onde  $d_j$  é o atraso em número inteiro de períodos do usuário  $j$ .

O sinal recebido dos demais usuários é dado pela seguinte equação:

$$\mathbf{r}_x(l) = \tilde{\mathbf{s}}_x b_x(l-1) + \tilde{\mathbf{s}}_x b_x(l+1) \quad (6.38)$$

No entanto, essa modelagem faz o usuário desejado ter uma potência maior que os demais na medida em que sua expressão inclui três termos enquanto a dos demais usuários só inclui dois.

No artigo [21], o algoritmo MOV descrito é considerado apenas para o caso em que as formas de onda que interferem com o usuário em questão estão defasadas em relação à forma de onda do usuário desejado por um atraso maior que o comprimento dos canais. Com isso, a modelagem do sinal recebido dada pelas equações (6.36) e (6.38) está de acordo com a descrição. No entanto, essa modelagem dá ao sinal desejado uma grande vantagem em relação à separação multiusuário que não está presente nos demais algoritmos.

Optou-se, nesta tese, por se fazer uma descrição do algoritmo MOV usando uma modelagem em que todos os usuários fossem transmitidos de acordo com a equação (6.36), formando o sinal total através da soma dos sinais de todos os usuários. Com isso, uma modelagem mais precisa do CDMA é feita. No capítulo 8 serão comparados os resultados das duas implementações, a fim de se verificar melhor qual o efeito de se usar a modelagem contida no artigo de Xu e Tsatsanis com a modelagem proposta nesta tese.

## 6.7 Conclusões

Este capítulo apresenta mais uma família de métodos de equalização cega, desta vez o método MOV. É o segundo de três tipos de algoritmos apresentados na tese que usam estatísticas de segunda ordem para equalização cega em CDMA. Neste capítulo, a equalização é feita minimizando-se a variância das estimativas da seqüência transmitida. Após essa minimização, que tem como objetivo a eliminação

das perturbações aditivas do sinal, maximiza-se a contribuição do sinal do usuário em questão na mistura recebida. Isso é facilmente realizado pela separação em duas bandas proporcionada pela estrutura do GSC.

A contribuição deste capítulo é a de apresentar um dos algoritmos implementados com detalhes, colocá-lo com a mesma formatação e notação do algoritmo de subespaço (apresentado no capítulo 5) e explorar suas relações com o detetor MMSE.

Assim como no capítulo anterior, verifica-se mais uma vez o uso da separação do subespaço do sinal e do ruído para se fazer a supressão do ruído. No entanto, desta vez, isso é realizado de maneira mais natural através da estrutura adaptativa utilizada, evitando-se o uso de análise dos componentes principais, que é muito exigente computacionalmente.

Na seção 6.6, é apresentada uma discussão sobre a modelagem do sinal transmitido pelo sistema CDMA. Mostra-se que, no artigo [21] de Xu e Tsatsanis, o sinal do usuário desejado é transmitido com uma potência maior que dos demais usuários, o que influencia na computação da SER. É proposta uma nova modelagem para o sistema.

---

**Algoritmo 6.1** Algoritmo de Equalização Cega por Variância Mínima de Saída

---

```

Para j=1:NúmeroDeUsuários,
    cj = GeraEspalhamento('CódigoDeEspalhamento',j);
    Cmatj = GeraMatrizEspalhamento(cj);
    CHCj = (CmatjHCmatj)-1
    Cnj = AchaComplementoOrtogonal(Cmatj);
    bj = GeraSinal(NúmeroDeSímbolos);
    sj = Convolução(cj,hj);
    hj = GeraCanal(ComprimentoDoCanal);
    r = r + Convolução(bj,sj);
    Potencia_Sinal_dB(j) = MedePotenciaSinalEmDB(Convolução(bj,sj));
    Potencia_Ruido_dB(j) = SNR_Escolhida_dB - Potencia_Sinal_dB;
    r = r + GeraRuído(Potencia_Ruido_dB(j));
fim Para

Para j=1:NúmeroDeUsuários,
    Para k= 1:NúmeroDeSímbolos,
        r̃(k) = CmatjHr;
        r̄(k) = CnjHr;
        aux = r̃H(j) Cmatjgj(k) - r̄H(k)wj,l(k)
        wj,l(k+1) = wj,l(k) + μwaux r̄(k)
        gj(k+1) = gj(k) + μgaux CHCjr̃(k)
        gj(k+1) = gj(k+1)/Norma2(gj(k+1));
    fim Para
    fj = CmatjCHCjgj(k+1) - Cnjwj,l(k)
fim Para

Para j=1:NúmeroDeUsuários,
    Para k=1:NúmeroDeSímbolos,
        símbolo_estimado(j,k) = fjHr;
    fim Para
fim Para

```

---

# Capítulo 7

## Equalização Cega Usando Predição Linear

### 7.1 Introdução

A técnica de predição linear para se realizar equalização cega vem sendo utilizada para sistemas monousuários já há um certo tempo. O artigo [25], de Gesbert e Duhamel, é um bom exemplo da aplicação dessa técnica no caso de sistemas SIMO. Nesses métodos, tenta-se prever a amostra atual com base nas saídas anteriores de todos os canais. O erro dessa predição é proporcional à inovação contida no símbolo transmitido. Na suposição de entrada branca e de ruído do tipo AWGN, o erro de predição é proporcional ao próprio símbolo transmitido.

O uso de sistemas adaptativos baseados em Kalman são mais adequados a ambientes não-estacionários que seus equivalentes do tipo LMS, devido à estrutura interna dos filtros de Kalman, baseados em equações de atualização de estados, que rastream mais facilmente a variações das estatísticas do ambiente. Aproveitando-se dessa característica foram desenvolvidas implementações que se utilizam de atualizações de Kalman para detecção multiusuário, como em [26].

Neste capítulo é apresentado um método de equalização cega que utiliza

predição linear para sistemas multiusuário com atualização do detetor multiusuário. Esse algoritmo foi originalmente descrito no artigo [27] de Fan e Li, com atualizações do tipo LMS, e no artigo [28] de Liu, Teh e Gunawan, com atualizações de Kalman. O algoritmo de Liu, Teh e Gunawan pode ser visto como uma generalização do detetor multiusuário baseado em Kalman apresentado em [26] para o caso em que há ISI.

A fim de que se entenda melhor como é feita equalização cega através de predição linear, na seção 7.2 é mostrado o uso de predição linear para equalização no caso monousuário. A seção 7.3 explora o caso multiusuário, apresentando algoritmos adaptativos baseados em Kalman para se fazer a MUD, no caso sem ISI. A seção 7.4 aborda a utilização de predição linear no caso multiusuário, com as atualizações do filtro detetor sendo feitas num método baseado em Kalman, conjuntamente com eliminação de ISI. Foi implementada também a versão LMS de Fan e Li de atualização de detector, como é apresentado na seção 7.5.

## 7.2 Predição Linear no Caso Monousuário

Considere um sistema SIMO em que um único usuário transmite a seqüência de símbolos  $\{b_1(k)\}$  pelos canais  $\mathbf{h}_j$ ,  $1 \leq j \leq J$ . As amostras  $\{r_j(k)\}$  correspondem à saída de cada um dos  $J$  canais. A figura 7.1 representa esse sistema.

Define-se um filtro preditor  $\mathbf{p}$  de ordem  $N$  e formam-se os seguintes vetores

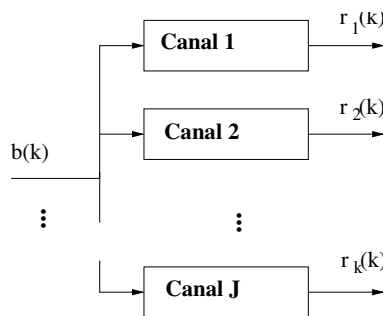


Figura 7.1: Modelo de sistema SIMO.

[25]:

$$\mathbf{x}(k) = \begin{bmatrix} b_1(k) \\ b_2(k) \\ \vdots \\ b_J(k) \end{bmatrix} \quad \bar{\mathbf{x}}(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{x}(k-1) \\ \vdots \\ \mathbf{x}(k-N+1) \end{bmatrix} \quad (7.1)$$

$$\mathbf{y}(k) = \begin{bmatrix} r_1(k) \\ r_2(k) \\ \vdots \\ r_J(k) \end{bmatrix} \quad \bar{\mathbf{y}}(k) = \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k-1) \\ \vdots \\ \mathbf{y}(k-N+1) \end{bmatrix} \quad (7.2)$$

O símbolo transmitido  $b_1(k)$  é descoberto através da seguinte equação [25]:

$$\tilde{b}_1(k) = \frac{r_i(k) - \mathbf{p}^H \bar{\mathbf{y}}(k)}{h_i(1)} \quad (7.3)$$

ou seja, tenta-se prever a saída do  $i$ -ésimo canal com base nas saídas anteriores dos canais. O erro nessa predição é uma estimativa do símbolo transmitido a menos de um fator de escala que corresponde ao primeiro coeficiente do  $i$ -ésimo canal. É mostrado ainda em [25] que, para preditores com norma maior que um dado limiar  $\gamma_0$ , a estimativa feita em (7.3) é não-tendenciosa, mesmo na presença de ruído branco.

## 7.3 MUD Cega Baseada em Kalman

No caso multiusuário, considera-se uma estrutura GSC como a apresentada na seção 6.4, apenas desconsiderando a presença de ISI. Com isso, a matriz de restrições  $\mathbf{C}$  se reduz a um vetor de pesos  $\mathbf{q}$  e o vetor  $\mathbf{g}$  se reduz a um escalar  $g$ . O novo sistema pode ser visto na figura 7.2. Nota-se que, a equação (6.27) pode ser reescrita como:

$$\mathbf{f}_j = g(\mathbf{q}^H \mathbf{q})^{-1} \mathbf{q} - \mathbf{Q}_n \mathbf{w}_{j,l} \quad (7.4)$$

$$= \frac{g}{\mathbf{q}^H \mathbf{q}} \mathbf{q} - \mathbf{Q}_n \mathbf{w}_{j,l} \quad (7.5)$$

Ao se escolher  $g = 1$  e  $\mathbf{q} = \mathbf{c}_j$ , ou seja, sendo o vetor de restrições igual à seqüência de espalhamento normalizada do  $j$ -ésimo usuário, tem-se:

$$\mathbf{f}_j = \mathbf{c}_j - \mathbf{Q}_n \mathbf{w}_{j,l} \quad (7.6)$$

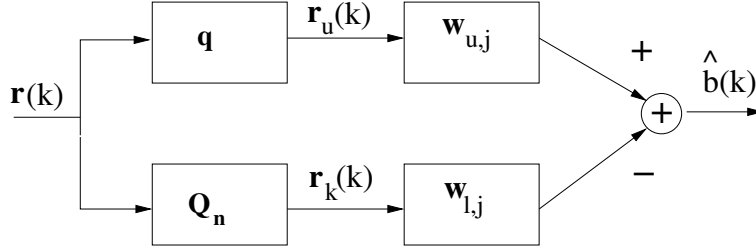


Figura 7.2: Modelo GSC para o sistema sem ISI.

Na seção 6.2, viu-se que a equação (6.1) expressa uma representação paramétrica para o detetor multiusuário  $\mathbf{f}_j$  para um sistema sem ISI [29] [26]. Suponha que a constelação transmitida seja simétrica em relação à origem, que os símbolos da constelação tenham a mesma probabilidade de serem transmitidos, e que a estimação do símbolo transmitido feita através do filtro  $\mathbf{f}_j$  seja não-tendenciosa. Nessas condições,

$$E[\mathbf{f}_j^H(k)\mathbf{r}(k)] = 0 \quad (7.7)$$

Considerando os dados transmitidos como aproximadamente brancos, pode-se interpretar a estimação dos dados de entrada como um AWGN de média zero. Define-se então

$$e(k) \triangleq \mathbf{f}_j^H(k)\mathbf{r}(k) \quad (7.8)$$

Substituindo o valor de  $\mathbf{f}_j$  da equação (7.6), obtém-se [26]

$$e(k) = \mathbf{c}_j^H \mathbf{r}(k) - \mathbf{r}^H(k) \mathbf{Q}_n \mathbf{w}_{j,l}(k) \quad (7.9)$$

Chamando  $\tilde{y}(k) = \mathbf{c}_j^H \mathbf{r}(k)$  e  $\mathbf{d}^H(k) = \mathbf{r}^H(k) \mathbf{Q}_n$ , chega-se à seguinte expressão:

$$\tilde{y}(k) = \mathbf{d}^H(k) \mathbf{w}_{j,l}(k) + e(k) \quad (7.10)$$

Pode-se interpretar essa equação como a equação de medidas numa modelagem de espaço-estado [26]. A variável  $e(k)$  pode ser vista como o ruído de medidas.



Observa-se que, para um sistema invariante no tempo, tem-se, em regime permanente,

$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) \quad (7.11)$$

o que, nesse caso, poderia ser visto como a equação de estado, na qual a matriz de transição de estado é uma identidade e o ruído de processo seja nulo [26].

Considerando a modelagem espaço-estado dada pelas equações (7.10) e (7.11), chega-se a atualizações do detetor multiusuário por um sistema baseado em Kalman [26]:

$$\mathbf{v}(k) = \frac{\mathbf{K}(k, k-1)\mathbf{d}(k)}{\mathbf{d}^H(k)\mathbf{K}(k, k-1)\mathbf{d}(k) + \epsilon} \quad (7.12)$$

$$\mathbf{K}(k+1, k) = \mathbf{K}(k, k-1) - \mathbf{v}(k)\mathbf{d}^H(k)\mathbf{K}(k, k-1) \quad (7.13)$$

$$\mathbf{w}_{j,l}(k) = \mathbf{w}_{j,l}(k-1) + \mathbf{v}(k)[\tilde{y}(k) - \mathbf{d}^H(k)\mathbf{w}_{j,l}(k-1)] \quad (7.14)$$

$$\mathbf{f}_j(k) = \mathbf{c}_j - \mathbf{Q}_n \mathbf{w}_{j,l}(k) \quad (7.15)$$

onde  $\mathbf{K}$  é a matriz de correlação do erro de Kalman e  $\mathbf{v}$  é o vetor de erro de Kalman.

## 7.4 MUD Cega com ISI Baseada em Kalman Usando Predição

Nesta seção tem-se uma abordagem do sistema levando-se em conta ISI. Define-se uma matriz  $\mathbf{C}_j$  contendo versões atrasadas da matriz de espalhamento.  $\mathbf{C}_j$  tem dimensões  $2P \times Q$ , sendo  $P$  o tamanho da seqüência de espalhamento e  $Q$  o comprimento do canal:

$$\mathbf{C}_j = \begin{bmatrix} \mathbf{c}_j & 0 & \dots & 0 \\ 0 & \mathbf{c}_j & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{c}_j \end{bmatrix} \quad (7.16)$$

Seja  $\mathbf{h}_j$  o canal referente ao usuário  $j$  e  $\mathbf{s}_j$  a assinatura do  $j$ -ésimo usuário, ou seja, a convolução de  $\mathbf{c}_j$  e  $\mathbf{h}_j$ . Como o tamanho da convolução excede o da seqüência

de espalhamento, seu valor está contido em  $\mathbf{s}_j(0)$  e espalha-se também para  $\mathbf{s}_j(1)$ . Então,

$$\mathbf{C}_j \mathbf{h}_j = \hat{\mathbf{s}}_j = \begin{bmatrix} \mathbf{s}_j(0) \\ \mathbf{s}_j(1) \end{bmatrix} \quad (7.17)$$

Considera-se

$$\mathbf{y}_1(k) = \begin{bmatrix} \mathbf{r}(k+1) \\ \mathbf{r}(k) \end{bmatrix} \quad (7.18)$$

O novo detetor  $\mathbf{f}_j$  será aplicado sobre  $\mathbf{y}_1(k)$  e, portanto, tem o dobro da dimensão de  $\mathbf{r}(k)$ . Baseado em (7.6), chega-se a uma representação canônica para  $\mathbf{f}_j$  [28]:

$$\mathbf{f}_j = \hat{\mathbf{s}}_j - \mathbf{S}_n \mathbf{w}_j(k) \quad (7.19)$$

onde  $\mathbf{S}_n$  tem em suas colunas uma base para o espaço nulo de  $\hat{\mathbf{s}}_j$ . Considera-se agora outro vetor auxiliar  $\mathbf{y}_2(k)$ :

$$\mathbf{y}_2(k) = \begin{bmatrix} \mathbf{r}(k+2) \\ \mathbf{S}_n \mathbf{y}_1(k) \\ \mathbf{r}(k-1) \end{bmatrix} \quad (7.20)$$

Nota-se que, ao pré-multiplicar  $\mathbf{y}_1(k)$  por  $\mathbf{S}_n$  retirou-se do vetor  $\mathbf{y}_2(k)$  toda a informação referente aos dados transmitidos pelo  $j$ -ésimo usuário. Tenta-se então estimar o vetor  $\mathbf{y}_1(k)$  através de um vetor  $\mathbf{y}_2(k)$  e de um preditor  $\mathbf{P}$ , com dimensões  $(P+Q-1) \times 2(P+Q-1)$ . O erro de predição  $\boldsymbol{\epsilon}_1$  é armazenado em  $\boldsymbol{\epsilon}$ :

$$\boldsymbol{\epsilon}_1(k) = \mathbf{y}_1(k) - \mathbf{P} \mathbf{y}_2(k) \quad (7.21)$$

Definindo-se

$$\mathbf{R}_1 = E[\mathbf{y}_1(k) \mathbf{y}_1^H(k)] \quad (7.22)$$

$$\mathbf{R}_2 = E[\mathbf{y}_2(k) \mathbf{y}_2^H(k)] \quad (7.23)$$

o preditor é então definido como [28]

$$\mathbf{P} = \mathbf{R}_1 \mathbf{R}_2^\dagger \quad (7.24)$$

onde o símbolo  $\dagger$  representa a pseudo-inversa de uma matriz [28].

Uma estimativa da assinatura do usuário em questão pode ser feita a partir de  $\epsilon_1$  [28]

$$\mathbf{s}_j \mathbf{s}_j^H = E[\epsilon_1(k) \epsilon_1^H(k)] \quad (7.25)$$

Sabe-se que  $\mathbf{s}_j \mathbf{s}_j^H$  é uma matriz de posto 1. Então, uma estimativa  $\mathbf{u}_j$  para  $\mathbf{s}_j$  pode ser feita a partir do vetor singular de  $E[\epsilon_1(k) \epsilon_1^H(k)]$  associado ao seu maior valor singular. Chega-se a uma expressão canônica para  $\mathbf{f}_j$  baseada na estimativa da assinatura do usuário desejado:

$$\mathbf{f}_j = \mathbf{u}_j(k) - \mathbf{U}_n \mathbf{w}_j(k) \quad (7.26)$$

onde  $\mathbf{U}_n$  tem em suas colunas uma base para o espaço nulo de  $\mathbf{u}_j$ . Assim como no sistema sem ISI, mostrado na seção 7.3, definem-se as seguintes variáveis auxiliares [28]:

$$\tilde{y}(k) = \mathbf{u}_j^H(k) \mathbf{y}_1(k) \quad (7.27)$$

$$\mathbf{d}^H(k) = \mathbf{y}_1^H(k) \mathbf{U}_n \quad (7.28)$$

Chega-se mais uma vez a uma expressão para a equação de medidas do espaço-estado:

$$\tilde{y}(k) = \mathbf{d}^H(k) \mathbf{w}_j(k) + \mathbf{f}^H(k) \mathbf{y}_1(k) \quad (7.29)$$

A equação de atualização de estados mantém-se a mesma do modelo de sistema sem ISI:

$$\mathbf{w}_j(k+1) = \mathbf{w}_j(k) \quad (7.30)$$

Então, a partir de (7.29) e (7.30), chega-se nas equações de atualização de Kalman para o detetor  $\mathbf{f}_j$  [28]:

$$\mathbf{v}(k) = \frac{\mathbf{K}(k, k-1) \mathbf{d}(k)}{\mathbf{d}^H(k) \mathbf{K}(k, k-1) \mathbf{d}(k) + \epsilon} \quad (7.31)$$

$$\mathbf{K}(k+1, k) = \mathbf{K}(k, k-1) - \mathbf{v}(k) \mathbf{d}^H(k) \mathbf{K}(k, k-1) \quad (7.32)$$

$$\mathbf{w}_j(k) = \mathbf{w}_j(k-1) + \mathbf{v}(k) [\tilde{y}(k) - \mathbf{d}^H(k) \mathbf{w}_j(k-1)] \quad (7.33)$$

$$\mathbf{f}_j(k) = \mathbf{u}_j(k) - \mathbf{U}_n \mathbf{w}_j(k) \quad (7.34)$$

onde o parâmetro  $\epsilon$ , na prática, pode ser fixado em 1 [28].

O cálculo da pseudo-inversa de  $\mathbf{R}_2$  é bastante custoso computacionalmente. No artigo [28] é oferecida a opção de se computar a pseudo-inversa da maneira tradicional por algum tempo (algo como as 100 primeiras iterações). A seguir, a pseudo-inversa pode ser computada iterativamente, através de uma atualização recursiva:

$$\mathbf{e}(k) = \frac{\alpha \mathbf{R}_2^\dagger(k-1) \mathbf{y}_2(k)}{1 + \alpha \mathbf{y}_2^H(k) \mathbf{R}_2(k-1) \mathbf{y}_2(k)} \quad (7.35)$$

$$\mathbf{R}_2^\dagger(k) = \alpha \mathbf{R}_2^\dagger(k-1) - \alpha \mathbf{e}(k) \mathbf{y}_2^H(k) \mathbf{R}_2^\dagger(k-1) \quad (7.36)$$

onde  $\alpha$  é um número menor que 1, normalmente escolhido bem próximo de 1 [28].

Um esquema de funcionamento do algoritmo pode ser visto a partir da figura 7.3. A medida que os dados chegam no receptor é realizada uma transformação linear retirando as informações do usuário de interesse do vetor auxiliar  $\mathbf{y}_2(k)$ . A seguir, faz-se uma predição linear dos dados recebidos com base em  $\mathbf{y}_2(k)$ . A partir do erro de predição encontra-se uma estimativa para a assinatura do usuário em questão. Essa estimativa é usada nas atualizações baseadas no filtro de Kalman dos coeficientes do filtro detetor  $\mathbf{f}_j$ . Uma descrição mais detalhada da implementação deste algoritmo pode ser encontrada no quadro Algoritmo 7.1.

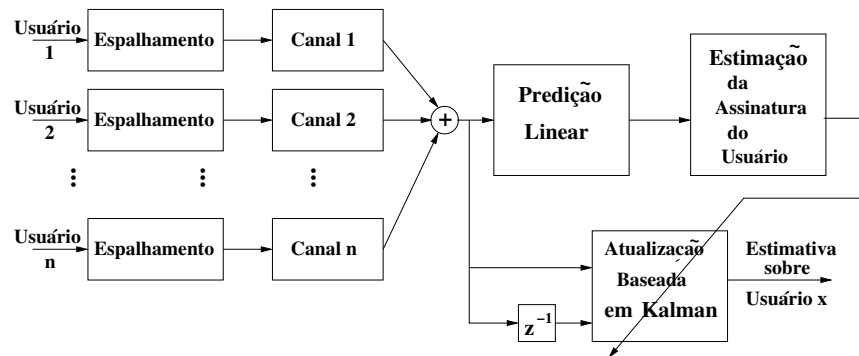


Figura 7.3: Esquema de funcionamento do algoritmo de equalização cega por predição linear baseado em Kalman.

## 7.5 Versão LMS da Atualização do Detector

A fim de se estudar melhor qual o efeito da filtragem de Kalman sobre as atualizações do algoritmo de predição linear, foi implementada uma versão do algoritmo em que as atualizações do detector seriam feitas através do algoritmo adaptativo LMS. Outra razão para a implementação LMS é a tentativa de se comparar os algoritmos num ambiente mais uniforme possível, no caso, unificando a todos para o uso de algoritmos do tipo gradiente decrescente.

A implementação utilizada foi baseada nos artigos [27] e [30], de Fan e Li, que se baseiam na variante LMS-Frost para resolver o seguinte problema de otimização:

$$\min \|\mathbf{f}_j^H \mathbf{y}_1\|^2 \quad \text{sujeito a} \quad \mathbf{f}_j^H \mathbf{u} = 1 \quad (7.37)$$

A atualização do detector é feita a partir de [27]:

$$\mathbf{f}_j(k+1) = \mathbf{u}(\mathbf{u}^H \mathbf{u})^{-1} + [\mathbf{I} - \mathbf{u}(\mathbf{u}^H \mathbf{u})^{-1} \mathbf{u}^H][\mathbf{f}_j(k) - \mu \mathbf{y}_1 \mathbf{y}_1^H \mathbf{f}_j(k)] \quad (7.38)$$

onde  $\mathbf{I}$  representa a matriz identidade de dimensões adequadas e  $\mu$  é o passo de atualização do sistema. Essa estrutura adaptativa é utilizada em lugar das atualizações de estado de Kalman baseadas na estrutura GSC, como descrita na seção 7.4. A versão LMS do algoritmo é mostrada no quadro 7.2.

## 7.6 Conclusões

Neste capítulo é apresentada a terceira e última família de métodos para equalização cega usando estatísticas de segunda ordem para CDMA que são apresentados nesta tese. Desta vez, o método apresentado é o que utiliza predição cega, tanto com atualizações dos coeficientes através de LMS quanto com atualizações baseadas em Kalman.

A implementação está baseada no artigo [28] de Liu, Teh e Gunawan que procura reunir três técnicas diferentes para MUD cega: o uso de filtragem de

Kalman, a estimação da assinatura via predição linear e a eliminação da ISI durante a detecção multiusuário. A união de duas dessas técnicas já havia sido realizada, pelo menos em [26] e [27], mas a aplicação conjunta das três é o passo mais importante do algoritmo de Liu-Teh-Gunawan. Como a eliminação da ISI durante a MUD já havia sido abordada nos capítulos 5 e 6, optou-se neste capítulo por um maior aprofundamento do uso de predição linear, o que é feito na seção 7.2 e da utilização da filtragem de Kalman, explicada na seção 7.3. A seção 7.4 apresenta, com base nas seções anteriores, o algoritmo de Liu-Teh-Gunawan. Por fim, uma estrutura adaptativa LMS é mostrada na seção 7.5, com fins de se aprofundar na comparação entre os algoritmos.

A contribuição deste capítulo é o de mostrar o algoritmo de Liu, Teh e Gunawan dentro de um contexto maior, em que podem ser vistos tanto como uma evolução de algoritmo de equalização cega com predição linear usando atualizações LMS, como também sendo uma evolução do algoritmo MUD com filtragem de Kalman, ou ainda, como um passo adiante na incorporação tanto de filtragem de Kalman como de predição linear em estruturas GSCs que eliminam ISI como a mostrada no capítulo 6. Além disso, é mostrada a versão LMS-Frost do algoritmo, uniformizando-se o ambiente de trabalho em relação aos demais algoritmos. [27].

## 7.6. CONCLUSÕES

---

**Algoritmo 7.1** Algoritmo de Equalização Cega por Predição Linear, usando Kalman

---

```
Para j=1:NúmeroDeUsuários,  
    cj = GeraEspalhamento('CódigoDeEspalhamento',j);  
    Cmatj = GeraMatrizEspalhamento(cj);  
    bj = GeraSinal(NúmeroDeSímbolos);  
    sj = Convolução(cj,hj);  
     $\hat{\mathbf{S}}_j$  = AchaComplementoOrtogonal(sj);  
    hj = GeraCanal(ComprimentoDoCanal);  
    r = r + Convolução(bj,sj);  
    Potencia_Sinal_dB(j) = MedePotenciaSinalEmDB(Convolução(bj,sj));  
    Potencia_Ruido_dB(j) = SNR_Escolhida_dB - Potencia_Sinal_dB;  
    r = r + GeraRuído(Potencia_Ruido_dB(j));  
fim Para
```

```
Para j=1:NúmeroDeUsuários,  
    Para k = 1:NúmeroDeSímbolos,  
        y1(k) = [r(k+1); r(k)];  
        y2(k) = [r(k+2);  $\hat{\mathbf{S}}_j \mathbf{y}_1$ ; r(k-1)];  
         $\mathbf{R}_1(k)$  =  $\frac{1}{k}$  [(k-1) $\mathbf{R}_1(k-1)$ ] + y1(k)y2H(k);  
  
        Se k ≤ 100,  
            Então  
                 $\mathbf{R}_2(k)$  =  $\frac{1}{k}$  [(k-1) $\mathbf{R}_2(k-1)$ ] + y2(k)y2H(k);  
  
                 $\mathbf{Z}(k)$  = AchaPseudoInversa( $\mathbf{R}_2(k)$ );  
            Fim Então  
        Senão  
            e(k) =  $\frac{\alpha \mathbf{Z}(k-1) \mathbf{y}_2(k)}{1 + \alpha \mathbf{y}_2^H(k) \mathbf{Z}(k-1)}$ ;  
  
             $\mathbf{Z}(k)$  =  $\alpha \mathbf{Z}(k-1) - \alpha \omega \mathbf{y}_2^H(k) \mathbf{Z}(k-1)$ ;
```

---

Fim Senão

$$\mathbf{P} = \mathbf{R}_1(k) \mathbf{Z}(k);$$

$$\epsilon_1 = \mathbf{y}_1(k) - \mathbf{P}\mathbf{y}_2(k);$$

$$\mathbf{R}_\epsilon = \frac{1}{k} [(k-1)\mathbf{R}_\epsilon(k-1) + \epsilon_1(k)\epsilon_1^H(k);$$

$$\mathbf{u}_j = \text{AchaVetorSingularDoMaiorValorSingular}(\mathbf{R}_\epsilon);$$

$$\mathbf{U}_{j,n} = \text{AchaComplementoOrtogonal}(\mathbf{u}_j);$$

$$\mathbf{d}(k) = (\mathbf{y}_1^H(k)\mathbf{U}_{j,n})^H ;$$

$$\tilde{y}(k) = \mathbf{u}_j^H(k)\mathbf{y}_1(k);$$

$$\mathbf{g}(k) = \frac{\mathbf{k}(k, k-1)\mathbf{d}(k)}{\mathbf{d}^H(k)\mathbf{k}(k, k-1)\mathbf{d}(k) + 1};$$

$$\mathbf{k}(k+1, k) = \mathbf{k}(k, k-1) - \mathbf{g}(k)\mathbf{d}^H(k)\mathbf{k}(k, k-1);$$

$$\mathbf{w}_j(k) = \mathbf{w}_j(k-1) + \mathbf{g}(k)[\tilde{y}(k) - \mathbf{d}^H(k)\mathbf{w}_j(k-1)];$$

$$\mathbf{f}_j(k) = \mathbf{u}_j(k) - \mathbf{U}_n\mathbf{w}_j(k)$$

fim Para

fim Para

Para j=1:NúmeroDeUsuários,

Para k=1:NúmeroDeSímbolos,

$$\text{símbolo\_estimado}(j, k) = \mathbf{f}_j^H \mathbf{y}_1(k);$$

fim Para

fim Para

---



## 7.6. CONCLUSÕES

---

**Algoritmo 7.2** Algoritmo de Equalização Cega por Predição Linear, com Atualizações LMS

---

```
Para j=1:NúmeroDeUsuários,
    cj = GeraEspalhamento('CódigoDeEspalhamento',j);
    Cmatj = GeraMatrizEspalhamento(cj);
    bj = GeraSinal(NúmeroDeSímbolos);
    sj = Convolução(cj,hj);
     $\hat{\mathbf{S}}_j$  = AchaComplementoOrtogonal(sj);
    hj = GeraCanal(ComprimentoDoCanal);
    r = r + Convolução(bj,sj);
    Potencia_Sinal_dB(j) = MedePotenciaSinalEmDB(Convolução(bj,sj));
    Potencia_Ruido_dB(j) = SNR_Escolhida_dB - Potencia_Sinal_dB;
    r = r + GeraRuído(Potencia_Ruido_dB(j));
fim Para
```

```
Para j=1:NúmeroDeUsuários,
    Para k = 1:NúmeroDeSímbolos,
        y1(k) = [r(k+1); r(k)];
        y2(k) = [r(k+2);  $\hat{\mathbf{S}}_j \mathbf{y}_1$ ; r(k-1)];
         $\mathbf{R}_1(k)$  =  $\frac{1}{k}$  [(k-1) $\mathbf{R}_1(k-1)$ ] + y1(k)y1H(k);

         $\mathbf{R}_2(k)$  =  $\frac{1}{k}$  [(k-1) $\mathbf{R}_2(k-1)$ ] + y2(k)y2H(k);

         $\mathbf{Z}(k)$  = AchaPseudoInversa( $\mathbf{R}_2(k)$ );
         $\mathbf{P}$  =  $\mathbf{R}_1(k)\mathbf{Z}(k)$  ;
         $\epsilon_1$  = y1(k) -  $\mathbf{P}\mathbf{y}_2(k)$ ;
         $\mathbf{R}_\epsilon$  =  $\frac{1}{k}$  [(k-1) $\mathbf{R}_\epsilon(k-1)$ ] +  $\epsilon_1(k)\epsilon_1^H(k)$ ;

        uj = AchaVetorSingularDoMaiorValorSingular( $\mathbf{R}_\epsilon$ );
        uaux =  $\frac{\mathbf{u}}{\mathbf{u}^H \mathbf{u}}$  ;
```

## 7.6. CONCLUSÕES

---

$$\mathbf{f}_j = \mathbf{u}_{aux} + \mathbf{I} - (\mathbf{u}_{aux}\mathbf{u}) (\mathbf{f}_j - \mu \mathbf{y}_1(k) \mathbf{y}_1^H(k) \mathbf{f}_j);$$

fim Para

fim Para

Para j=1:NúmeroDeUsuários,

Para k=1:NúmeroDeSímbolos,

$$\text{símbolo\_estimado}(j,k) = \mathbf{f}_j^H \mathbf{y}_1(k);$$

fim Para

fim Para

---

# Capítulo 8

## Análise Comparativa

### 8.1 Introdução

Nos capítulos 5, 6 e 7 foram descritos três métodos de equalização cega para CDMA. Cada um desses algoritmos utiliza uma abordagem diferente para fazer as três funções básicas de um equalizador cego em ambiente multiusuário: eliminação da ISI, supressão do ruído aditivo perturbações aditivas do sinal e separação dos usuários.

Neste capítulo pretende-se comparar os algoritmos existentes, colocando-se como foco principal a medida da taxa de erro de símbolo (SER - *symbol error rate*) dos algoritmos em relação à razão sinal-ruído (SNR - *signal-to-noise ratio*). Enquanto nos capítulos anteriores procurou-se dar ênfase a cada algoritmo separadamente, neste capítulo pretende-se fazer comparações e se destacarem as diferenças entre eles.

A seção 8.2 procura dar o panorama geral dos métodos de equalização cega implementados, através de uma visão comparativa entre os algoritmos. As três funções básicas de um equalizador cego são o foco da seção 8.3, mais uma vez com ênfase nas diferenças apresentadas pelos algoritmos já apresentados. Por fim, a seção 8.4 tem a função de apresentar e discutir os resultados produzidos pelas

simulações dos algoritmos de subespaço, CMV e de predição linear.

## 8.2 Visão Geral dos Algoritmos

De uma maneira geral, sistemas CDMA apresentam inerentemente uma certa resistência à presença de ISI, principalmente pelo uso de seqüências de espalhamento. A principal motivação para o desenvolvimento de métodos de equalização cega surgiu com o desenvolvimento de sistemas de detecção multiusuário (MUD - *multiuser detection*), detalhados na seção 4.6. Detetores multiusuário representam melhorias nos sistemas CDMA em relação à SER, por diminuírem o nível da interferência entre os usuários (MUI - *multiuser interference*).

O desenvolvimento de detetores multiusuários cegos representou a dispensa do uso de seqüência de treinamento ou do conhecimento da seqüência de espalhamento dos demais usuários para a detecção do usuário de interesse [22]. O uso da métrica MMSE representa resistência à presença de ruído branco. A MUD, por si só, já melhora a separação entre os usuários. Dos três tipos principais de interferência do CDMA, o único tipo de interferência que faltava ser combatida era a ISI. Foram então desenvolvidos detetores multiusuários cegos com estimação cega de canal. Com isso, a função desses detetores cegos passou a ser a mesma de equalizadores cegos em ambiente multiusuário. Em geral, pode-se dizer que os algoritmos de equalização cega para multiusuário surgiram a partir de modificações nas técnicas já existentes para o caso monousuário, combinando-as com o uso de MUD.

O uso de método de subespaço para equalização cega em CDMA pode ser visto desde os artigos de Bensley e Aazhang [31] e de Liu e Xu [32]. Suas origens estão ligadas à técnica de análise de componentes principais (PCA - *principal component analysis*), muito usada em diversos problemas de processamento de sinais em geral, e em algumas aplicações em comunicações, como processamento de sinais em *arrays* (receptores com diversidade espacial). Esses métodos têm como ponto mais criticado a alta complexidade computacional exigida pela PCA, por exigir

estimação da matriz de correlação do sinal recebido e por se fazer a decomposição em autovalores e autovetores dessa matriz. O artigo de Wang e Poor [17] é importante por separar claramente as etapas de eliminação do ruído, estimação de canal e separação de usuários. É proposta uma implementação LMS, algoritmo adaptativo com baixa complexidade computacional, e a substituição da PCA por algoritmos que rastreiam o subespaço do sinal e do ruído, como o PASTd [20].

Os métodos de equalização cega para CDMA baseados em CMV, também chamados de métodos de MOE (*minimum output energy*), tiveram sua origem no artigo [33] de Tsatsanis, onde o detetor multiusuário cego adaptativo baseado em CMV proposto por Honig e Verdú em [22] é generalizado para o caso de sistemas com ISI. A motivação principal de se empregar CMV como métrica de equalização cega está ligado à teoria de estimação, e, em particular, ao estimador de mínima variância. Além disso, como mostrado na seção 6.2, essa métrica tem uma relação estreita com a métrica MMSE, sendo, no entanto, bem mais adequada a ser usada em equalização cega, por não depender do conhecimento do ganho originado pelos canais nos dados transmitidos. Além disso, os métodos baseados em CMV não requerem cálculos de pseudo-inversas de grandes matrizes e nem decomposições em autovalores e autovetores, principais problemas do método de subespaço. No entanto, por ter uma função custo baseada em minimização do efeito do ruído e, simultaneamente, na maximização do sinal do usuário de interesse, além da obediência a um conjunto grande de restrições, o custo computacional do método é também considerável. A importância do artigo [21] de Xu e Tsatsanis, que foi o artigo escolhido para representar família de algoritmos é a proposição de métodos adaptativos para o CMV, facilitando assim seu uso em aplicações de tempo real. Em particular, a estrutura GSC, mostrada na seção 6.4, revelou-se bastante eficiente computacionalmente e adequada ao problema da equalização cega, por colocar em sub-bandas diferentes a questão da separação dos usuários e a eliminação do ruído.

Algoritmos de equalização cega para CDMA utilizando predição linear têm sua origem no artigo [30], de Fan e Li. A motivação para o desenvolvimento desses

algoritmos é uma série de propriedades estatísticas, com ênfase na equivalência entre os modelos média móvel e auto-regressivo para o sinal recebido [34]. Os métodos que utilizam predição linear não requerem geralmente um alto custo computacional, apesar de precisar fazer duas atualizações por passo de iteração, tanto no filtro detetor quanto no preditor. Uma das vantagens desses métodos é uma robustez quanto à estimação da ordem do canal utilizado [27]. O algoritmo de Liu, Teh e Gunawan, descrito em [28], tem requisitos computacionais mais elevados, principalmente devido à utilização de atualizações de Kalman e por ter que identificar o subespaço do sinal da mistura que chega no receptor. Sua principal contribuição é unir o uso de predição linear para estimação da assinatura do usuário em questão, a incorporação do problema de multipercursos ao detetor multiusuário e o uso de atualizações de Kalman para o filtro detetor.

## 8.3 Funcionalidades Básicas

As funções básicas de um equalizador em um ambiente multiusuário são a supressão do ruído (*denoising*), a eliminação da ISI e a separação dos usuários. Essas funções estão ligadas aos principais obstáculos à transmissão nesses tipos de sistema, ou seja, a presença de outros sinais elétricos completamente desconhecidos, a presença de multipercursos e a necessidade de multiplexação para transmissão de vários usuários.

Como se pode ver da tabela 8.1, no método de subespaço, a função de supressão do ruído é realizada numa etapa bem separada das demais, através da PCA. A eliminação de ISI e separação dos usuários são feitas no processo de minimização da função custo  $\|\mathbf{D}\mathbf{w}\|^2$ , sendo a eliminação de ISI mais ligada à estimação do subespaço de assinaturas. No caso do CMV, a separação da função de supressão do ruído das demais é ainda mais evidente, por colocar a ISI e a separação dos usuários em sub-bandas distintas. Por fim, o algoritmo de Liu, Teh e Gunawan usou também uma estrutura GSC, tendo a estimativa do efeito do canal calculada indiretamente por uma predição linear.

Tabela 8.1: Comparação das funcionalidades básicas dos três algoritmos analisados.

Funcionalidade	Subespaço	CMV	Predição Linear
Supressão de Ruído	PCA	Minimização da Variância	Minimização da Variância
Eliminação de ISI	Estimação do Subespaço das Assinaturas	Maximização da Sub-banda Principal	Maximização da Sub-banda Principal
Separação de Usuários	Minimização de $\ \mathbf{D}\mathbf{w}\ ^2$	Obediência a: $\mathbf{C}_j^H \mathbf{f} = \mathbf{g}_j$	Predição Linear

Na verdade, percebe-se que a função de supressão do ruído se diferencia bastante da eliminação da ISI e separação dos usuários, que podem, em conjunto, ser vistas como estimação cega da assinatura do usuário em questão. Uma explicação para essa diferenciação é a própria natureza das interferências que são combatidas em cada uma dessas etapas. Enquanto a ISI e a MUI são interferências correlacionadas consigo mesmas (coloridas) por terem passado pelo canal, o modelo de ruído introduzido é branco, descorrelacionado, e por isso, requer um tratamento estatístico bem menos rigoroso.

## 8.4 Simulações

A fim de se verificarem algumas propriedades do sistema, foram realizadas algumas simulações usando-se implementações baseados nos pseudo-códigos contidos nos Algoritmos 5.1, 6.1 e 7.1. Alguns parâmetros gerais usados nas simulações:

- Foram transmitidos 20.000 símbolos.
- A SER foi medida a partir da iteração 10.000 a fim de dar tempo para os algoritmos convergirem.
- Foi utilizada modulação BPSK.
- Foram feitas simulações para número de usuários de 1 a 5.

- Foram realizadas 5 execuções no *ensemble*.
- Seqüência de espalhamento: Códigos de Gold de tamanho 31.
- Canais de comprimento 3.

Além disso, cada algoritmo tem alguns parâmetros específicos, encontrados após um ajuste experimental.

### 1. Subespaço

- $\mu = 0,007$ .
- $\alpha = 1$ .

### 2. CMV

- $\mu_g = 0,0001$ . (modelagem da tese)
- $\mu_g = 0,001$ . (modelagem do artigo)
- $\mu_w = 0,01$

### 3. Predição Linear - Kalman

- Período com cálculo da pseudo-inversa: 300 iterações.
- $\beta = 0,99$ .

### 4. Predição Linear - LMS

- $\mu = 0,005$ .

Tradicionalmente, escolhe-se um usuário como alvo e as simulações são feitas medindo-se a SNR e a SER sobre esse usuário. No entanto, a fim de se colocar uma situação mais realista e, com isso, tendo-se uma compreensão mais ampla dos resultados, em todas as simulações tanto a SNR quanto a SER são medidos para cada um dos usuários. A SNR é determinada *a priori*, calculando-se a potência do ruído com base na potência do sinal que chega referente a um dos usuários, o qual apresenta flutuações. No caso de se haver mais de um usuário no sistema, a



potência do ruído é calculada com base no sinal com potência mais alta, a fim de se analisar o pior caso. No caso da SER, a taxa apresentada é a média de todos os usuários.

### 8.4.1 Simulações Usando Transmissão Síncrona e Canais com Memória

A escolha dos canais a serem utilizados revelou-se bastante difícil. Na verdade os algoritmos são sensíveis à presença de canais diferentes com zeros próximos entre si. Por isso, ao se aumentar o número de usuários, vai-se tornando progressivamente mais difícil que se ache zeros adequados. Nas simulações iniciais foram utilizados os canais representados nas equações (8.1) até (8.5):

$$\text{Canal do Usuário 1: } 0,5592 - 0,0395z^{-1} + 0,1380z^{-2} \quad (8.1)$$

$$\text{Canal do Usuário 2: } 0,2518 - 0,4698z^{-1} + 0,2219z^{-2} \quad (8.2)$$

$$\text{Canal do Usuário 3: } 0,3945 - 0,1748z^{-1} + 0,3835z^{-2} \quad (8.3)$$

$$\text{Canal do Usuário 4: } 0,3023 - 0,3996z^{-1} + 0,2869z^{-2} \quad (8.4)$$

$$\text{Canal do Usuário 5: } 0,4360 - 0,3605z^{-1} + 0,1151z^{-2} \quad (8.5)$$

Pode-se visualizar os zeros dos canais através da figura 8.1. Nota-se que há uma certa distância entre os zeros e que esses zeros estão concentrados no lado direito do plano. Isso foi feito pois se percebeu, empiricamente, que os algoritmos tendiam a apresentar erros não só quando os zeros eram próximos entre si, mas também quando tinham apenas a parte imaginária próxima.

Foram feitas simulações para os algoritmos de subespaço, CMV e de predição linear. Os resultados estão resumidos nas tabelas 8.2 a 8.6. A formatação básica da tabela é feita de maneira que na vertical encontram-se valores de SNR em dB, enquanto na horizontal está o número de usuários do sistema. Cada entrada da tabela representa a SER (*symbol error rate*).

Lembra-se que são mostradas duas implementações do algoritmo de subespaço, de acordo com a questão discutida na seção 5.5. A implementação baseada

Tabela 8.2: Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais com memória - Equação (5.43 - Modelagem do Artigo.)

<b>Algoritmo de Subespaço</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,336	0,349	0,354	0,353	0,335
<b>-15</b>	0,227	0,193	0,204	0,194	0,212
<b>-10</b>	0,065	0,041	0,062	0,079	0,114
<b>-5</b>	0,0006	0,0012	0,005	0,013	0,028
<b>0</b>	0	0	0	0,00028	0,0007
<b>5</b>	0	0	0	0	0

Tabela 8.3: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais com memória - Modelagem do Artigo.

<b>Algoritmo CMV</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,319	0,318	0,318	0,316	0,315
<b>-15</b>	0,179	0,174	0,175	0,176	0,175
<b>-10</b>	0,044	0,045	0,045	0,046	0,045
<b>-5</b>	0,0011	0,0011	0,0011	0,0011	0,0013
<b>0</b>	0	0	0	0	0

Tabela 8.4: Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gunawan - Canais com memória.

<b>Algoritmo de Predição Linear - Kalman</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,290	0,297	0,300	0,305	0,311
<b>-15</b>	0,165	0,171	0,190	0,197	0,215
<b>-10</b>	0,043	0,047	0,069	0,085	0,106
<b>-5</b>	0,0013	0,0017	0,0065	0,014	0,0286
<b>0</b>	0	0	0,0000014	0,00048	0,0015
<b>5</b>	0	0	0	0	0

Tabela 8.5: Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais com Memória - Equação (4.12) - Modelagem da Tese.

<b>Algoritmo de Subespaço</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,291	0,300	0,315	0,319	0,319
<b>-15</b>	0,160	0,171	0,190	0,207	0,218
<b>-10</b>	0,042	0,047	0,074	0,089	0,116
<b>-5</b>	0,0010	0,0014	0,0092	0,0069	0,041
<b>0</b>	0	0	0,00005	0,0055	0,0093
<b>5</b>	0	0	0	0,0013	0,0028
<b>10</b>	0	0	0	0,0004	0,0009
<b>15</b>	0	0	0	0,00007	0,0003
<b>10</b>	0	0	0	0	0,00007

Tabela 8.6: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais com memória - Modelagem da Tese.

<b>Algoritmo CMV</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,314	0,367	0,360	0,371	0,366
<b>-15</b>	0,174	0,237	0,253	0,263	0,262
<b>-10</b>	0,047	0,105	0,130	0,140	0,147
<b>-5</b>	0,0015	0,0018	0,039	0,043	0,055
<b>0</b>	0	0	0,00036	0,0056	0,0093
<b>5</b>	0	0	0,00002	0,00013	0,00022
<b>10</b>	0	0	0	0	0

Tabela 8.7: Resultados para o Algoritmo de Predição Linear de Fan e Li - Canais com memória.

<b>Algoritmo de Predição Linear - LMS</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,308	0,317	0,324	0,327	0,324
<b>-15</b>	0,173	0,184	0,201	0,210	0,222
<b>-10</b>	0,045	0,053	0,075	0,091	0,087
<b>-5</b>	0,0016	0,0022	0,0075	0,016	0,014
<b>0</b>	0	0,00008	0,0001	0,00084	0,0010
<b>5</b>	0	0	0	0	0

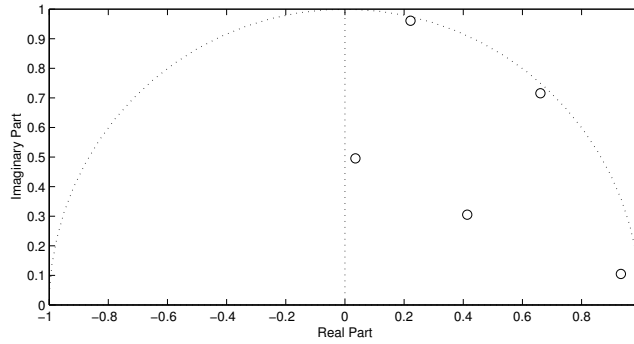


Figura 8.1: Zeros dos canais das equações (8.1) até (8.5). Como os coeficientes dos canais são reais, os zeros são pares conjugados. Com isso, só se representa o semi-plano  $Z$  superior, pois o semi-plano inferior é simétrico.

na equação (5.43) está de acordo com a modelagem feita pelo artigo de Wang e Poor [17]. A essa implementação, será referenciada como método de subespaço com modelagem do artigo. Já a implementação baseada na equação (4.12) tem a modelagem que reflete a ISI no receptor. Essa implementação é referenciada como método de subespaço com modelagem da tese.

Do mesmo modo que o método de subespaço, a seção 6.6 mostra que o artigo [21] contém uma modelagem do sinal CDMA recebido que o favorece em comparações com os demais algoritmos. Assim, é feita também uma comparação entre a modelagem do sistema a partir do artigo [21], baseada tanto na equação (6.36) quanto na equação (6.38) e a modelagem proposta nesta tese, baseada apenas na equação (6.36).

A partir dos resultados apresentados nas tabelas 8.2 a 8.7, pode-se notar que os sistemas são, em geral, robustos à presença de ruído aditivo e branco. Em geral, sistemas de comunicação que utilizam espalhamento espectral são caracterizados por apresentar resistência a ruídos de banda estreita, por aumentar a banda transmitida, escapando da presença de ruídos concentrados em uma curta faixa de frequência [12]. No entanto, sistemas CDMA com código de espalhamento longo permitem que uma grande seqüência de chips possa ser usada para a detecção do símbolo em questão. Numa transmissão em que se utiliza uma constelação simples

como BPSK, o uso de vários chips para a determinação de um símbolo representa uma grande resistência ao ruído, inclusive ao branco. Além disso, técnicas de supressão de ruído já estão bastante amadurecidas, o que influencia no bom funcionamento das partes relativas à supressão de ruído em sistemas de equalização cega. Com isso, pode-se atribuir boa parte da robustez quanto à presença de ruído ao código de Gold de tamanho 31, ao uso do BPSK e às técnicas de supressão de ruído nos equalizadores.

Nos gráficos 8.2, 8.3 e 8.4, pode-se ver uma comparação de desempenho entre os equalizadores, medida através da SER. Analisando-se o desempenho do sistema com apenas um usuário, pode-se eliminar *a priori* o fator multiusuário das interferências no sistema, analisando-se melhor os algoritmos quanto a resistência à ISI e ao ruído aditivo. Nota-se que, para SNRs mais baixas, na faixa entre -20dB e -15dB, o desempenho de todos os métodos é bastante parecido. No entanto, ao se diminuir o nível de ruído, chegando-se à faixa em torno de -5dB, nota-se que o desempenho do algoritmo subespaço com modelagem da tese revelou-se o pior de todos para 5 usuários. A comparação dos algoritmos nessa faixa de SNR revela que a função de eliminação de ISI, quando analisada isoladamente, é melhor realizada pelo algoritmo de predição linear, se forem consideradas somente as modelagens feitas nesta tese. O CMV com modelagem da tese teve um desempenho intermediário, sendo ligeiramente pior para 3 usuários.

Uma explicação para o que está acontecendo para o algoritmo de subespaço com modelagem da tese é uma saturação do algoritmo quanto ao número de usuários. A diferença do algoritmo de subespaço com modelagem da tese para o que foi escrito no artigo [17] é a introdução de ISI para dados do mesmo usuário. Essa ISI deve aumentar a correlação nas seqüências de espalhamento de diferentes usuários de uma maneira que não seja possível se fazer uma identificação do subespaço do sinal.

Com relação ao CMV, pode-se perceber que a modelagem do artigo têm resultados melhores que a feita nesta tese, com uma tendência a acentuar essa vantagem ao se aumentar o número de usuários. A partir do que foi colocado na

seção 6.6, essa tendência é explicada pela maior potência que o sinal desejado chega no receptor com relação aos demais. A modelagem contida no artigo [21] dá uma vantagem artificial ao algoritmo CMV na questão da separação multiusuário.

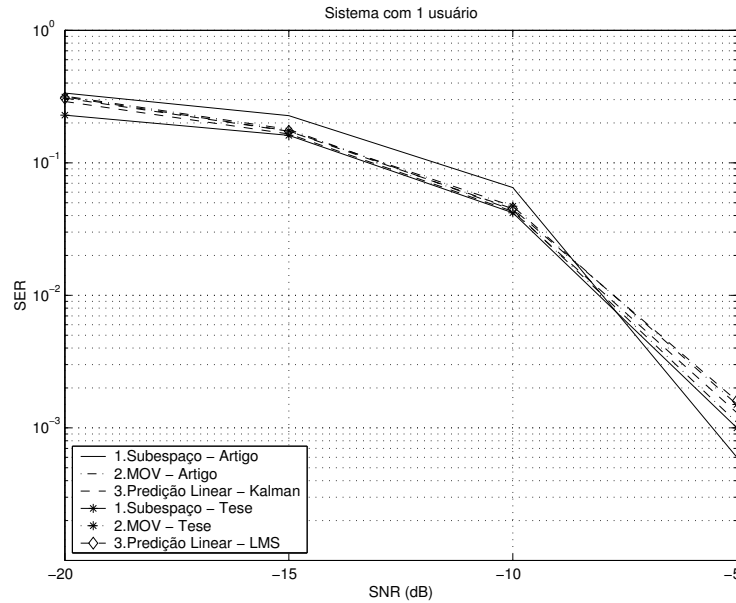


Figura 8.2: Comparação de desempenho dos algoritmos de equalização cega com 1 usuário no sistema.

Pode-se verificar, pelo conjunto total de dados, que há uma tendência ao desempenho do sistema sofrer mudanças menores na medida em que se acrescenta usuários num sistema que já apresenta um número grande de usuários para os algoritmos CMV e de predição linear, ou seja, o aumento da SER é mais significativo ao se passar de 1 para 2 usuários do que no caso de se aumentar de 4 para 5 usuários. Isso é de certa forma esperado, pois, para um dado usuário, o sinal de um outro usuário pode ser visto como ruído. No caso limite em que não se tem ISI e que se tem um número grande de usuários, esse ruído tende a ser um AWGN (*additive white gaussian noise*) que se soma ao AWGN presente no sistema. A partir de um certo valor, a quantidade de interferência acrescentada pela entrada de mais um usuário no sistema é bem menor que o próprio valor da interferência já existente, fazendo o sistema pouco sentir a entrada de mais um usuário.

Percebe-se, de uma maneira geral, que o método de subespaço com mode-

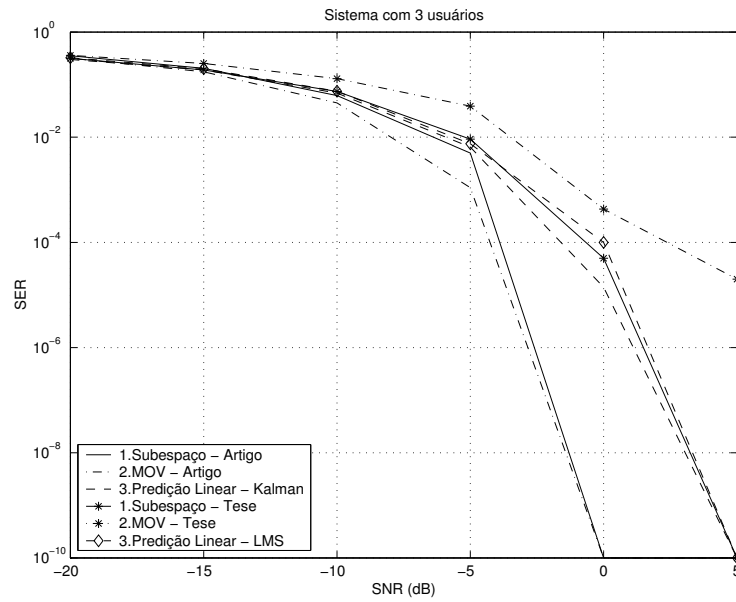


Figura 8.3: Comparação de desempenho dos algoritmos de equalização cega com 3 usuários no sistema.

lagem do artigo apresenta um desempenho superior aos demais na maior parte da faixa simulada de SNRs e de usuários. Um fator importante é, como visto a partir da seção 5.5, a desconsideração da ISI na modelagem feita pelo artigo [17]. Nota-se que, quando se utiliza a modelagem da equação (4.12), o desempenho do sistema cai sensivelmente. Lembra-se ainda que o método de subespaço foi desenvolvido para o uso com transmissão síncrona, o que foi usado, nesta seção. Como o algoritmo de subespaço utilizado é específico para o uso de BPSK, esse algoritmo tende a se aproveitar melhor dessas condições no seu funcionamento, enquanto os métodos CMV e de predição linear devem apresentar robustez quanto ao uso de outras constelações e ao assincronismo de transmissão. O peso disso no desempenho não pode ser desprezado, na medida em que se trabalha com algoritmos de equalização que dispensam o uso de seqüência de treinamento. Com isso, qualquer informação estatística sobre o sinal transmitido (como a constelação utilizada) e do sinal recebido (como o modelo de transmissão síncrona, que facilita a manutenção de boa parte da ortogonalidade das assinaturas dos usuários) são valiosas no projeto do receptor.



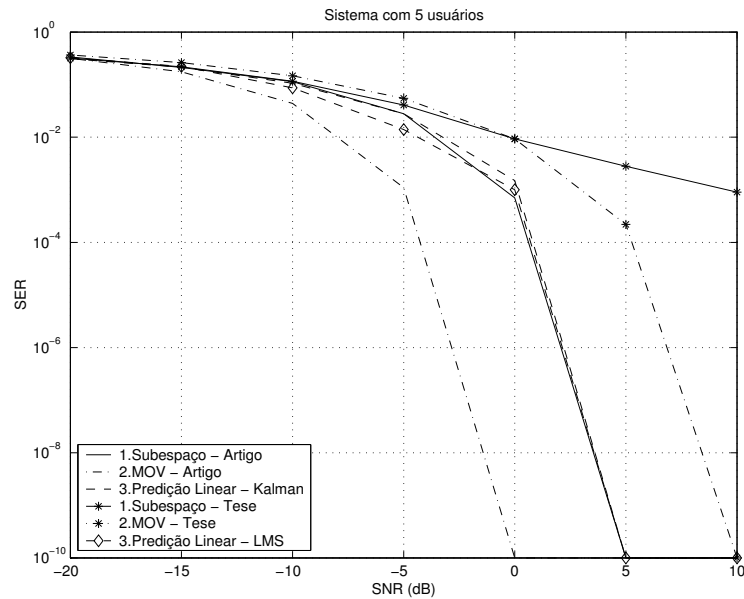


Figura 8.4: Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no sistema.

Nota-se ainda que, ao se excluir da comparação os algoritmos com modelagem “do artigo”, os algoritmos que usam predição linear apresentam o melhor desempenho, principalmente quando se aumenta o número de usuários.

### 8.4.2 Simulações de Canais sem Memória

A fim de que se pudesse visualizar melhor qual o efeito gerado pelo fato de o canal ter memória, foram feitas, com objetivo de comparação, simulações em que o canal se compõe de um filtro com um único coeficiente. Como não são utilizados sistemas de controle de potência, os canais foram normalizados para terem norma 1, e, com isso, não causam também distorções na amplitude do sinal. Na prática, o efeito da transmissão nesse tipo de ambiente é apenas o de somar as formas de onda de todos os usuários e também o AWGN presente no ambiente.

O objetivo principal desta subseção é estudar melhor as funcionalidades correspondentes a supressão de ruído e separação de usuários sem a presença de ISI, propiciando assim uma comparação mais detalhada dos algoritmos implementados

em relação a essas funcionalidades.

No caso do algoritmo CMV, houve uma necessidade de se mudar o valor do passo de convergência para 0,003 a fim de que se conseguisse chegar nas menores taxas de SER. Isso revela uma certa sensibilidade deste algoritmo quanto ao ajuste de parâmetros em relação aos demais.

A partir das tabelas 8.8 a 8.12, nota-se que, no caso do canal sem memória, a inclusão de mais usuários no sistema não influencia o desempenho do sistema. A causa disso é o uso de códigos de espalhamento e um modelo de transmissão síncrona de *uplink*. Como no caso síncrono os códigos permanecem praticamente ortogonais, a separação entre os usuários é praticamente total. O resíduo de não-ortogonalidade, no entanto, não gera erros adicionais pois usa-se ganhos de processamento de 31, a partir dos quais são usados 31 chips na detecção do símbolo em questão. Essa grande redundância de informação evita que a pequena não-ortogonalidade seja propagada para uma detecção errada. Quando se insere ISI, a seqüência de espalhamento deixa de ser a assinatura efetiva de cada usuário, que passa a ser a convolução do código de espalhamento com a resposta ao impulso do canal. Com isso, há uma perda de ortogonalidade entre as assinaturas dos usuários, resultando em perda de desempenho ao se aumentar o número de usuários do sistema.

Nota-se que foram apresentados resultados de apenas uma versão do algoritmo de subespaço. Isso ocorre pois as duas modelagens utilizadas (das equações (5.43) e (4.12)) tornam-se a mesma na ausência de múltiplos percursos.

A partir da figura 8.5, percebe-se que todos os métodos têm desempenho equivalente em toda a faixa de SNRs. O comportamento do algoritmo CMV é praticamente o mesmo tanto no caso de se usar modelagem da tese quanto do artigo. Uma explicação para esse comportamento é a quase ortogonalidade dos códigos de espalhamento do CDMA o que garante uma separação quase perfeita quando não há ISI e nem assincronismos. No entanto, a separação não é perfeita como se pode ver quando aumentamos a SNR. Nesse caso, a modelagem do artigo [21] já apresenta uma SER menor que a da tese, o que é resultado da ausência

Tabela 8.8: Resultados para o Algoritmo de Subespaço de Wang e Poor - Canais sem Memória.

<b>Algoritmo de Subespaço</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,295	0,288	0,290	0,292	0,291
<b>-15</b>	0,162	0,160	0,163	0,162	0,162
<b>-10</b>	0,039	0,039	0,040	0,040	0,040
<b>-5</b>	0,0006	0,00076	0,0009	0,0009	0,0010
<b>0</b>	0	0	0	0	0

Tabela 8.9: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais sem Memória - Modelagem do Artigo.

<b>Algoritmo CMV</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,290	0,290	0,292	0,293	0,292
<b>-15</b>	0,162	0,162	0,163	0,163	0,164
<b>-10</b>	0,040	0,040	0,040	0,039	0,040
<b>-5</b>	0,0009	0,0008	0,0011	0,0009	0,0008
<b>0</b>	0	0	0	0	0

Tabela 8.10: Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gu-nawan - Canais sem Memória.

<b>Algoritmo de Predição Linear - Kalman</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,289	0,290	0,290	0,290	0,289
<b>-15</b>	0,161	0,162	0,161	0,162	0,162
<b>-10</b>	0,040	0,040	0,041	0,041	0,040
<b>-5</b>	0,0011	0,0012	0,0010	0,0010	0,0012
<b>0</b>	0	0	0	0	0

Tabela 8.11: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Canais sem Memória - Modelagem da Tese.

<b>Algoritmo CMV</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,306	0,308	0,308	0,308	0,306
<b>-15</b>	0,173	0,172	0,172	0,172	0,173
<b>-10</b>	0,044	0,044	0,045	0,045	0,046
<b>-5</b>	0,0011	0,0011	0,0015	0,0013	0,0013
<b>0</b>	0	0	0	0	0

Tabela 8.12: Resultados para o Algoritmo de Predição Linear de Fan e Li - Canais sem Memória.

<b>Algoritmo de Predição Linear - LMS</b>					
	<b>Usuários</b>				
<b>SNR (dB)</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,303	0,304	0,301	0,302	0,302
<b>-15</b>	0,166	0,169	0,169	0,169	0,170
<b>-10</b>	0,042	0,042	0,043	0,043	0,043
<b>-5</b>	0,0026	0,0023	0,0025	0,0026	0,0028
<b>0</b>	0,00004	0,00019	0,000053	0,000095	0,000048
<b>5</b>	0	0	0	0	0

completa de interferência multiusuário no caso em que não há multipercursos, como pode ser percebido pela equação (6.38).

Nesta subseção nota-se que a função de separação entre usuários está intimamente ligada à eliminação de ISI. Por isso, no quadro 8.1 pode-se verificar inter-relações entre essas funcionalidades para cada um dos algoritmos. Em alguns casos, pode-se falar em duas etapas de equalização: supressão de ruído e estimação de canal, que englobaria separação de usuários e eliminação de ISI.

Percebe-se pelos dados desta seção que o algoritmo de predição linear, ao contrário do que se imaginava, não apresenta o melhor comportamento em relação à separação multiusuário. Na verdade seu desempenho foi equivalente aos demais, como pode-se perceber no caso em que há 5 usuários e alta razão sinal-ruído no sistema. Conclui-se que o método de predição linear revelou-se mais eficiente para separação entre usuários dentro de um ambiente com ISI.

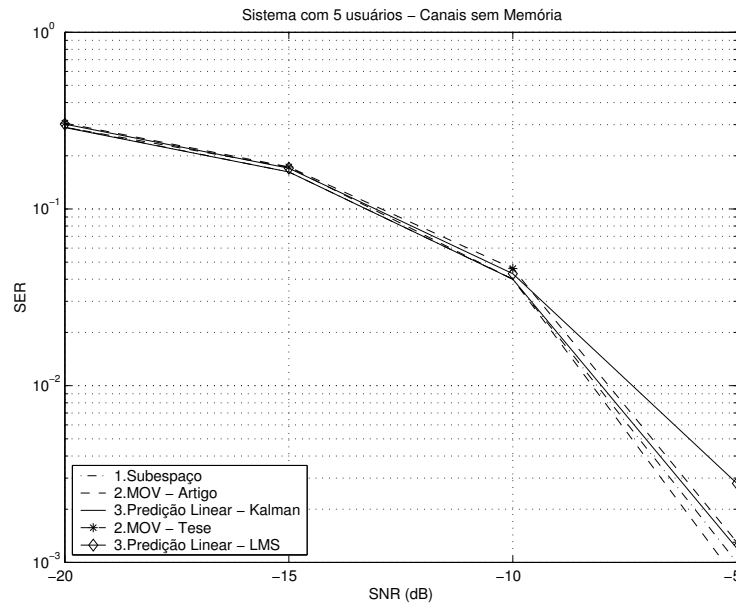


Figura 8.5: Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no sistema para canais sem memória.

### 8.4.3 Simulação de Recepção de Sinais Assíncronos

A fim de se simular a situação mais realista de uma transmissão de *uplink* assíncrona, foram realizadas simulações em que se introduziu um atraso aleatório para cada usuário que não fosse o usuário de interesse. Manteve-se com isso a hipótese de o receptor estar sincronizado com o sinal do usuário de interesse.

Nas simulações foram utilizados os canais apresentados nas equações 8.1 a 8.5. O sistema foi projetado de maneira que o usuário  $j$  tenha atraso de 1 chip em relação ao usuário  $j + 1$ . Os principais parâmetros de simulação seguem o descrito na subseção 8.4.1. Os resultados encontrados estão mostrados nas tabelas 8.13 a 8.18. Não está apresentado o caso com apenas 1 usuário, pois recairia no caso síncrono já visto, pois, por suposição desta subseção, o receptor é sincronizado com o usuário de interesse.

Nota-se que os resultados, em geral, apresentam uma pequena degradação em relação ao caso síncrono. Isso mostra o quanto os algoritmos são relativamente robustos quanto ao problema de assincronismo. Verifica-se ainda uma presença

Tabela 8.13: Resultados para o Algoritmo de Subespaço de Wang e Poor - Assíncrono - Equação (5.43 - Modelagem do Artigo).

<b>Algoritmo de Subespaço</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,316	0,319	0,304	0,309
<b>-15</b>	0,178	0,178	0,178	0,206
<b>-10</b>	0,060	0,070	0,079	0,090
<b>-5</b>	0,0034	0,010	0,028	0,058
<b>0</b>	0,0090	0,018	0,019	0,0043
<b>5</b>	0	0,00001	0,000013	0,0006
<b>10</b>	0	0	0	0

Tabela 8.14: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Assíncrono - Modelagem do Artigo.

<b>Algoritmo CMV</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,327	0,332	0,330	0,328
<b>-15</b>	0,183	0,184	0,183	0,183
<b>-10</b>	0,048	0,049	0,048	0,049
<b>-5</b>	0,0014	0,0016	0,0017	0,0014
<b>0</b>	0	0	0	0

Tabela 8.15: Resultados para o Algoritmo de Predição Linear de Liu, Teh e Gunawan - Assíncrono.

<b>Algoritmo de Predição Linear - Kalman</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,297	0,299	0,309	0,310
<b>-15</b>	0,182	0,190	0,202	0,204
<b>-10</b>	0,064	0,076	0,097	0,111
<b>-5</b>	0,0071	0,0087	0,032	0,043
<b>0</b>	0,00019	0,0005	0,0076	0,010
<b>5</b>	0	0	0,0020	0,0015
<b>10</b>	0	0	0	0,0008

Tabela 8.16: Resultados para o Algoritmo de Subespaço de Wang e Poor - Assíncrono - Equação (4.12 - Modelagem da Tese).

<b>Algoritmo de Subespaço</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,309	0,318	0,320	0,328
<b>-15</b>	0,193	0,217	0,224	0,226
<b>-10</b>	0,077	0,084	0,112	0,124
<b>-5</b>	0,0057	0,026	0,048	0,074
<b>0</b>	0,0007	0,0021	0,042	0,068
<b>5</b>	0	0,014	0,031	0,065
<b>10</b>	0	0,013	0,029	0,028
<b>15</b>	0	0,013	0,040	0,043
<b>20</b>	0	0,012	0,035	0,0078



Tabela 8.17: Resultados para o Algoritmo CMV de Xu e Tsatsanis - Assíncrono - Modelagem da Tese.

<b>Algoritmo CMV</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,332	0,348	0,356	0,354
<b>-15</b>	0,183	0,218	0,240	0,253
<b>-10</b>	0,051	0,089	0,124	0,161
<b>-5</b>	0,0018	0,014	0,039	0,071
<b>0</b>	0	0,00026	0,0033	0,018
<b>5</b>	0	0	0,00031	0,0027
<b>10</b>	0	0	0	0,00035
<b>15</b>	0	0	0	0,00003

Tabela 8.18: Resultados para o Algoritmo de Predição Linear de Fan e Li - Assíncrono.

<b>Algoritmo de Predição Linear - LMS</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,325	0,331	0,334	0,310
<b>-15</b>	0,213	0,221	0,227	0,204
<b>-10</b>	0,094	0,102	0,114	0,111
<b>-5</b>	0,019	0,019	0,028	0,043
<b>0</b>	0,0008	0,00063	0,0015	0,0016
<b>5</b>	0	0	0	0

pequena de uma certa inconsistência entre os dados, quando às vezes se aumenta o número de usuários e a SER cai, ou quando a SNR aumenta e a SER também aumenta. O método de subespaço parece um pouco mais sensível a esse tipo de fenômeno, o que pode ser uma consequência de se basear em estimações do subespaço do sinal, que pode ser particularmente sensível a algumas configurações de atraso.

Verifica-se que o desempenho de todos os algoritmos simulados apresenta-se bastante parecido para poucos usuários, mas o algoritmo de subespaço com modelagem do artigo tem um desempenho melhor quando se aumenta o número de usuários. Isso pode ser visto nas comparações para 2 e 5 usuários mostradas nas figuras 8.6 e 8.7, respectivamente. Observa-se também que o algoritmo de subespaço com modelagem da tese tem um desempenho bem pior que os demais, com a SER se mantendo praticamente constante mesmo com o aumento da SNR. Isso ocorre pois, além de se ter uma modelagem do CDMA diferente do colocado no artigo [17], fez-se uma adaptação do algoritmo para funcionar com uma transmissão assíncrona, diferentemente do artigo [17]. As muitas modificações devem dificultar a SER ser zerada para um número de usuários maior que 2.

Ao se comparar apenas os algoritmos com modelagem da tese, verifica-se que, para 2 usuários, o algoritmo CMV tem melhor desempenho, enquanto o de predição linear tem menores SER quando se aumenta o número de usuários no sistema.

## 8.5 Conclusões

Objetivo deste capítulo é proporcionar uma visão mais comparativa dos algoritmos apresentados nos capítulos 5 a 7 através da análise de simulações das três famílias de algoritmos. Em contraposição aos capítulos anteriores da tese, o presente capítulo tira o foco da descrição de algoritmos e aspectos teóricos, passando a dar mais ênfase nas diferenças de abordagens apresentadas por esses algoritmos.

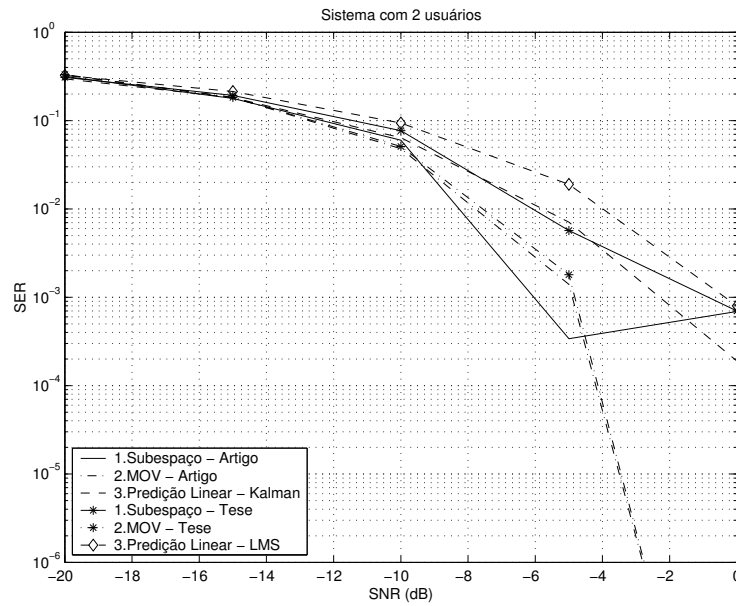


Figura 8.6: Comparação de desempenho dos algoritmos de equalização cega com 2 usuários no caso assíncrono.

As seções 8.2 e 8.3 mostram comparações mais teóricas, enquanto na seção 8.4 extraem-se informações a partir das simulações feitas.

Através das simulações aqui apresentada, foi possível analisar situações em que vários elementos teóricos diferentes se combinam, proporcionando uma visão sobre o funcionamento conjunto desses elementos. Verificou-se que as funções de eliminação de ruído e de separação entre usuários na ausência de ISI são mais eficientes para os algoritmos de subespaço e CMV que o de predição linear. Já a função de eliminação de ISI e a separação de usuários na presença de ISI é feita de maneira mais eficiente pelo algoritmo de predição linear. Para o caso assíncrono, o desempenho dos algoritmos CMV e predição linear foram próximos para poucos usuários, mas o algoritmo de predição linear é melhor quando se aumenta o número de usuários.

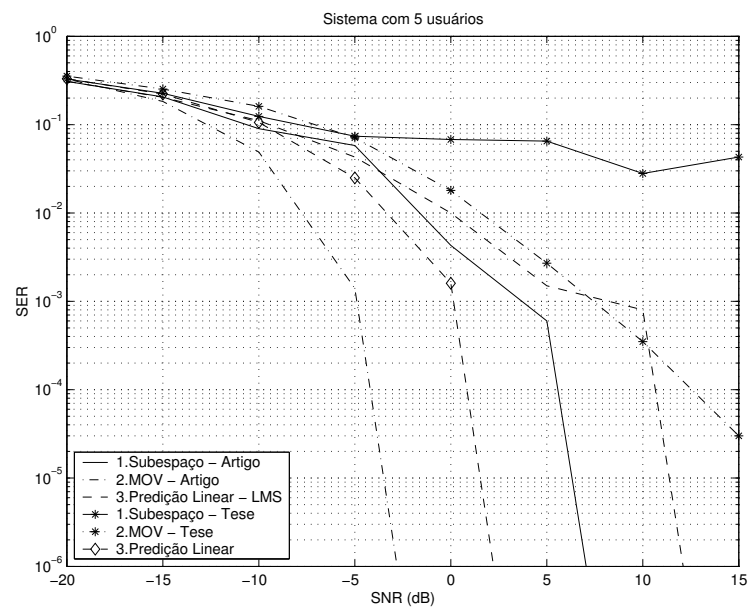


Figura 8.7: Comparação de desempenho dos algoritmos de equalização cega com 5 usuários no caso assíncrono.

# Capítulo 9

## Implementações com Atualização Seletiva

### 9.1 Introdução

Uma das idéias utilizadas para se diminuir a complexidade de um algoritmo adaptativo é o conceito de atualização seletiva dos coeficientes. A atualização seletiva consiste na definição de um critério a partir do qual apenas em algumas iterações os coeficientes do filtro adaptativo são atualizados.

O conceito de identificação *Set-Membership* está dentro do paradigma de atualização seletiva, tendo-se estabelecido na área de identificação de sistemas, dentro de algumas restrições [35]. A partir do artigo [36], de Gollamudi *et. al*, estendeu-se o conceito de filtragem *Set-Membership* para um grande conjunto de problemas adaptativos.

O objetivo deste capítulo é o de apresentar uma das inovações conseguidas nesta tese, que é o de incluir os conceitos de filtragem *Set-Membership* nos três algoritmos apresentados nos capítulos 5, 6 e 7.

Nas seção 9.2 será mostrado o conceito geral de filtragem *set-membership*. A seguir, na seção 9.3, é apresentada a implementação do algoritmo de subespaço

usando SM. Já a seção 9.4 mostra a implementação do algoritmo CMV usando os conceitos de SM. A seção 9.5 apresenta a implementação SM do algoritmo de predição linear. Já a seção 9.6 apresenta os resultados de simulações para as versões SM dos algoritmos.

## 9.2 Filtragem *Set-Membership*

Considerando-se um filtro adaptativo com entrada  $x(k)$ , saída  $y(k)$ , sinal desejado  $d(k)$  e coeficientes  $\mathbf{w}(k)$ . Na abordagem tradicional de filtragem adaptativa, os coeficientes do filtro são atualizados de maneira a minimizar uma dada função objetivo  $\mathcal{F}(\cdot)$ . No caso dos algoritmos SM, a atualização é feita no sentido de se manter a função objetivo sempre limitada por um valor  $\gamma$  [36]. A solução ótima, pelos critérios SM, formam um conjunto e não apenas um ponto, como no caso do processamento adaptativo tradicional. Formalizando, o critério de otimização SM é tal que:

$$|\mathcal{F}(\mathbf{w}_{\text{opt}})|^2 \leq \gamma^2 \quad (9.1)$$

O conjunto de coeficientes que satisfazem à equação (9.1) é chamado de conjunto de viabilidade  $\Theta$  [36]. O conjunto  $\Theta$  é bastante dependente da escolha do parâmetro  $\gamma$ . À medida em que se diminui o valor de  $\gamma$ , a região de elementos de  $\Theta$  também decresce. A diminuição desse parâmetro pode, em uma situação limite, levar a uma situação em que  $\Theta = \emptyset$ . Nesse caso não há um vetor de coeficientes  $\mathbf{w}$  que atinja ao critério ótimo SM dado por (9.1). O critério de otimização SM apenas garante que a função objetivo se mantenha abaixo do limiar  $\gamma$ , portanto, ao se aumentar  $\gamma$ , o algoritmo SM não garante mais uma minimização adequada para a função objetivo. Portanto, a escolha do valor de  $\gamma$  deve obedecer a esse compromisso, sendo bastante dependente do problema em questão.

## 9.3 Método de Subespaço Usando SM

No caso do algoritmo de subespaço descrito em [17], a atualização dos filtros é feita por um algoritmo do tipo gradiente, definido pela equação (5.40):

$$\mathbf{w}(k) = \mathbf{w}(k-1) - \mu[\mathbf{B}(k-1)\mathbf{w}(k-1) + (\|\mathbf{w}(k-1)\|^2 - 1)\mathbf{w}(k-1)] \quad (9.2)$$

Essa equação foi gerada através da minimização da seguinte função objetivo:

$$\min_{\mathbf{w}} \|\mathbf{D}\mathbf{w}\|^2 \quad \text{restrito a} \quad \|\mathbf{w}\| = 1 \quad (9.3)$$

Transformando-se o problema de otimização com restrições num equivalente sem restrições, obtém-se:

$$\mathcal{F}(\mathbf{w}) \triangleq \frac{1}{2}\mathbf{w}^H\mathbf{D}^H\mathbf{D}\mathbf{w} + \frac{1}{4}(\mathbf{w}\mathbf{w}^H - 1) \quad (9.4)$$

Com isso, define-se um critério para atualização dos coeficientes, de acordo com a teoria SM:

- Atribui-se a  $\mathcal{F}(\mathbf{w}, k)$  no instante  $k$  o valor dado pela equação (9.4)
  - Se  $|\mathcal{F}(\mathbf{w}, k)|^2 > \gamma^2$ ,  $\mathbf{w}(k)$  é atualizado de acordo com (9.2).
  - Caso contrário,  $\mathbf{w}(k) = \mathbf{w}(k-1)$ .

## 9.4 Método CMV Usando SM

No caso do método CMV, há duas sub-bandas onde se aplica o SM. Em ambas, o erro do adaptativo é dado pelo parâmetro

$$\mathbf{e}(k) = \mathbf{r}^H(k)\mathbf{C}_j^H(\mathbf{C}_j^H\mathbf{C}_j)^{-1}\mathbf{g}_j(k) - \mathbf{r}^H(k)\mathbf{C}_j^H\mathbf{w}_{j,l}(k) \quad (9.5)$$

onde  $\mathbf{r}(k)$  é o vetor recebido no instante  $k$ ,  $\mathbf{C}_j$  é a matriz com as seqüências de espalhamento atrasadas em suas colunas dada pela equação (6.5), e  $\mathbf{g}(k)$  e  $\mathbf{w}_{j,l}(k)$  são os filtros adaptativos de cada sub-banda.

O critério de atualização de cada sub-banda é dado por:

- Compara-se o valor de  $e(k)$  dado pela equação (9.5) com  $\gamma$ .
  - Se  $e^2(k) > \gamma^2$ , então  $\mathbf{g}(k)$  é atualizado de acordo com a equação 6.34 e  $\mathbf{w}_{j,l}(k)$  pela equação (6.33).
  - Caso contrário,  $\mathbf{g}(k) = \mathbf{g}(k - 1)$  e  $\mathbf{w}_{j,l}(k) = \mathbf{w}_{j,l}(k - 1)$ .

## 9.5 Método de Predição Linear usando SM

A implementação do método de predição linear usando SM foi feita a partir da modelagem LMS originalmente feita em [27] e [30], de Fan e Li, e detalhada no Capítulo 7, mais especificamente na seção 7.5. A expressão de erro do sistema foi encontrada a partir da equação (7.37):

$$\min \|\mathbf{f}_j^H \mathbf{y}_1\|^2 \quad \text{sujeito a} \quad \mathbf{f}_j^H \mathbf{u} = 1 \quad (9.6)$$

Como as próprias atualizações dos coeficientes garantem que  $\mathbf{f}_j^H \mathbf{u} = 1$ , então basta usar como parâmetro de erro a expressão  $\|\mathbf{f}_j^H \mathbf{y}_1\|^2$ .

O critério de atualização dos coeficientes do filtro adaptativo torna-se então:

- Se  $\|\mathbf{f}_j^H \mathbf{y}_1\|^2 > \gamma^2$ ,  $\mathbf{f}(k)$  é atualizado de acordo com (7.38).
- Caso contrário,  $\mathbf{f}(k) = \mathbf{f}(k - 1)$ .

## 9.6 Resultados de Simulações com SM

### 9.6.1 Resultados para o Subespaço

Como foi visto nos capítulos 5 e 8, há duas modelagens para o algoritmo de subespaço. A modelagem do artigo [17] não coloca ISI no sistema, gerando uma vantagem ilusória quanto à eliminação do efeito do canal com memória. Então o mais correto seria a utilização do SM no algoritmo com modelagem da tese. No entanto, os resultados do Capítulo 8 para o sistema assíncrono com modelagem



da tese mostram um mau funcionamento do algoritmo. Com isso, foi simulado o algoritmo de subespaço com modelagem da tese e transmissão síncrona para 3 usuários, na faixa de SNR entre -10 dB e 5 dB.

Foi simulado um sistema CDMA num canal sem memória. Os demais parâmetros usados nas simulações foram:

- $\mu = 0,007$ .
- $\alpha = 1$ .
- 5 execuções no *ensemble*.
- 20.000 símbolos transmitidos.
- Cálculo da SER iniciado após 10.000 símbolos.
- 3 usuários no sistema.

Os resultados das simulações para diferentes valores de  $\gamma$  estão representados na tabela 9.1. Verifica-se que para valores menores de  $\gamma$  a taxa de erro de símbolos diminui. No entanto, um dos objetivos do algoritmo SM é o de diminuir a taxa de atualização dos coeficientes do sistema. Então, na tabela 9.2 pode-se ver a razão entre o número de iterações do sistema em que os coeficientes foram atualizados em relação ao número total de iterações do sistema. Nota-se claramente o compromisso entre o número de iterações com atualização e a taxa de erro de símbolos. A presença de um mecanismo de decisão sobre a atualização do sistema acrescenta mais um componente de aleatoriedade ao processo.

A partir da figura 9.2 e das tabelas 9.1 e 9.2, percebe-se que a diminuição nas atualizações do algoritmo adaptativo, através do uso de SM, leva a um aumento da SER. A figura 9.2 mostra que, neste exemplo, para um limiar  $\gamma = 0,1$ , há um bom compromisso entre quantidade de atualizações e SER, pois para a SER ainda se encontrou zero enquanto a taxa de atualização para 5 dB está em 0,187. Pelos dados da Figura 9.1, percebe-se que, para o caso de SNR = 0 dB, o melhor

Tabela 9.1: Resultados para o Algoritmo de Subespaço de Wang e Poor - SM - Modelagem do Artigo.

<b>Algoritmo de Subespaço - SER</b>						
	$\gamma$					
<b>SNR(dB)</b>	<b>0</b>	<b>0,01</b>	<b>0,05</b>	<b>0,1</b>	<b>0,25</b>	<b>0,5</b>
<b>-10</b>	0,040	0,073	0,079	0,093	0,174	0,178
<b>-5</b>	0,009	0,0099	0,010	0,019	0,173	0,166
<b>0</b>	0	0,0001	0,00012	0,0006	0,029	0,200
<b>5</b>	0	0	0	0	0,112	0,187

Tabela 9.2: Resultados para o Algoritmo de Subespaço de Wang e Poor - SM - Modelagem do Artigo.

<b>Algoritmo de Subespaço</b>						
<b>Taxa de Atualização (%)</b>						
	$\gamma$					
<b>SNR(dB)</b>	<b>0</b>	<b>0,01</b>	<b>0,05</b>	<b>0,1</b>	<b>0,25</b>	<b>0,5</b>
<b>-10</b>	100	97,9	34,5	21,1	3,2	0,71
<b>-5</b>	100	90,3	29,8	17,2	2,5	0,58
<b>0</b>	100	79,2	20,0	10,9	2,1	0,64
<b>5</b>	100	77,7	18,7	14,4	2,3	0,65

compromisso também é atingido para o limiar de 0,05, onde a SER é igual a 0,00012 e a taxa de atualização é de 20%.

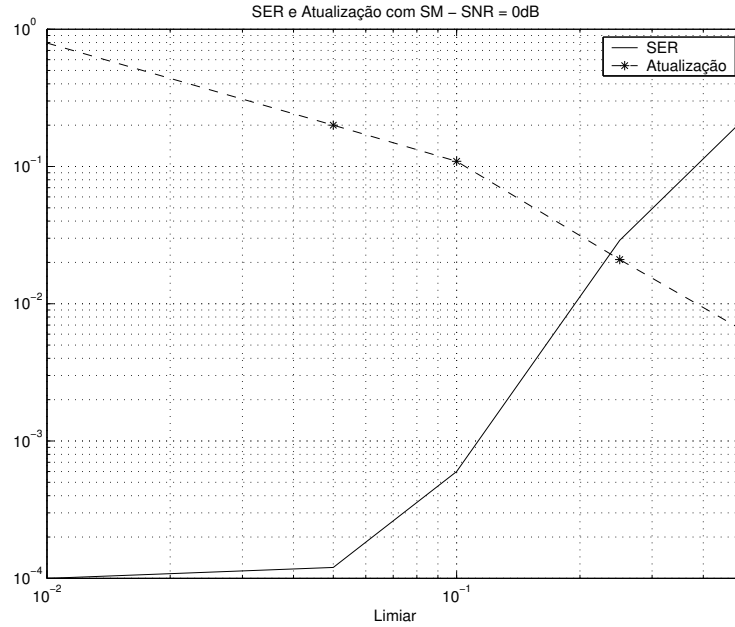


Figura 9.1: SER e taxa de atualização para SNR = 0 dB para o algoritmo de subespaço.

### 9.6.2 Resultados Usando CMV

Desta vez, foi usada uma implementação assíncrona com os canais (8.1) a (8.5) e, mais uma vez, 3 usuários. O demais parâmetros são os seguintes.

- $\mu_g = 0,015$ .
- $\mu_u = 10^{-3}$ .
- 8 execuções no *ensemble*.
- 20.000 símbolos transmitidos.
- Cálculo da SER iniciado após 10.000 símbolos.

Os resultados das simulações para diversos valores de  $\gamma$  estão representados na tabela 9.3. Já a taxa de atualização para cada valor de  $\gamma$  pode ser visto a partir

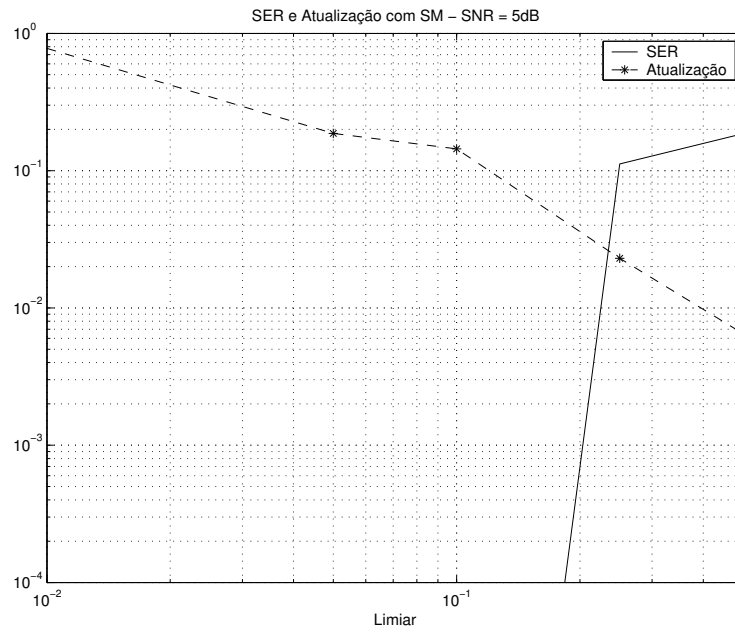


Figura 9.2: SER e taxa de atualização para SNR = 5 dB para o algoritmo de subespaço.

da tabela 9.4. As figuras 9.3 e 9.4 mostram as curvas de SER e de atualização para 0 dB e 5 dB, respectivamente. Nota-se, nos dois casos, que a escolha do parâmetro  $\gamma = 0,5$  representa um bom compromisso entre o aumento da SER e a diminuição na atualização do sistema. No caso de SNR = 0 dB, para  $\gamma = 0,5$ , os coeficientes do filtro são atualizados em somente 47,7% das iterações enquanto a SER sobe apenas de 0,00026 para 0,00047. Para SNR = 5 dB, ainda com  $\gamma = 0,5$ , os coeficientes do filtro sofrem atualização em 51,6% das iterações, enquanto a SER sobe de 0 para 0,000013.

### 9.6.3 Resultados usando Predição Linear

Foram feitas simulações usando uma implementação assíncrona do algoritmo de predição linear com atualização LMS dos coeficientes do detector conforme descrito na seção 7.5. Os principais parâmetros usados na simulação foram:

- $\mu = 0,001$

Tabela 9.3: Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM.

Algoritmo CMV - SER						
	$\gamma$					
SNR(dB)	0	0,1	0,2	0,5	0,7	1
-10	0,089	0,091	0,091	0,090	0,090	0,096
-5	0,014	0,014	0,014	0,015	0,024	0,052
0	0,00026	0,00027	0,00024	0,00047	0,0054	0,021
5	0	0	0	0,000013	0,00008	0,068

Tabela 9.4: Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM.

Algoritmo CMV - Taxa de Atualização (%)						
	$\gamma$					
SNR(dB)	0	0,1	0,2	0,5	0,7	1
-10	100	92,0	83,8	55,9	35,6	11,1
-5	100	96,6	91,9	57,1	19,8	2,4
0	100	99,4	97,9	47,7	11,1	0,35
5	100	99,8	99,4	51,6	2,7	0,065

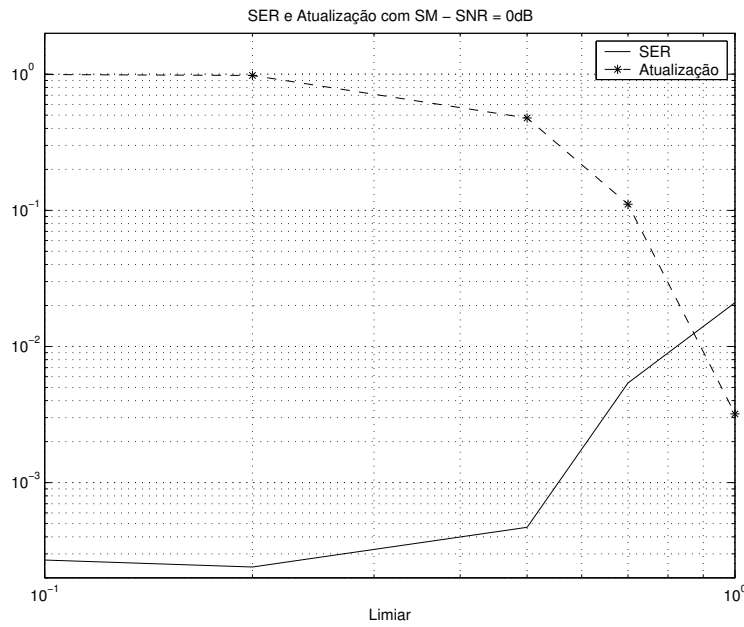


Figura 9.3: SER e taxa de atualização para SNR = 0 dB para o algoritmo CMV.

- 8 execuções no *ensemble*.
- 20.000 símbolos transmitidos.
- Cálculo da SER iniciado após 10.000 símbolos.

Realizaram-se então várias simulações para diversos valores de  $\gamma$  como mostrado nas tabelas 9.5 e 9.6 e nas figuras 9.5 e 9.6. Percebe-se que o algoritmo de predição linear implementado apresenta uma sensibilidade bem maior que o algoritmo CMV quanto à diminuição na taxa de atualização dos coeficientes do detector. Para um ambiente com SNR = 0 dB, o valor considerado ideal de  $\gamma$  é de 0,25, pois a SER sobe de 0,00063 para 0,00073 enquanto a taxa de atualização fica em 92,5%. Já para SNR = 5 dB, com  $\gamma = 0,1$ , a taxa de atualização é bem próxima de 100%, enquanto a taxa de atualização fica em 92,5%.

Tabela 9.5: Resultados para o Algoritmo de Predição Linear de Fan e Li - SM.

<b>Algoritmo de Predição Linear - SER</b>						
	$\gamma$					
<b>SNR(dB)</b>	<b>0</b>	<b>0,01</b>	<b>0,1</b>	<b>0,25</b>	<b>0,5</b>	<b>0,75</b>
<b>-10</b>	0,109	0,109	0,112	0,119	0,133	0,143
<b>-5</b>	0,019	0,040	0,054	0,051	0,064	0,078
<b>0</b>	0,00063	0,0057	0,0068	0,0073	0,013	0,032
<b>5</b>	0	0,00007	0,00008	0,0005	0,0043	0,016

Tabela 9.6: Resultados para o Algoritmo de Predição Linear de Fan e Li - SM.

<b>Algoritmo de Predição Linear</b>						
<b>Taxa de Atualização (%)</b>						
	$\gamma$					
<b>SNR(dB)</b>	<b>0</b>	<b>0,01</b>	<b>0,1</b>	<b>0,25</b>	<b>0,5</b>	<b>0,75</b>
<b>-10</b>	100	99,2	92,0	79,2	57,9	38,5
<b>-5</b>	100	99,5	94,0	83,6	58,9	34,5
<b>0</b>	100	99,8	98,2	92,5	63,2	30,7
<b>5</b>	100	99,9	99,9	96,8	67,0	29,3

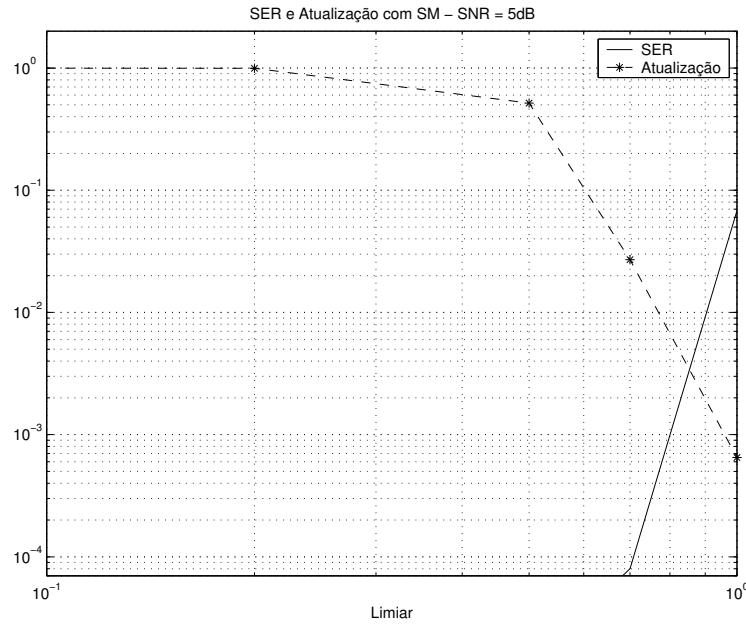


Figura 9.4: SER e taxa de atualização para  $\text{SNR} = 5$  dB para o algoritmo CMV.

## 9.7 Conclusões

O objetivo deste capítulo foi mostrar a incorporação de técnicas de atualização adaptativa seletiva aos algoritmos apresentados nos capítulos 5 a 7. Isso constitui o primeiro conjunto de inovações apresentados por esta tese. As novas implementações são baseadas no conceito de filtragem *Set-Membership*, mostrado nos artigos [37] e [36]. O objetivo do SM é diminuir a complexidade do algoritmo adaptativo, de modo que a filtragem só ocorra em determinadas situações.

A seção 9.3 descreve umas alterações no algoritmo de subespaço a fim de se adequar ao SM. É feita a descrição do algoritmo CMV na seção 9.4. Na seção 9.5 é apresentada uma implementação LMS do algoritmo de predição linear usando SM. A seguir, na seção 9.6, são mostrados os resultados das simulações para diferentes valores de parâmetro  $\gamma$  encontrando-se um valor considerado ideal para o parâmetro  $\gamma$  do SM para cada algoritmo.

Em geral, pode-se concluir a partir deste capítulo que é possível se diminuir bastante o número de iterações do sistema sem aumentar consideravelmente a SER do sistema para os algoritmos de subespaço e CMV. Já o algoritmo de predição



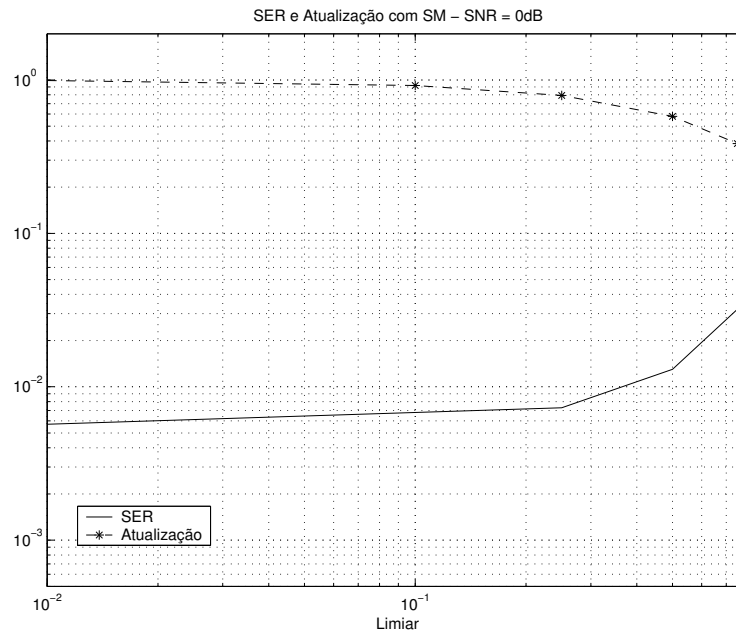


Figura 9.5: SER e taxa de atualização para  $\text{SNR} = 0$  dB para o algoritmo de predição linear.

linear revelou-se bastante sensível quanto à diminuição na taxa de atualização. Nota-se, no entanto, que mesmo nas situações em que há uma grande sensibilidade da SER em relação às atualizações do sistema, o desempenho do algoritmo com SM será, no caso extremo  $\gamma = 0$ , igual ao desempenho do algoritmo sem SM. Por isso, conclui-se que o uso do algoritmo SM é válido em todas as situações, principalmente pela liberdade proporcionada pelo parâmetro  $\gamma$  no compromisso entre SER e complexidade do algoritmo.

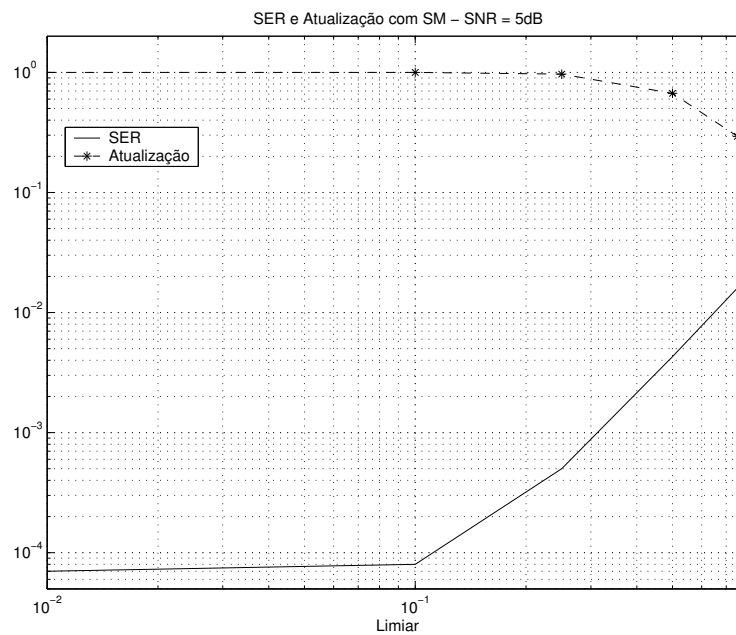


Figura 9.6: SER e taxa de atualização para SNR = 5 dB para o algoritmo de predição linear.

# Capítulo 10

## Implementações com Reuso de Dados

### 10.1 Introdução

O algoritmo LMS tradicional tornou-se bastante popular devido à simplicidade de sua implementação e a sua reduzida complexidade computacional. No entanto, uma das características indesejáveis do LMS é a lenta convergência do algoritmo [8]. Em [8] são apresentadas várias modificações no algoritmo LMS tradicional a fim de aumentar sua velocidade de convergência. Uma das soluções para aumentar a velocidade de convergência do algoritmo é através do reuso de dados, no qual cada entrada do sistema é processada em várias iterações. Este capítulo tem como foco o algoritmo de projeção afim, que visa a aumentar a velocidade de convergência do sistema através do reuso de dados. Esse algoritmo, no entanto, aumenta o desajuste do sistema, piorando as suas taxas de erro.

Neste capítulo é apresentada a incorporação do algoritmo de projeção afim aos algoritmos de equalização cega descritos nos capítulos 6 e 7. Além disso, para resolver a questão do desajuste, foi implementada uma versão do algoritmo de projeção afim que agrega o conceito de reuso de dados apresentado no capítulo 9. Esse algoritmo é conhecido como SMAP (*Set-Membership Affine Projection*),

apresentado no artigo [38].

A seção 10.2 aborda a teoria básica de filtragem adaptativa por projeção afim. A seção 10.3 apresenta uma modificação no algoritmo CMV para que as atualizações dos coeficientes dos filtros de cada sub-banda seja feita através do algoritmo de projeção afim. A seção 10.4 procura unir a parte da filtragem seletiva ao reuso de dados ao descrever o algoritmo SMAP. Por fim, a seção 10.5 mostra o algoritmo CMV com atualizações SMAP e a seção 10.9 apresenta os resultados das simulações referentes aos algoritmos apresentados neste capítulo.

## 10.2 Algoritmo de Projeção Afim

Um dos algoritmos mais conhecidos que empregam o reuso de dados é o algoritmo de projeção afim (APA - *affine projection algorithm*). Apesar de apresentar uma velocidade maior de convergência do algoritmo em relação ao LMS, o APA tem a desvantagem de aumentar o desajuste do sistema, ou seja, é maior o valor do erro residual após a convergência do filtro adaptativo [8]

No APA, a entrada de dados do filtro não é mais um escalar em cada instante de tempo. A entrada agora consiste em um vetor com as  $L$  entradas mais recentes, em cada instante de tempo, sendo  $L$  um parâmetro de projeto.

Considera-se um filtro adaptativo  $\mathbf{w}(k)$  de ordem  $N$ . Seja  $\mathbf{X}(k)$  a matriz que contém em suas colunas as  $L$  entradas mais recentes do sistema [8]:

$$\mathbf{X}(k) = \begin{bmatrix} x(k) & x(k-1) & \dots & x(k-L+1) & x(k-L) \\ x(k-1) & x(k-2) & \dots & x(k-L) & x(k-L-1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x(k-N) & x(k-N-1) & \dots & x(k-L-N+1) & x(k-L-N) \end{bmatrix} \quad (10.1)$$

e seja  $\mathbf{d}(k)$  o vetor contendo as  $L$  amostras mais recentes do sinal desejado.

A função objetivo do APA é dada por [8]:

$$\begin{aligned} \min \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \quad \text{sujeito a} \\ \mathbf{d}(k) - \mathbf{X}^H(k)\mathbf{w}(k+1) = \mathbf{0} \end{aligned} \quad (10.2)$$

ou seja, o vetor de coeficientes  $\mathbf{w}(k+1)$  deve-se manter o mais próximo do vetor de coeficientes  $\mathbf{w}(k)$  da iteração anterior, enquanto o erro *a posteriori* é mantido nulo.

A função de otimização com restrições (10.2) pode ser transformada num problema sem restrições através do uso de multiplicadores de Lagrange vetoriais  $\boldsymbol{\lambda}$  [8]:

$$\mathcal{F}(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 + \boldsymbol{\lambda}^H(k) [\mathbf{d}(k) - \mathbf{X}^H(k)\mathbf{w}(k+1)] \quad (10.3)$$

Derivando-se essa equação e igualando-se a zero, chega-se ao valor de  $\boldsymbol{\lambda}$ :

$$\boldsymbol{\lambda} = (\mathbf{X}^H \mathbf{X})^{-1} \mathbf{e}(k) \quad (10.4)$$

onde  $\mathbf{e}(k)$  é dado por:

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^H(k)\mathbf{w}(k) \quad (10.5)$$

Aplicando-se o valor de  $\boldsymbol{\lambda}$  da equação (10.4) na expressão (10.3) e acrescentando-se o passo de atualização  $\mu$ , obtém-se a expressão geral de atualização adaptativa do APA:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \mathbf{X}(k) (\mathbf{X}^H(k)\mathbf{X}(k))^{-1} \mathbf{e}(k) \quad (10.6)$$

## 10.3 Algoritmo CMV com Atualizações Feitas por APA

Foi feita uma modificação no algoritmo CMV descrito no Capítulo 6 para que a atualização dos coeficientes em cada sub-banda fosse feita de acordo com o APA. Dessa maneira foram criadas as matrizes  $\tilde{\mathbf{Y}}$  e  $\bar{\mathbf{Y}}$  dadas por:

$$\tilde{\mathbf{Y}}(k) = \begin{bmatrix} \tilde{\mathbf{y}}(k) & \tilde{\mathbf{y}}(k-1) & \dots & \tilde{\mathbf{y}}(k-L+1) \end{bmatrix} \quad (10.7)$$

$$\bar{\mathbf{Y}}(k) = (\mathbf{C}^H \mathbf{C})^{-1} \begin{bmatrix} \bar{\mathbf{y}}(k) & \bar{\mathbf{y}}(k-1) & \dots & \bar{\mathbf{y}}(k-L+1) \end{bmatrix} \quad (10.8)$$

onde:

$$\tilde{\mathbf{y}}(k) = \mathbf{C}^H \mathbf{r}(k) \quad (10.9)$$

$$\bar{\mathbf{y}}(k) = \mathbf{C}_n^H \mathbf{r}(k) \quad (10.10)$$

Definiu-se também a variável  $\mathbf{e}(k)$ :

$$\mathbf{e}(k) = \tilde{\mathbf{Y}}^H(k) \mathbf{g}(k) - \bar{\mathbf{Y}}^H(k) \mathbf{w}_{j,l}(k) \quad (10.11)$$

Com isso, a atualização dos coeficientes dos filtros adaptativos de cada sub-banda da estrutura GSC do algoritmo CMV passa a ser feita por:

$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) + \mu_u \bar{\mathbf{Y}}(k) \left( \bar{\mathbf{Y}}^H(k) \bar{\mathbf{Y}}(k) \right)^{-1} \mathbf{e}(k) \quad (10.12)$$

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g \tilde{\mathbf{Y}}(k) \left( \tilde{\mathbf{Y}}^H(k) \tilde{\mathbf{Y}}(k) \right)^{-1} \mathbf{e}(k) \quad (10.13)$$

$$\mathbf{g}(k+1) := \frac{\mathbf{g}(k+1)}{\|\mathbf{g}(k+1)\|} \quad (10.14)$$

A implementação do algoritmo CMV com atualizações APA está representada em pseudo-código através do algoritmo 10.1 no fim do capítulo.

## 10.4 SM com Atualizações por APA

No artigo [38] é descrito um algoritmo adaptativo que combina em suas atualizações o reuso de dados (mais especificamente APA) com a atualização seletiva de dados através de filtragem SM. Este novo algoritmo é conhecido como SMAP - *Set-Membership Affine Projection*. Seu objetivo é resolver a questão do aumento do desajuste gerado pelo APA através do uso de filtragem SM.

A função objetivo a ser minimizada é definida pela seguinte expressão [38]:

$$\min \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \quad \text{sujeito a} \quad \mathbf{d}(k) - \mathbf{X}^H(k) \mathbf{w}(k+1) = \mathbf{g}(k) \quad (10.15)$$

onde  $\mathbf{w}(k)$  é o filtro adaptativo,  $\mathbf{d}(k)$  é o sinal desejado,  $\mathbf{X}(k)$  é a matriz de sinais de entrada definida pela equação (10.1). O vetor  $\mathbf{g}(k)$  engloba as restrições do sistema obedecendo a  $|g_i(k)| \leq \gamma$ , sendo  $\gamma$  o limiar da filtragem SM e  $g_i(k)$  um elemento qualquer do vetor  $\mathbf{g}(k)$ , como definido na seção 9.2.

Utilizando-se multiplicadores de Lagrange, o problema de otimização com restrições transforma-se na seguinte otimização sem restrições [38]:

$$\mathcal{F}(\mathbf{w}) = \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 + \boldsymbol{\lambda}^H [\mathbf{d}(k) - \mathbf{X}^H(k)\mathbf{w}(k+1) - \mathbf{g}(k)] \quad (10.16)$$

Obtém-se o valor de  $\gamma$  minimizando-se a expressão (10.16):

$$\frac{\partial \mathcal{F}}{\partial \mathbf{w}} = 0 \quad (10.17)$$

$$\Rightarrow \boldsymbol{\lambda} = \mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)\boldsymbol{\lambda} \quad (10.18)$$

$$\Rightarrow \mathbf{X}^H(k)\mathbf{w}(k+1) = \mathbf{X}^H(k)\mathbf{w}(k) + \mathbf{X}^H(k)\mathbf{X}(k)\boldsymbol{\lambda} \quad (10.19)$$

Usando-se a equação (10.15) chega-se a [38]:

$$\mathbf{X}^H(k)\mathbf{X}(k)\boldsymbol{\lambda} = \mathbf{d}(k) - \mathbf{X}^H(k)\mathbf{w}(k) - \mathbf{g}(k) = \mathbf{e}(k) - \mathbf{g}(k) \quad (10.20)$$

O vetor  $\mathbf{e}(k)$  tem como elementos os erros *a posteriori*  $e_i(k)$  do sistema medidos nas  $L$  últimas iterações, sendo  $L$  a quantidade de reuso de dados do sistema. Com isso, a equação de atualização do filtro adaptativo é dada por:

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mu \mathbf{X}(k) (\mathbf{X}^H(k)\mathbf{X}(k))^{-1} (\mathbf{e}(k) - \mathbf{g}(k)) & \text{se } |e(k)| > \gamma \\ \mathbf{w}(k) & \text{caso contrário} \end{cases} \quad (10.21)$$

onde  $\mu$  é um parâmetro introduzido como passo de atualização do filtro adaptativo.

No artigo [38] é apresentada uma escolha para o parâmetro  $\mathbf{g}(k)$ . Essa escolha se baseia na restrição  $|g_i(k)| \leq \gamma$ . A restrição é respeitada se forem escolhidos  $g_i(k) = e_i(k)$  para  $i \neq 0$ , ou seja, para todos os elementos de  $\mathbf{g}(k)$  exceto o primeiro. Para o primeiro é sugerido o valor  $g_0(k) = \gamma \text{sign}(e_0(k))$ , onde  $\text{sign}(\cdot)$  representa a função sinal, com valor 1 caso seu argumento seja maior que zero, 0 caso o argumento seja zero e  $-1$  caso contrário. A atualização do filtro adaptativo passa a ser feita a partir de:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \mathbf{X}(k) (\mathbf{X}^H(k)\mathbf{X}(k))^{-1} \alpha(k) \mathbf{e}(k) \mathbf{t}^H \quad (10.22)$$

$$\alpha(k) = \begin{cases} 1 - \frac{\gamma}{|e_0(k)|} & \text{se } |e_0(k)| > \gamma \\ 0 & \text{caso contrário} \end{cases} \quad (10.23)$$

onde  $\mathbf{t} = [1 \ 0 \ \dots \ 0]^T$ .

## 10.5 Algoritmo CMV com Atualizações por SMAP

A partir do algoritmo CMV com atualizações de coeficientes por APA apresentado na seção 10.3 e no quadro 10.1, foi aplicado o método SM para atualização seletiva dos coeficientes.

O erro nas sub-bandas segue a equação (9.5) da seção 9.4 apenas com modificações para estar compatível com o caso de reuso de dados. Com isso, a definição do vetor de erros passa a ser a da expressão (10.11), ou seja:

$$\mathbf{e}(k) = \tilde{\mathbf{Y}}^H(k)\mathbf{g}(k) - \bar{\mathbf{Y}}^H(k)\mathbf{w}_{j,l}(k) \quad (10.24)$$

A regra de atualização do sistema usada é:

- Compara-se o valor de  $e_0(k)$  dado pela equação (10.24) com  $\gamma$ .

– Se  $e_0^2(k) > \gamma^2$ , então  $\alpha(k)$  é atualizado de acordo com

$$\alpha(k) = 1 - \frac{\gamma}{|e_0(k)|} \quad (10.25)$$

– Caso contrário:  $\alpha(k) = 0$ .

- Por fim, a atualização dos coeficientes nas sub-bandas é dada por:

$$\mathbf{g}(k+1) = \mathbf{g}(k) + \mu_g \tilde{\mathbf{Y}}(k) \left( \tilde{\mathbf{Y}}^H(k) \tilde{\mathbf{Y}}(k) \right)^{-1} \alpha(k) \mathbf{e}(k) \mathbf{t}^H \quad (10.26)$$

$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) + \mu_u \bar{\mathbf{Y}}(k) \left( \bar{\mathbf{Y}}^H(k) \bar{\mathbf{Y}}(k) \right)^{-1} \alpha(k) \mathbf{e}(k) \mathbf{t}^H \quad (10.27)$$

## 10.6 Algoritmo de Predição Linear numa Estrutura GSC

A implementação direta do algoritmo de predição linear através de APA apresenta dificuldades, devido às atualizações do tipo LMS-Frost. Por isso, procurou-se primeiramente achar uma estrutura adaptativa alternativa para esse algoritmo.



Da mesma forma que o algoritmo CMV, a função custo do algoritmo de equalização por predição linear vista no capítulo 7 também apresenta restrições. Como visto na seção 7.5, sua expressão é dada por:

$$\mathcal{F}(\mathbf{f}_j) = \|\mathbf{f}_j^H \mathbf{y}_1\|^2 \quad \text{sujeito a} \quad \mathbf{f}_j^H \mathbf{u} = 1 \quad (10.28)$$

Com isso, sua implementação pode ser feita seguindo uma estrutura GSC como a usada na implementação do algoritmo CMV e apresentada nas seções 6.4 e 6.5.

A restrição  $\mathbf{f}_j^H \mathbf{u} = 1$  é equivalente a um conjunto de restrições da forma  $\mathbf{u}\mathbf{u}^H \mathbf{f}_j = \mathbf{u}$ . Considerando-se  $\mathbf{U}_1 = \mathbf{u}\mathbf{u}^H$  e sendo  $\mathbf{U}_N$  o complemento ortogonal de  $\mathbf{U}_1$ , o detector  $\mathbf{f}_j$  pode ser parametrizado da seguinte forma:

$$\mathbf{f}_j = \mathbf{U}_1 \mathbf{s}_1 - \mathbf{U}_N \mathbf{s}_2 \quad (10.29)$$

Como  $\mathbf{U}_1^H \mathbf{U}_N = 0$  e  $\mathbf{U}_1^H \mathbf{f}_j = \mathbf{u}$ , uma outra parametrização possível para  $\mathbf{f}_j$  é:

$$\mathbf{f}_j = (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u} - \mathbf{U}_N \mathbf{s}_2 \quad (10.30)$$

A partir dessa expressão, nota-se que o detector multiusuário  $\mathbf{f}_j$  pode ser obtido a partir de uma estrutura GSC com atualizações de  $\mathbf{u}$  numa sub-banda e de  $\mathbf{s}_2$  na outra sub-banda. No entanto, o vetor  $\mathbf{u}$  pode ser obtido, iterativamente, a partir do erro de predição linear, como visto na seção 7.4. A fim de se obter as atualizações de  $\mathbf{s}_2$ , deve-se primeiro obter uma nova expressão para a função custo do sistema. Considerando então  $\mathbf{R}_y = \mathbf{y}_1 \mathbf{y}_1^H$ , substitui-se o valor de (10.30) em (10.28):

$$\begin{aligned} \mathcal{F}(\mathbf{f}_j) = \mathbf{f}_j^H \mathbf{R}_y \mathbf{f}_j = & \mathbf{u}^H (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{U}_1^H \mathbf{R}_y \mathbf{U} (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u} + \mathbf{s}_2^H \mathbf{U}_N^H \mathbf{R}_y \mathbf{U}_N \mathbf{s}_2 - \\ & \mathbf{u}^H (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{U}_1^H \mathbf{R}_y \mathbf{U}_N \mathbf{s}_2 - \mathbf{s}_2^H \mathbf{U}_N^H \mathbf{R}_y \mathbf{U} (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u} \end{aligned} \quad (10.31)$$

usou-se a propriedade do hermitiano de  $(\mathbf{U}_1^H \mathbf{U}_1)^{-1}$  ser a própria matriz, visto que o hermitiano do inverso de uma matriz é igual ao inverso de seu hermitiano [18].

Para uma atualização do tipo gradiente para  $\mathbf{s}_2$  tem-se a seguinte expressão:

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) - \mu_2 \nabla_{\mathbf{s}_2} \mathcal{F}(\mathbf{f}_j) \quad (10.32)$$

Derivando-se (10.31) em relação a  $\mathbf{s}_2$  obtém-se:

$$\nabla_{\mathbf{s}_2} \mathcal{F}(\mathbf{f}_j) = 2\mathbf{U}_N^H \mathbf{R}_y \mathbf{U}_N \mathbf{s}_2 - 2\mathbf{U}_N^H \mathbf{R}_y \mathbf{U}_1 (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u} \quad (10.33)$$

A atualização de  $\mathbf{s}_2$  passa a ser dada por:

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) - \mu_2 [\mathbf{U}_N^H \mathbf{R}_y \mathbf{U}_N \mathbf{s}_2(k) - \mathbf{U}_N^H \mathbf{R}_y \mathbf{U}_1 (\mathbf{U}_1^H \mathbf{U}_1) \mathbf{u}(k)] \quad (10.34)$$

Definem-se as seguintes variáveis auxiliares:

$$\tilde{\mathbf{y}}(k) = \mathbf{U}_1^H \mathbf{y}_1 \quad (10.35)$$

$$\bar{\mathbf{y}}(k) = \mathbf{U}_N^H \mathbf{y}_1 \quad (10.36)$$

$$e(k) = \tilde{\mathbf{y}}^H(k) (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u}(k) - \bar{\mathbf{y}}^H(k) \mathbf{s}_2(k) \quad (10.37)$$

Com isso, a equação (10.34) pode ser reescrita numa forma bem mais próxima de um LMS:

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) + \mu_2 e(k) \bar{\mathbf{y}}(k) \quad (10.38)$$

A expressão do detector multiusuário cego  $\mathbf{f}_j(k)$  é então dada por:

$$\mathbf{f}_j(k) = \mathbf{U}_1 (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \mathbf{u}(k) - \mathbf{U}_N \mathbf{s}_2(k) \quad (10.39)$$

mas, como  $\mathbf{U}_1 = \mathbf{u}\mathbf{u}^H$ , então:

$$\begin{aligned} \mathbf{U}_1 (\mathbf{U}_1^H \mathbf{U}_1)^{-1} &= \mathbf{u}\mathbf{u}^H (\mathbf{u}\mathbf{u}^H \mathbf{u}\mathbf{u}^H)^{-1} = \\ &= \mathbf{u}\mathbf{u}^H (\mathbf{u}\mathbf{u}^H)^{-1} (\mathbf{u}\mathbf{u}^H)^{-1} = \\ &= (\mathbf{u}\mathbf{u}^H)^{-1} = \mathbf{U}_1^{-1} \end{aligned} \quad (10.40)$$

Com isso, a equação (10.39) é simplificada para:

$$\mathbf{f}_j(k) = \mathbf{U}_1^{-1} \mathbf{u}(k) - \mathbf{U}_N \mathbf{s}_2(k) \quad (10.41)$$

Um esquema de funcionamento do algoritmo de predição linear usando GSC pode ser visto na figura 10.1

## 10.7 Algoritmo de Predição Linear com GSC Usando APA

Do mesmo modo que no caso do algoritmo CMV em sua versão GSC, foi implementado o algoritmo de predição linear com GSC usando atualizações APA,

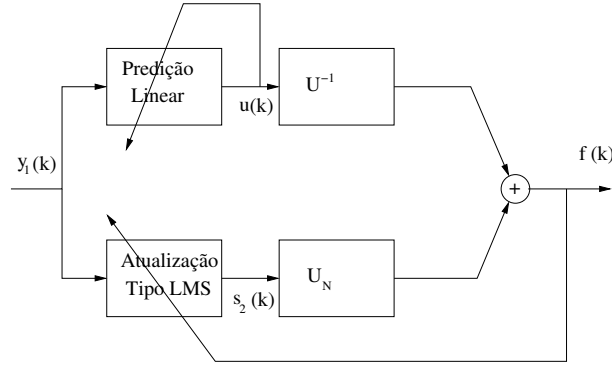


Figura 10.1: Esquema do algoritmo de predição linear usando GSC.

a fim de se aumentar a velocidade de convergência. No entanto, diferentemente do CMV, o APA só é usado para atualizar uma das sub-bandas (a que atualiza o parâmetro  $\mathbf{s}_2$ ), de acordo com a modelagem da equação (10.30), já que a sub-banda referente ao parâmetro  $\mathbf{u}$  já utiliza reuso de dados em sua atualização, devido à predição linear.

Considera-se que se faça reuso de  $L$  dados, ou seja, que cada dado recebido seja usado em  $L$  iterações consecutivas. Sendo:

$$\mathbf{Y}_1(k) = \begin{bmatrix} \mathbf{y}_1(k) \\ \mathbf{y}_1(k-1) \\ \vdots \\ \mathbf{y}_1(k-L+1) \end{bmatrix} \quad (10.42)$$

$$\tilde{\mathbf{Y}}(k) = \begin{bmatrix} \tilde{\mathbf{y}}(k) & \tilde{\mathbf{y}}(k-1) & \dots & \tilde{\mathbf{y}}(k-L+1) \end{bmatrix} \quad (10.43)$$

$$\bar{\mathbf{Y}}(k) = (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \begin{bmatrix} \bar{\mathbf{y}}(k) & \bar{\mathbf{y}}(k-1) & \dots & \bar{\mathbf{y}}(k-L+1) \end{bmatrix} \quad (10.44)$$

forma-se então o seguinte vetor de erros:

$$\mathbf{e}(k) = \tilde{\mathbf{Y}}^H(k) \mathbf{u}(k) - \bar{\mathbf{Y}}^H(k) \mathbf{s}_2(k) \quad (10.45)$$

Com isso, as atualizações da sub-banda de  $\mathbf{s}_2$  passam a ser feitas por:

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) + \mu_2 \mathbf{Y}_1(k) (\mathbf{Y}_1^H(k) \mathbf{Y}_1(k))^{-1} \mathbf{e}(k) \quad (10.46)$$

A expressão do detector  $\mathbf{f}_j(k)$  continua sendo dada por 10.41, ou seja,

$$\mathbf{f}_j(k) = \mathbf{U}_1^{-1} \mathbf{u}(k) - \mathbf{U}_N \mathbf{s}_2(k) \quad (10.47)$$

O algoritmo descrito nesta seção está representado em pseudo-código, ao final do capítulo, no quadro algoritmo 10.2.

## 10.8 Algoritmo de Predição Linear com Implementação GSC e Atualizações SMAP

Nesta seção incorpora-se o conceito de SM ao algoritmo de predição linear com GSC e atualizações APA mostrado na seção 10.7. De maneira semelhante ao feito para o algoritmo CMV, o objetivo da incorporação do SM ao algoritmo é diminuir o desajuste do sistema, diminuindo assim sua taxa de erros.

De maneira semelhante ao apresentado na seção 10.5, e obedecendo ao algoritmo SMAP, mostrado na seção 10.4, o critério de atualização do sistema obedece a:

- Compara-se o valor de  $e_0(k)$  (primeiro coeficiente do vetor de erro) com o parâmetro  $\gamma$ .

– Se  $e_0^2(k) > \gamma^2$ , então  $\alpha(k)$  é atualizado de acordo com

$$\alpha(k) = 1 - \frac{\gamma}{|e_0(k)|} \quad (10.48)$$

– Caso contrário:  $\alpha(k) = 0$ .

- Por fim, a sub-banda referente a  $\mathbf{s}_2$  é atualizada por:

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) + \mu_2 \bar{\mathbf{Y}}(k) (\bar{\mathbf{Y}}^H(k) \bar{\mathbf{Y}}(k))^{-1} \alpha(k) \mathbf{e}(k) \mathbf{t} \quad (10.49)$$

onde  $\mathbf{t} = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T$ .

## 10.9 Simulações dos Algoritmos Apresentados

### 10.9.1 Resultados para CMV com Atualizações APA

Foram feitas simulações com o algoritmo CMV com atualizações APA como descrito na 10.3. Foi utilizada a implementação assíncrona do sistema. Os valores de  $\mu_u$  e  $\mu_g$  foram reajustados para se maximizar a SER. Os parâmetros utilizados na simulação foram os seguintes:

- Foi utilizada a modelagem CMV da tese, conforme o discutido na seção 6.6.
- Foram transmitidos 25.000 símbolos.
- A SER foi medida depois de 20.000 símbolos transmitidos a fim de dar tempo para os algoritmos convergirem.
- Foi utilizada modulação BPSK.
- Foram feitas simulações para número de usuários de 1 a 5.
- Foram realizadas 5 execuções no *ensemble*.
- Foram utilizados os canais com respostas ao impulso dadas pelas equações (8.1) a (8.5).
- $\mu_u = 0,005$ .
- $\mu_g = 0,0001$ .

Na tabela 10.1 estão resumidos os resultados das simulações do algoritmo CMV com atualizações APA. Verifica-se que as SERs estão aproximadamente na mesma ordem de grandeza dos resultados encontrados com atualização LMS (ver tabela 8.17). No entanto, a característica mais importante do APA não está diretamente relacionada à diminuição da SER e sim à velocidade de convergência dos coeficientes do filtro.

Tabela 10.1: Resultados para o Algoritmo CMV de Xu e Tsatsanis - APA.

<b>Algoritmo CMV</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,361	0,359	0,372	0,372
<b>-15</b>	0,266	0,265	0,290	0,303
<b>-10</b>	0,145	0,142	0,195	0,212
<b>-5</b>	0,052	0,048	0,087	0,085
<b>0</b>	0,0035	0,0022	0,015	0,015
<b>5</b>	0	0	0,0010	0,0008
<b>10</b>	0	0	0,000002	0,000002
<b>15</b>	0	0	0	0

A partir das figuras 10.2 e 10.3 pode-se comparar a convergência de dois coeficientes dos filtros adaptativos de cada sub-banda para atualizações LMS e APA. Nota-se que as escalas horizontais são diferentes para o LMS e APA. Verifica-se claramente a maior velocidade de convergência do APA em ambos os casos. No entanto, houve também um aumento do desajuste, o que pôde ser verificado no caso da 1ª sub-banda. A aparente ausência do desajuste na 2ª sub-banda pode ser causada pela pequena escala em que este fenômeno se apresenta na 2ª sub-banda.

### 10.9.2 Resultados para CMV com Atualizações por SMAP

Foram realizadas simulações utilizando-se os mesmos parâmetros apresentados na seção 10.9.1. A única modificação é o fato de as simulações serem feitas apenas para 3 usuários no sistema. Os resultados encontrados estão apresentados nas tabelas 10.2 e 10.3 que se referem à SER e à taxa de atualização para diversos valores de  $\gamma$ .

Nas Figuras 10.6 e 10.7 estão apresentados a SER e a taxa de atualização

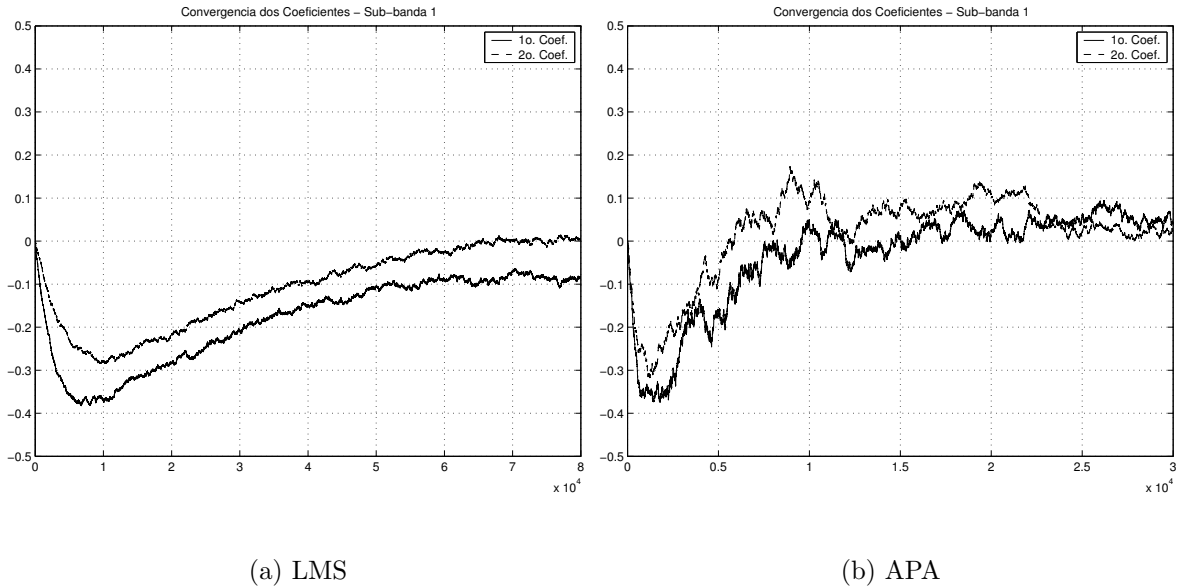


Figura 10.2: Comparação da convergência de coeficientes do filtro adaptativo na 1ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 5 dB.

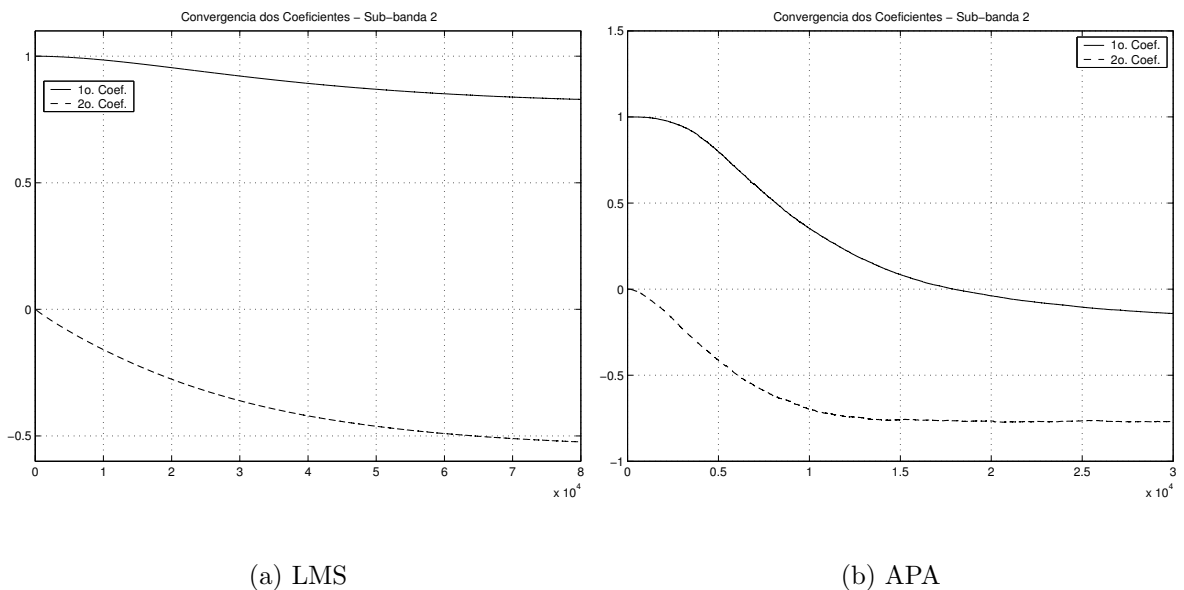


Figura 10.3: Comparação da convergência de coeficientes do filtro adaptativo na 2ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 5 dB.

para 0dB e 5dB, respectivamente. Percebe-se que, em ambos os casos, a escolha do parâmetro  $\gamma$  para 0,2 representa um bom compromisso entre o aumento da SER e a diminuição da taxa de atualização do sistema. Nesse caso, a SER para 5 dB aumentou de 0,0022 para 0,0035 enquanto a taxa de atualização foi de 88,4%. Verifica-se ainda que a degradação do sistema é menor que a percebida no caso da implementação SM sem o reuso de dados, causada por uma convergência mais rápida do sistema, devido ao APA.

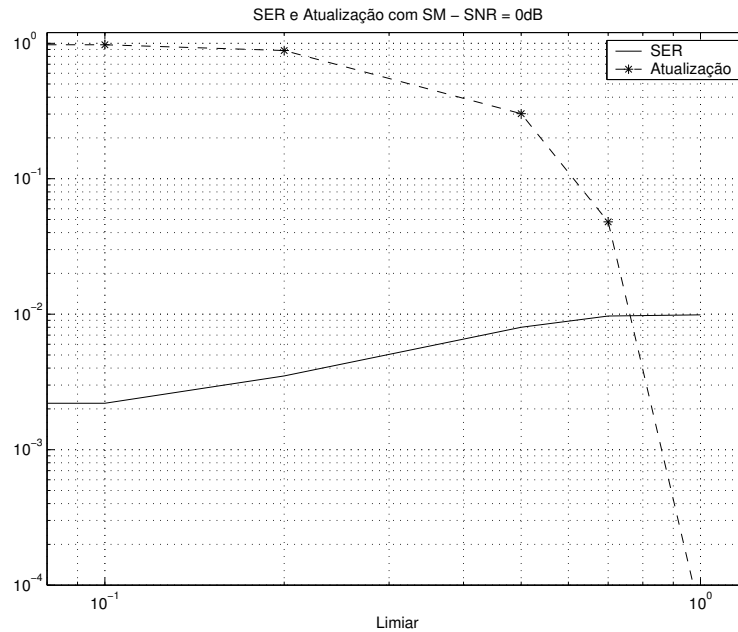


Figura 10.4: SER e taxa de atualização para SNR = 0dB para o algoritmo CMV.

Nas figuras 10.9.2 e 10.9.2 são mostradas comparações da convergência do algoritmo CMV usando APA apenas e usando SMAP para dois valores de  $\gamma$  para uma SNR de 5dB e 5 usuários no sistema. Observa-se que o uso da seletividade na atualização representada pelo SM diminui um dos principais problemas do APA: o elevado desajuste. Percebe-se ainda que para  $\gamma = 0,35$  (taxa de atualização de 53,9% para 5 usuários) o desajuste é ainda menor que no caso de  $\gamma = 0,2$  (taxa de atualização de 86,3% para 5 usuários). A convergência, no entanto, é um pouco mais lenta ao se usar um valor maior de  $\gamma$ .



Tabela 10.2: Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM.

Algoritmo CMV - SER						
	$\gamma$					
SNR(dB)	0	0,1	0,2	0,5	0,7	1
-10	0,142	0,141	0,139	0,139	0,144	0,156
-5	0,048	0,041	0,042	0,047	0,053	0,061
0	0,0022	0,0022	0,0035	0,0080	0,0097	0,0099
5	0	0	0	0,0002	0,0003	0,0004

Tabela 10.3: Resultados para o Algoritmo CMV de Xu e Tsatsanis - SM.

Algoritmo CMV - Taxa de Atualização (%)						
	$\gamma$					
SNR(dB)	0	0,1	0,2	0,5	0,7	1
-10	100	88,0	76,0	40,8	22,5	7,0
-5	100	91,5	80,6	34,5	11,8	1,1
0	100	97,5	88,4	30,2	4,8	0,006
5	100	99,8	95,6	29,2	2,5	0,002

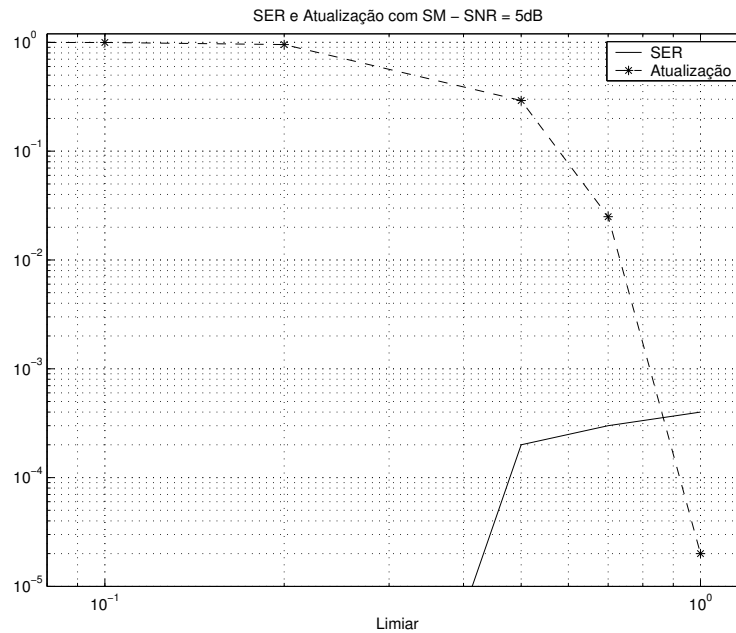
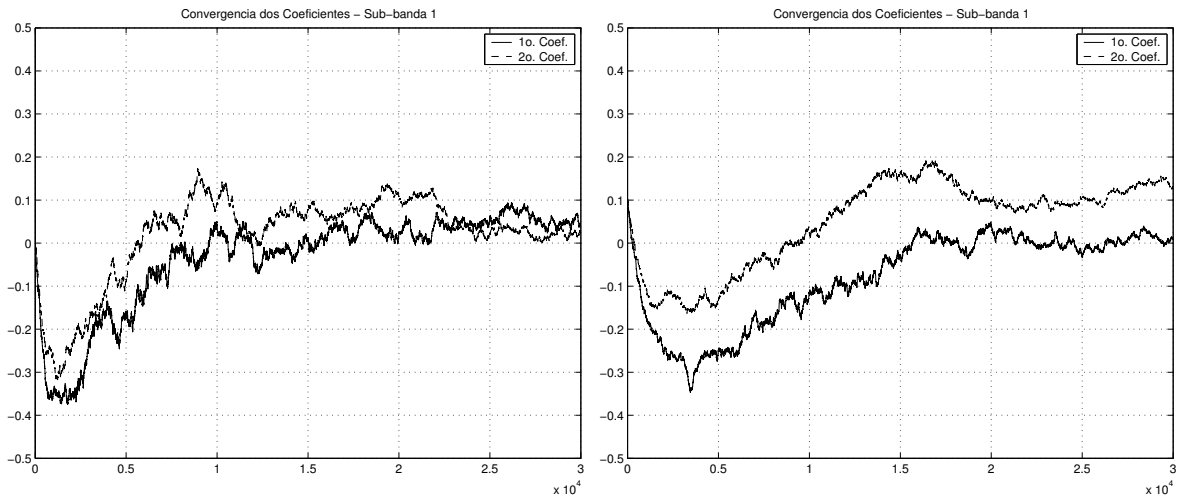


Figura 10.5: SER e taxa de atualização para  $\text{SNR} = 5\text{dB}$  para o algoritmo CMV.

### 10.9.3 Resultados para a Implementação GSC do Algoritmo de Predição Linear

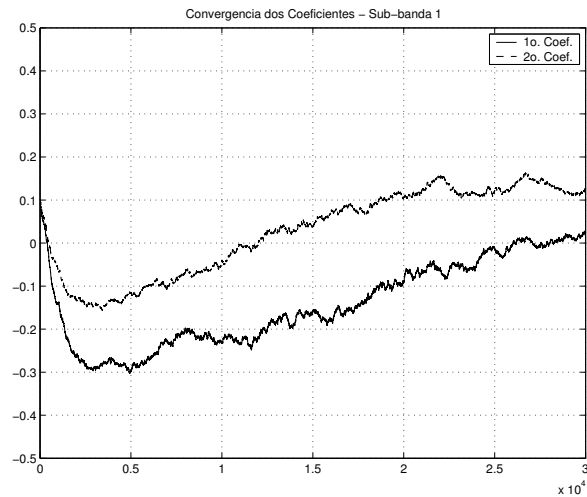
Foram feitas simulações para a estrutura GSC do algoritmo de predição linear, primeiramente sem reuso de dados, como descrito na seção 10.6, e depois com atualizações feitas por APA, como mostrado na seção 10.7. Em ambos os casos, o modelo de transmissão usada foi assíncrona, com os canais dados pelas equações (8.1) a (8.5). Foram transmitidos 20.000 símbolos, dando-se um intervalo de 15.000 símbolos para a convergência do algoritmo. A constelação usada na transmissão foi BPSK. O valor de  $\mu$  usado nos dois casos é de 0,001. No caso do APA foram usados quatro amostras no reuso.

Na tabela 10.4 estão apresentados os resultados para a implementação GSC do algoritmo de predição linear, enquanto na tabela 10.5 são mostrados os resultados para o algoritmo de predição linear com GSC e atualizações com reuso de dados através do APA. Nota-se que os resultados foram bastante parecidos entre si, não havendo um ganho expressivo no desempenho, ao contrário do algoritmo CMV. O ganho proporcionado pela convergência mais rápida do sistema é compensado



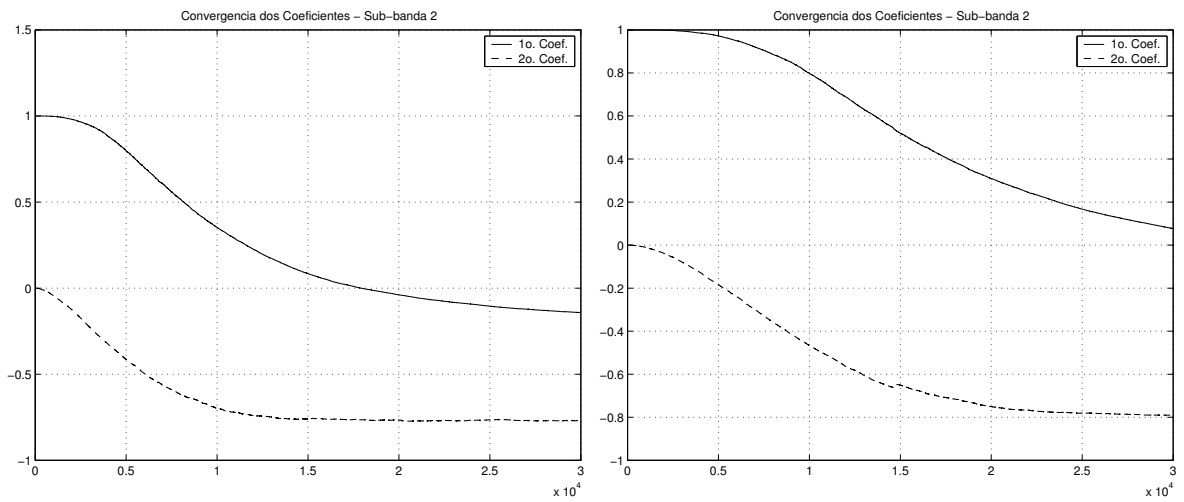
(a) APA

(b) SMAP -  $\gamma = 0,2$



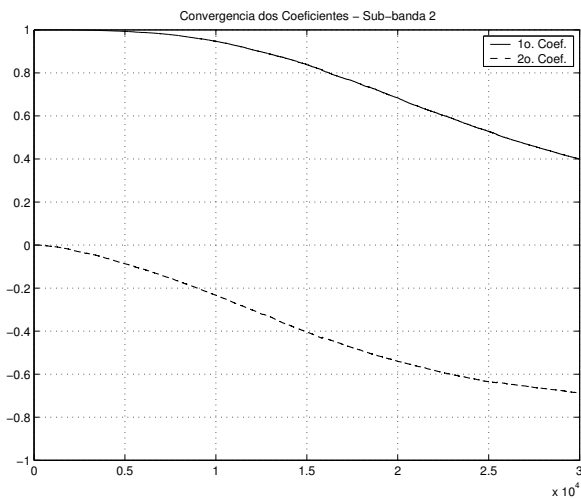
(c) SMAP -  $\gamma = 0,35$

Figura 10.6: Comparação da convergência de coeficientes do filtro adaptativo na 1ª sub-banda do algoritmo CMV: (a) com atualização APA; (b) com atualização SMAP para  $\gamma = 0,2$ ; (c) com atualização SMAP para  $\gamma = 0,35$ ; ambos para SNR = 5 dB.



(a) APA

(b) SMAP -  $\gamma = 0, 2$



(c) SMAP -  $\gamma = 0, 35$

Figura 10.7: Comparação da convergência de coeficientes do filtro adaptativo na 2ª sub-banda do algoritmo CMV: (a) com atualização LMS; (b) com atualização SMAP para  $\gamma = 0, 2$ ; (c) com atualização SMAP para  $\gamma = 0, 35$ ; ambos para SNR = 5 dB.

pelo seu maior desajuste. Na figura 10.9.3 pode-se ver uma comparação da convergência de dois coeficientes do filtro adaptativo na sub-banda relativa a  $\mathbf{s}_2$  para 0dB. Nota-se que as escalas dos gráficos são diferentes a fim de melhor mostrar a convergência dos dois algoritmos. Observa-se a convergência mais rápida do APA.

### 10.9.4 Resultados para o Algoritmo de Predição Linear com GSC-SMAP

Foi então implementado o algoritmo de predição linear com atualizações GSC-SMAP descrito na seção 10.8, simulando-o para 3 usuários, com passo de atualização  $\mu_2 = 0,001$ , 20.000 símbolos transmitidos e intervalo de 15.000 símbolos para a convergência do algoritmo. Os resultados encontrados são mostrados nas tabelas 10.6 e 10.7. A partir dos gráficos 10.9 e 10.10, pode-se ver as curvas de SER e taxa de atualização para SNR = -5 dB e 0 dB, respectivamente. O valor de  $\gamma = 0,6$  pode ser considerado um bom compromisso entre o aumento da SER e a diminuição da taxa de atualização. Para esse valor a SER em -5 dB sobe de 0,0017 para 0,0068 enquanto a taxa de atualização é de 47,0%.

Na figura 10.9.3 mostra-se a convergência de dois coeficientes do filtro adaptativo para  $\gamma = 0,45$  e SER = 0 dB para 5 usuários (taxa de atualização de 69,3%). Nota-se que o desajuste do sistema diminui em relação ao APA, mas a convergência ainda é mais rápida que a implementação GSC com atualização LMS pura nas sub-bandas, ou seja, sem APA nem SM.

## 10.10 Conclusões

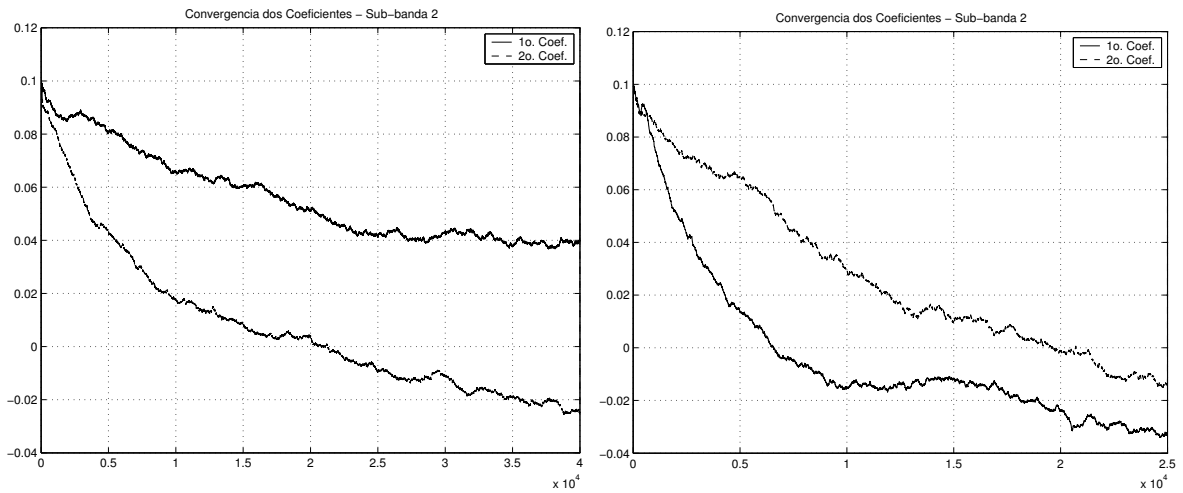
O objetivo deste capítulo é apresentar a incorporação de técnicas de reuso de dados nos algoritmo CMV e de predição linear para equalização cega. Isso constitui o segundo conjunto de inovações apresentado nesta tese. Essas modificações têm origem numa técnica de reuso de dados conhecida como APA. Além disso, é feita uma integração do algoritmo SM, visto no capítulo 9, com o APA a fim de diminuir

Tabela 10.4: Resultados para o Algoritmo de Predição Linear - GSC.

<b>Algoritmo de Pred. Linear - GSC</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,296	0,302	0,309	0,305
<b>-15</b>	0,165	0,174	0,194	0,203
<b>-10</b>	0,043	0,049	0,073	0,088
<b>-5</b>	0,00093	0,0014	0,0077	0,016
<b>0</b>	0	0	0,00006	0,0014
<b>5</b>	0	0	0	0,00004
<b>10</b>	0	0	0	0

Tabela 10.5: Resultados para o Algoritmo de Predição Linear - GSC - APA.

<b>Algoritmo de Predição Linear - GSC - APA</b>				
	<b>Usuários</b>			
<b>SNR (dB)</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
<b>-20</b>	0,294	0,299	0,303	0,305
<b>-15</b>	0,163	0,172	0,192	0,202
<b>-10</b>	0,040	0,047	0,072	0,087
<b>-5</b>	0,0011	0,0017	0,0077	0,016
<b>0</b>	0	0	0,0001	0,0009
<b>5</b>	0	0	0	0,00001
<b>10</b>	0	0	0	0



(a) LMS

(b) APA

Figura 10.8: Comparação da convergência de coeficientes do filtro adaptativo no algoritmo de Predição Linear, com implementação GSC e: (a) com atualização LMS; (b) com atualização APA; ambos para SNR = 0 dB.

Tabela 10.6: Resultados para o Algoritmo de Predição Linear - GSC - SMAP.

Algoritmo de Pred. Linear - GSC-SMAP - SER							
	$\gamma$						
SNR(dB)	0	0,1	0,2	0,5	0,6	0,7	1
-10	0,047	0,053	0,057	0,069	0,073	0,079	0,091
-5	0,0017	0,0025	0,0034	0,0051	0,0068	0,0092	0,019
0	0	0	0	0	0,000017	0,00008	0,0017
5	0	0	0	0	0	0	0,00005

Tabela 10.7: Resultados para o Algoritmo de Predição Linear - GSC - SMAP.

<b>Algoritmo de Pred. Linear - GSC-SMAP</b>							
<b>Taxa de Atualização (%)</b>							
	$\gamma$						
<b>SNR(dB)</b>	<b>0</b>	<b>0,1</b>	<b>0,2</b>	<b>0,5</b>	<b>0,6</b>	<b>0,7</b>	<b>1</b>
<b>-10</b>	100	93,1	85,7	58,1	48,3	39,2	17,2
<b>-5</b>	100	98,5	95,1	63,0	47,0	32,2	6,9
<b>0</b>	100	99,9	99,5	71,0	44,9	24,2	1,42
<b>5</b>	100	100	99,9	81,4	42,9	18,8	0,47

o problema do desajuste. Essa integração obedece o descrito no artigo [38], sob o nome de SMAP.

A seção 10.2 apresenta a base teórica do APA, enquanto a seção 10.3 mostra como as atualizações de coeficientes do algoritmo CMV podem ser modificadas de maneira a serem feitas através do APA. Já a seção 10.9.1 apresenta os principais resultados de simulações do algoritmo CMV com atualização APA. Dessas simulações percebe-se que o APA aumenta a velocidade de convergência do sistema mas aumenta o desajuste, o que por sua vez pode se refletir num aumento da SER do sistema. A seção 10.4 mostra como pode ser combinado o algoritmo SM com o APA, gerando o SMAP. Esse algoritmo resolve um dos principais problemas do APA, que é o seu elevado desajuste e ao mesmo tempo apresenta uma convergência mais rápida que o CMV tradicional, valendo-se do reuso de dados. As seções 10.5 e 10.9.2 apresentam a implementação do CMV utilizando-se atualizações SMAP.

A partir dos dados apresentados neste capítulo, conclui-se que o uso do algoritmo SMAP apresenta um compromisso entre uma velocidade de convergência maior e um menor desajuste. Além disso, o SMAP diminui o número de iterações em que os coeficientes das sub-bandas são atualizados em relação ao APA, o que diminui em média a complexidade do algoritmo.

Em geral, pode-se dizer que o APA consegue aumentar a velocidade de



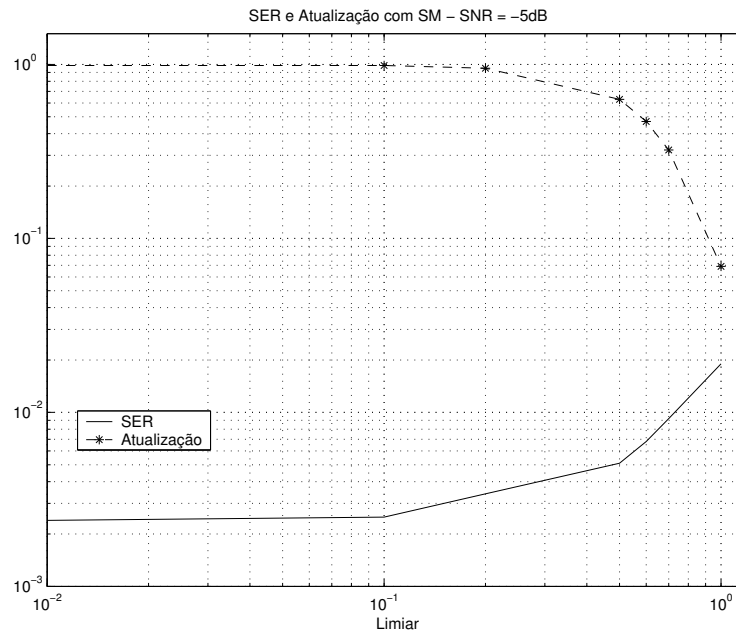


Figura 10.9: SER e taxa de atualização para  $\text{SNR} = -5\text{dB}$  para o algoritmo de predição linear - GSC-SMAP.

convergência do LMS ao preço do aumento do desajuste e da complexidade computacional, pois requer o processamento de uma quantidade maior de dados devido ao seu reuso. O aumento de complexidade, no entanto, não é tão grande como o de outra alternativa bastante popular para resolver a questão da baixa velocidade de convergência do LMS: o uso de RLS. Na verdade, o compromisso entre o aumento da velocidade de convergência e o aumento do desajuste aparece já no LMS tradicional, ao se ajustar o parâmetro  $\mu$ . No entanto, esse ajuste esbarra nas condições para convergência do algoritmo, podendo gerar instabilidade. A importância do APA está em se conseguir o aumento na velocidade de convergência ainda numa região estável para os coeficientes do filtro.

A inclusão do SM ao APA consegue atacar as duas principais desvantagens do APA em relação ao LMS tradicional: o aumento do desajuste e da complexidade computacional. Da mesma maneira que o algoritmo SM puro, o algoritmo SMAP proporciona, no limite  $\gamma = 0$ , as mesmas taxas de erro e a mesma complexidade que o APA. Conclui-se então que seu uso é válido pela liberdade do compromisso entre SER e taxa de atualização representada pelo parâmetro  $\gamma$ .

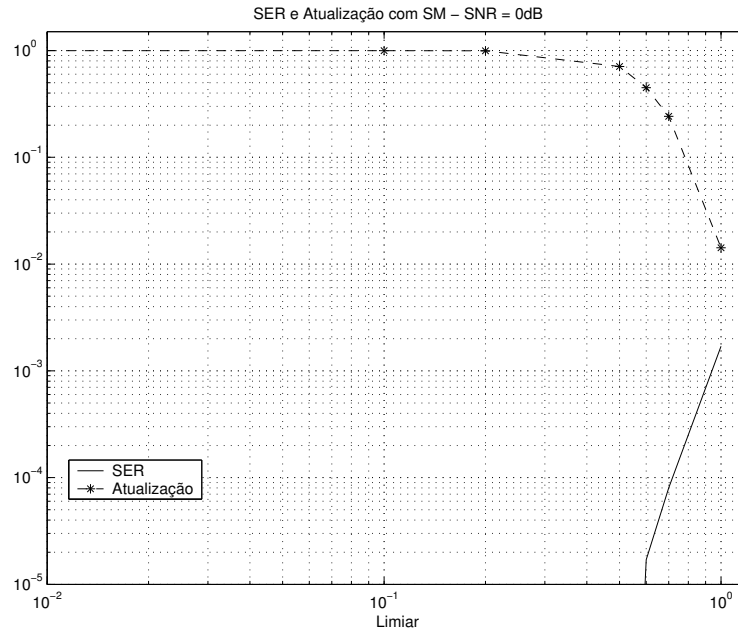


Figura 10.10: SER e taxa de atualização para  $\text{SNR} = 0\text{dB}$  para o algoritmo de predição linear - GSC-SMAP.

---

**Algoritmo 10.1** Algoritmo CMV com Atualizações por APA

---

Para  $j=1:\text{NúmeroDeUsuários}$ ,

$c_j = \text{GeraEspalhamento}(\text{'CódigoDeEspalhamento'}, j);$

$\text{Cmat}_j = \text{GeraMatrizEspalhamento}(c_j);$

$\text{CHC}_j = (\text{Cmat}_j^H \text{Cmat}_j)^{-1}$

$\text{Cn}_j = \text{AchaComplementoOrtogonal}(\text{Cmat}_j);$

$b_j = \text{GeraSinal}(\text{NúmeroDeSímbolos});$

$s_j = \text{Convolução}(c_j, h_j);$

$h_j = \text{GeraCanal}(\text{ComprimentoDoCanal});$

$r = r + \text{Convolução}(b_j, s_j);$

$\text{Potencia\_Sinal\_dB}(j) = \text{MedePotenciaSinalEmDB}(\text{Convolução}(b_j, s_j));$

$\text{Potencia\_Ruido\_dB}(j) = \text{SNR\_Escolhida\_dB} - \text{Potencia\_Sinal\_dB};$

$r = r + \text{GeraRuído}(\text{Potencia\_Ruido\_dB}(j));$

fim Para

Para  $j=1:\text{NúmeroDeUsuários}$ ,

Para  $k = 1:\text{NúmeroDeSímbolos}$ ,

---

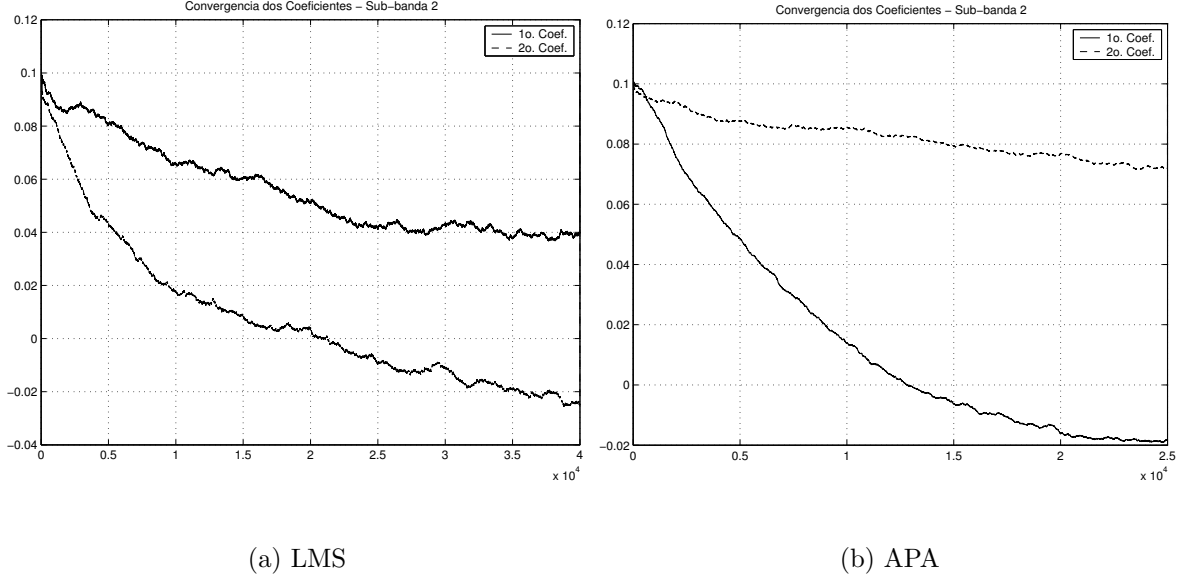


Figura 10.11: Comparação da convergência de coeficientes do filtro adaptativo no algoritmo de Predição Linear, com implementação GSC e: (a) com atualização LMS; (b) com atualização SMAP ( $\gamma = 0,45$ ); ambos para SNR = 0 dB.

---

```


$$\tilde{\mathbf{y}}(k) = \mathbf{Cmat}_j^H \mathbf{r};$$


$$\bar{\mathbf{y}}(k) = \mathbf{Cn}_j^H \mathbf{r};$$


$$\tilde{\mathbf{Y}}(k) = \begin{bmatrix} \tilde{\mathbf{y}}(k) & \tilde{\mathbf{y}}(k-1) & \dots & \tilde{\mathbf{y}}(k-L+1) \end{bmatrix}$$


$$\bar{\mathbf{Y}}(k) = (\mathbf{C}^H \mathbf{C})^{-1} \begin{bmatrix} \bar{\mathbf{y}}(k) & \bar{\mathbf{y}}(k-1) & \dots & \bar{\mathbf{y}}(k-L+1) \end{bmatrix}$$


$$\mathbf{e}(k) = \tilde{\mathbf{Y}}^H(k) \mathbf{g}_j(k) - \bar{\mathbf{Y}}^H(k) \mathbf{w}_{j,l}(k)$$


$$\mathbf{w}_{j,l}(k+1) = \mathbf{w}_{j,l}(k) + \mu_w \bar{\mathbf{Y}}(k) (\bar{\mathbf{Y}}^H(k) \bar{\mathbf{Y}}(k))^{-1} \mathbf{e}(k)$$


$$\mathbf{g}_j(k+1) = \mathbf{g}_j(k) + \mu_g \tilde{\mathbf{Y}}(k) (\tilde{\mathbf{Y}}^H(k) \tilde{\mathbf{Y}}(k))^{-1} \mathbf{e}(k)$$


$$\mathbf{g}_j(k+1) = \mathbf{g}_j(k+1) / \text{Norma2}(\mathbf{g}_j(k+1));$$

fim Para

fj = Cmat_j C H C_j g_j(k+1) - Cn_j w_{j,l}(k)

fim Para

Para j=1:NúmeroDeUsuários,
    Para k=1:NúmeroDeSímbolos,
        símbolo_estimado(j,k) = fj^H r;
    fim Para
fim Para

```

---

**Algoritmo 10.2** Algoritmo de Equalização Cega por Predição Linear, estrutura GSC, com Atualizações APA

---

```
Para j=1:NúmeroDeUsuários,  
    cj = GeraEspalhamento('CódigoDeEspalhamento',j);  
    Cmatj = GeraMatrizEspalhamento(cj);  
    bj = GeraSinal(NúmeroDeSímbolos);  
    sj = Convolução(cj,hj);  
     $\hat{S}_j$  = AchaComplementoOrtogonal(sj);  
    hj = GeraCanal(ComprimentoDoCanal);  
    r = r + Convolução(bj,sj);  
    Potencia_Sinal_dB(j) = MedePotenciaSinalEmDB(Convolução(bj,sj));  
    Potencia_Ruido_dB(j) = SNR_Escolhida_dB - Potencia_Sinal_dB;  
    r = r + GeraRuído(Potencia_Ruido_dB(j));  
fim Para
```

```
Para j=1:NúmeroDeUsuários,  
    Para k = 1:NúmeroDeSímbolos,  
        y1(k) = [r(k+1); r(k)];  
        y2(k) = [r(k+2);  $\hat{S}_j$ y1; r(k-1)];  
  
         $\mathbf{R}_1(k) = \frac{1}{k} [(k-1)\mathbf{R}_1(k-1)] + \mathbf{y}_1(k)\mathbf{y}_1^H(k);$   
         $\mathbf{R}_2(k) = \frac{1}{k} [(k-1)\mathbf{R}_2(k-1)] + \mathbf{y}_2(k)\mathbf{y}_2^H(k);$   
         $\mathbf{Z}(k) = \text{AchaPseudoInversa}(\mathbf{R}_2(k));$   
         $\mathbf{P} = \mathbf{R}_1(k)\mathbf{Z}(k);$   
         $\epsilon_1 = \mathbf{y}_1(k) - \mathbf{P}\mathbf{y}_2(k);$   
  
         $\mathbf{R}_\epsilon = \frac{1}{k} [(k-1)\mathbf{R}_\epsilon(k-1) + \epsilon_1(k)\epsilon_1^H(k);$   
  
        uj = AchaVetorSingularDoMaiorValorSingular( $\mathbf{R}_\epsilon$ );  
        U1 = ujujH;
```

---

$$\mathbf{U}_n = \text{AchaComplementoOrtogona}(\mathbf{U}_1);$$

$$\tilde{\mathbf{y}}(k) = \mathbf{U}_1^H \mathbf{y}_1;$$

$$\bar{\mathbf{y}}(k) = \mathbf{U}_n^H \mathbf{y}_1;$$

$$\tilde{\mathbf{Y}}(k) = \begin{bmatrix} \tilde{\mathbf{y}}(k) & \tilde{\mathbf{y}}(k-1) & \dots & \tilde{\mathbf{y}}(k-L+1) \end{bmatrix};$$

$$\bar{\mathbf{Y}}(k) = (\mathbf{U}_1^H \mathbf{U}_1)^{-1} \begin{bmatrix} \bar{\mathbf{y}}(k) & \bar{\mathbf{y}}(k-1) & \dots & \bar{\mathbf{y}}(k-L+1) \end{bmatrix};$$

$$\mathbf{e}(k) = \tilde{\mathbf{Y}}^H(j) \mathbf{u}_j(k) - \bar{\mathbf{Y}}^H(k) \mathbf{s}_2(k);$$

$$\mathbf{s}_2(k+1) = \mathbf{s}_2(k) + \mu_2 \bar{\mathbf{Y}}(k) (\bar{\mathbf{Y}}^H(k) \bar{\mathbf{Y}}(k))^{-1} \mathbf{e}(k);$$

fim Para

$$\mathbf{f}_j = \mathbf{U}_j^{-1} \mathbf{u}_j(k) - \mathbf{U}_j^H \mathbf{s}_2(k) ;$$

fim Para

---

# Capítulo 11

## Conclusões

### 11.1 Sobre a Equalização Cega

Os primeiros algoritmos de equalização cega, como vistos no capítulo 3, surgiram na década de 1970. De início, eram idealizados apenas para o caso monusuário e alguns tipos de transmissão em banda base. Pouco a pouco, esses algoritmos foram amadurecendo, utilizando ferramentas estatísticas cada vez mais variadas e se espalhando para o caso SIMO e multiusuário. Hoje em dia, a quantidade de trabalhos abordando equalização cega que surge na literatura beira o incontável.

A descrença que dominava os primeiros anos da equalização cega tinha como principais fatores a diferença de desempenho em relação aos robustos equalizadores treinados e o alto custo computacional gerado pelo uso de momentos estatísticos de 3ª e 4ª ordens. Com o passar do tempo, técnicas que utilizavam apenas estatísticas de segunda ordem foram desenvolvidas e o desempenho dos algoritmos cegos pouco a pouco melhorado. Hoje em dia, o originalmente impensável está acontecendo. Algoritmos de equalização treinada são sistematicamente questionados em relação ao uso mais eficiente das estatísticas do sinal, e métodos semi-cegos, que aplicam técnicas típicas de algoritmos cegos em ambientes treinados, começam a despontar como métodos com taxas de erro que se aproximam dos treinados para uma

mesma taxa de transmissão de informação, apesar de uma maior complexidade computacional.

De início, a entrada da equalização cega no cenário multiusuário tinha como foco a eliminação da ISI para o sistema em que ela se revelava especialmente nociva: o TDMA. Nesse caso, a ISI gera também interferência entre os usuários, visto que a transmissão é multiplexada no tempo. O CDMA, por seu lado, valia-se de seus ganhos de processamento para manter um status de imunidade em relação à ISI. Com o passar do tempo, houve um aumento nas taxas de transmissão e a concentração de pesquisa na melhoria do funcionamento da MUD, o que levou a eliminação de ISI de novo à cena no CDMA. O uso de MUD cega, por sua vez, exigiu que a estimativa do canal de cada usuário fosse feita também sem treinamento. Com isso, a equalização cega estava estabelecida como uma das áreas de pesquisa promissoras em transmissões CDMA. Com a padronização de três sistemas de terceira geração que utilizam compartilhamento de acesso via CDMA (WCDMA, cdma2000 e TDMA-CDMA), o cenário se tornou ideal para que mais investimentos e mais pesquisas fossem realizadas na área.

Por todas essas razões, o uso de equalizadores cegos e ainda mais em ambientes CDMA, que era uma receita inicialmente improvável, por unir a descrença na equalização cega e sua aparente inutilidade para CDMA, passou a ser uma das áreas de pesquisa de ponta em processamento de sinais.

A absorção de equalizadores cegos por sistemas comerciais de telefonia móvel, parece ainda muito distante. Na verdade, parece tão distante que várias técnicas de processamento de sinais já são cogitadas para a 5ª geração de móveis, como a transmissão por banda ultra-larga (UWB - *Ultra Wide-Band*) enquanto as previsões para equalização cega não cogitam nem levantar em qual geração a equalização cega se encaixaria. No entanto, pelo recente histórico de surpresas da área, e pelo seu rápido amadurecimento da técnica (os primeiros artigos de equalização cega para CDMA encontrados na literatura têm em torno de 10 anos), não há como se ter certezas quanto à necessidade de haver seqüências de treinamento trafegando sem fio nos céus de um futuro próximo.

## 11.2 Sobre os Algoritmos Atuais

De uma maneira geral, pode-se dizer que os algoritmos mostrados nos capítulos 5 a 7 se valem da robustez proporcionada pelo CDMA para o seu funcionamento, ou seja, a própria resistência inerente ao CDMA quanto à presença de ISI e de separação de usuários já é responsável em parte pelo bom funcionamento dos algoritmos. No entanto, somente as qualidades do espalhamento espectral não são suficientes para garantir as taxas de erro alcançadas, visto que há uma discrepância no desempenho comparado dos algoritmos. O amadurecimento de técnicas de supressão de ruído, por exemplo, também tem uma influência decisiva no funcionamento dos algoritmos.

A incorporação de estruturas adaptativas oriundas de outras áreas do conhecimento, como o caso do GSC, cuja origem está na exploração de diversidade espacial na recepção de sinais, também foi fundamental para a melhoria no desempenho dos algoritmos. A velocidade de convergência dos algoritmos também foi outro fator bastante ajudado pela importação de técnicas adaptativas já existentes.

Pode-se dizer que, apesar de pertencer a famílias de algoritmos diferentes (subespaço, variância mínima e predição), os algoritmos utilizados ainda apresentam muitos pontos estruturais em comum. Na verdade, ainda guardam semelhanças com alguns algoritmos que influenciaram bastante o desenvolvimento da área como o descrito no artigo [22] de Honig e Verdú. Essas semelhanças já não existem tanto ao se pesquisar equalização cega para o caso monousuário, o que aponta um certo espaço livre que ainda existe para ser explorado no caso multiusuário.

A ainda visível separação nas funcionalidades de supressão de ruído e estimação da assinatura (que engloba separação de usuários e eliminação de ISI) também pode mostrar outro espaço livre para avanços, já que a integração dessas funções, explorando as características comuns dos tipos de interferência presentes em cada um deles, potencialmente se traduz em menor complexidade computacional e eliminação mais eficiente das interferências.



O uso do conceito de filtragem SM proporciona uma diminuição na carga computacional do sistema. De acordo com o visto no Capítulo 9, é possível, através de um ajuste do parâmetro  $\gamma$  do sistema, evitar que essa diminuição da carga computacional leve a um aumento excessivo na SER do sistema.

Um resumo do desempenho dos algoritmos de acordo com as simulações feitas pode ser visto na tabela 11.1.

Tabela 11.1: Resumo da comparação entre os algoritmos.

<b>Usuários</b>	<b>Síncrono</b>	<b>Assíncrono</b>	<b>Sem Memória</b>
<b>1 ou 2</b>	Igual	CMV	-
<b>5</b>	Pred. Linear (ambos)	Pred. Linear (Kalman)	Igual

Nota-se que o algoritmo de predição linear revelou-se o melhor no caso de simulações para 5 usuários, tanto para o caso síncrono quanto o assíncrono. Isso indica que esse método é o mais resistente para separação multiusuário no caso em que há ISI, que gera uma perda de ortogonalidade das formas de onda dos usuários. O desempenho similar dos algoritmos para o canal sem memória mostra que a função de eliminação de ruído tem desempenho semelhante nos algoritmos utilizados nessa simulação.

## **11.3 Sobre Atualização Seletiva e Reuso de Dados**

De uma maneira geral, pode-se dizer que o algoritmo SM utilizado para a atualização seletiva mostrou-se capaz de diminuir a complexidade computacional do sistema, oferecendo uma liberdade para a escolha do parâmetro  $\gamma$  que oferecesse o melhor compromisso entre aumento da complexidade computacional e aumento da SER. O pequeno *overhead* de processamento causado pela presença do SM é mais do que compensado pela economia de complexidade representada pelo algoritmo. O uso do SM também causa uma diminuição no desajuste do sistema. A atualização seletiva, portanto, oferece duas vantagens pelo preço de um *overhead* computacional

bem pequeno. Além disso, o caso extremo de escolha do parâmetro  $\gamma = 0$ , que representa nenhuma economia de processamento recai no algoritmo original. Com isso, pode-se dizer que, tem-se, no pior caso, o compromisso do algoritmo original em relação à complexidade computacional e SER.

O reuso de dados, representado nesta tese pelo APA, têm como principal vantagem o aumento da velocidade de convergência do algoritmo, ao custo de um aumento no desajuste e da complexidade computacional do algoritmo. O algoritmo LMS puro também fornece a opção de aumento da velocidade de convergência em troca de aumento de desajuste, mas isso pode levar à instabilidade. Ao se usar o APA, o aumento da velocidade de convergência se dá numa faixa em que o algoritmo adaptativo ainda converge. Seu uso é válido em aplicações onde o desajuste e o aumento da complexidade não são tão críticos quanto a velocidade de convergência. Nessa descrição nem sempre se encaixam os algoritmos de equalização cega.

A fim de se minimizar o problema do desajuste, mas mantendo as vantagens da velocidade de convergência do APA, foi utilizado um algoritmo que combina atualização seletiva e reuso de dados: SMAP, descrito no artigo [38]. A presença do SM nesse algoritmo resolve as questões de desajuste e complexidade computacional do APA, oferecendo as características do SM original num cenário de reuso de dados. O uso combinado de SM com APA oferece um aumento na velocidade de convergência com um efeito pequeno sobre o desajuste e um efeito atenuado sobre a complexidade computacional. Com isso, o uso do SMAP é válido para algoritmos que possam suportar um pequeno aumento da carga computacional para se aumentar a velocidade de convergência sem aumento grande no desajuste. Esse pequeno aumento na complexidade é relativo e dependente de quão sensível é o sistema com relação à taxa de atualização para o aumento da SER. No entanto, como visto pelos resultados dos capítulos 9 e 10, em relação ao algoritmo de predição linear, a sensibilidade da SER do algoritmo em relação à taxa de atualização diminuiu ao se colocar o reuso de dados. Isso é intuitivamente compreensível, dado que o reuso de dados gera uma redundância natural nas atualizações do sistema.

## 11.4 Contribuições da Tese

As primeiras contribuições da tese podem ser vistas nos capítulos iniciais, nos quais se coloca a questão da equalização, desde o problema da ISI até o surgimento da equalização cega. Com isso, apresenta-se como a eliminação da ISI era feita na época dos primeiros equalizadores cegos, mostrando muitas das influências que de certa forma moldaram o desenvolvimento inicial da teoria de equalização cega.

O capítulo 3 tem como principal contribuição apresentar o desenvolvimento histórico dos equalizadores cegos, mostrando a evolução das soluções empregadas para se driblar o duplo problema de desconhecimento tanto da seqüência transmitida quanto do canal de transmissão. De certo modo, nesse capítulo está contida boa parte da explicação da opção de se limitar o conteúdo desta tese para o uso de momentos até segunda ordem e ambientes multiusuário, já que essas escolhas refletem o que há de mais novo no enorme escopo englobado pela teoria de equalização cega.

Os capítulos 5 a 7 têm a importância de colocar três algoritmos representativos das mais importantes famílias de equalizadores cegos dentro de uma mesma visão comparativa, uniformizando a notação e estabelecendo parâmetros comparativos comuns, como a divisão da questão da equalização cega em três objetivos básicos: supressão de ruído, separação entre usuários e eliminação de ISI. Em particular, as seções 5.5 e 6.6 procuram questionar algumas partes dos artigos [17] e [21] no que se refere à modelagem do CDMA, sugerindo modificações que procuram retratar melhor o funcionamento real do compartilhamento de acesso pelo CDMA.

O capítulo 8, por sua vez, procura estabelecer relações comparativas mais complexas entre os algoritmos a partir de resultados de simulação. Sua função é, em parte, a de fechar o raciocínio dos capítulos 5, 6 e 7. A análise comparativa nele contida apresenta um painel geral da situação atual da questão da equalização cega para multiusuário.

O capítulo 9 apresenta uma inovação: uma implementação alternativa dos

algoritmos baseados em LMS usando a técnica de filtragem SM. Essa implementação visa diminuir a complexidade computacional do sistema, atualizando seletivamente os coeficientes do filtro adaptativo.

No capítulo 10 são mostradas ainda duas outras inovações: a implementação com reuso de dados (no caso, usando o algoritmo APA) dos algoritmos CMV e de predição linear e a implementação dos mesmos algoritmos usando uma combinação do SM com APA (ou seja, o algoritmo SMAP descrito em [38]). Algoritmos de reuso de dados aumentam a velocidade de convergência do filtro adaptativo, mas, por outro lado, geram um aumento no desajuste do sistema. O algoritmo SMAP resolve esse problema, garantindo uma convergência rápida do algoritmo adaptativo com um pequeno desajuste. Como consequência, tem-se um detector multiusuário cego com convergência bem mais rápida que o original, mantendo o pequeno desajuste. Outra inovação presente nesse capítulo é a implementação LMS usando uma estrutura GSC para o algoritmo de predição linear.

## 11.5 Propostas de Trabalhos Futuros

A ampliação do conteúdo da análise comparativa é sempre bem vinda, por gerar maior volume e diversidade de dados necessários para conclusões mais gerais e para direcionar melhor o desenvolvimento de métodos que se aproveitem de partes de métodos já existentes. Nesse sentido propõe-se que seja ampliado o estudo comparativo a fim de conter os seguintes tópicos:

- Implementar um algoritmo SM com reuso de dados, como o *Set-Membership Binormalized* [37] ou o *Set-Membership Affine Projection* [38] para a atualização dos coeficientes dos equalizadores cegos baseados no método de subespaço.
- Métodos de suavização por mínimos quadrados, que são considerados, no caso monousuário, como parentes mais antigos dos métodos de predição linear, mas que foram revitalizados para o caso multiusuário. Um exemplo desses

métodos é o artigo [39] de Zhao e Tong.

- Comparações com métodos que utilizam estatísticas de ordem superior. Apesar de esses métodos serem criticados pelo custo computacional e por apresentar mínimos locais, métodos como o CMA (*constant modulus algorithm*) sobreviveram ao tempo, foram otimizados e adaptados ao caso multiusuário. Um artigo com um método baseado em CMA para multiusuário foi publicado por Tugnait [40].
- Comparação com técnicas de multiplexoras que também combatem a ISI e estão bem cotadas para, em combinação com o CDMA, dominar os padrões de 4<sup>a</sup> geração de telefonia móvel. Um bom tutorial para sistemas multiplexoras pode ser encontrado em [41].
- Estudo de técnicas que melhoram o desempenho de algoritmos cegos através de um grau de conhecimento maior sobre o sinal transmitido, como no caso de equalização semicega e cega em grupo. Um tutorial que abrange os principais aspectos da equalização cega em grupo é encontrado no livro [42] de Wang e Poor.

# Referências Bibliográficas

- [1] PROAKIS, J., *Digital Communications*. 4 ed. New York, McGraw-Hill, 2001.
- [2] CHANG, R. W., HANCOCK, J. C., “On Receiver Structures for Channels Having Memory”, *IEEE Transactions on Information Theory*, v. 12, n. 4, pp. 463–468, Outubro 1966.
- [3] PAPOULIS, A., *Probability, Random Variables and Stochastic Processes*. 3 ed. New York, McGraw-Hill, 1991.
- [4] KAY, S., *Fundamentals of Statistical Signal Processing*. Englewood Cliffs, Prentice Hall, 1993.
- [5] FORNEY, G. D., “Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference”, *IEEE Transactions on Information Theory*, v. 18, n. 3, pp. 363–377, Maio 1972.
- [6] RAPPAPORT, T., *Wireless Communications: Principles and Practice*. 2 ed. Upper Saddle River, Prentice Hall, 2002.
- [7] QURESHI, S., “Adaptive Equalization”, *Proceedings of the IEEE*, v. 73, n. 9, pp. 1349–1387, Setembro 1985.
- [8] DINIZ, P. S. R., *Adaptive Filtering: Algorithms and Practical Implementation*. 2 ed. Kluwer Academic Publishers, 2002.
- [9] TREICHLER, J., LARRIMORE, M., HARP, J., “Practical Blind Demodulators for High-Order QAM Signals”, *Proceedings of IEEE*, v. 86, n. 10, pp. 1907–1926, Outubro 1998.

- [10] DING, Z., LI, Y., *Blind Equalization and Identification*. New York, Marcel Dekker, 2001.
- [11] PICKHOLTZ, R., MILSTEIN, L., SCHILLING, D., “Spread Spectrum for Mobile Communications”, *IEEE Transactions on Vehicular Technology*, v. 40, n. 2, pp. 313–322, Maio 1991.
- [12] PICKHOLTZ, R., SCHILLING, D., MILSTEIN, L., “Theory of Spread-Spectrum Communications - A Tutorial”, *IEEE Transactions on Communications*, v. COM-30, n. 5, pp. 855–884, Maio 1982.
- [13] KORHONEN, J., *Introduction to 3G Mobile Communications*. 2 ed. Norwood, Artech House, 2003.
- [14] DINIZ, P. S. R., “Principles of Wireless Communications”, Abril 2003. Apostila do Curso COE798 - Comunicações Móveis do PEE/COPPE.
- [15] COVER, T., THOMAS, J., *Elements of Information Theory*. New York, John Wiley and Sons, 1991.
- [16] MOSHAVI, S., “Multi-User Detection for DS-CDMA Communications”, *IEEE Communications Magazine*, v. 34, n. 10, pp. 124–136, Outubro 1996.
- [17] WANG, X., POOR, V., “Blind Multiuser Detection: A Subspace Approach”, *IEEE Transactions on Information Theory*, v. 44, n. 2, pp. 677–690, Março 1998.
- [18] STRANG, G., *Linear Algebra and Its Applications*. 3 ed. San Diego, Harcourt Brace and Company, 1988.
- [19] LUENBERGER, D. G., *Linear and Nonlinear Programming*. 2 ed. Reading, Addison-Wesley, 1989.
- [20] YANG, B., “Projection Approximation Subspace Tracking”, *IEEE Transactions on Signal Processing*, v. 44, n. 1, pp. 95–107, Janeiro 1995.

- [21] XU, Z., TSATSANIS, M., “Blind Adaptive Algorithms for Minimum Variance CDMA Receivers”, *IEEE Transactions on Communications*, v. 49, n. 1, pp. 180–193, Janeiro 2001.
- [22] HONIG, M., MADHOW, U., VERDU, S., “Blind Adaptive Multiuser Detection”, *IEEE Transactions on Information Theory*, v. 41, n. 4, pp. 944–960, Julho 1995.
- [23] XU, Z., TSATSANIS, M., “Performance Analysis of Minimum Variance CDMA Receivers”, *IEEE Transactions on Signal Processing*, v. 5, n. 11, pp. 3014–3022, Novembro 1998.
- [24] GRIFFITHS, L., JIM, C., “An Alternative Approach to Linearly Constrained Adaptive Beamforming”, *IEEE Transactions on Antennas and Propagation*, v. AP-30, n. 1, pp. 27–34, Janeiro 1982.
- [25] GESBERT, D., DUHAMEL, P., “Unbiased Blind Adaptive Channel Identification and Equalization”, *IEEE Transactions on Signal Processing*, v. 48, n. 1, pp. 148–158, Janeiro 2000.
- [26] ZHANG, X.-D., WEI, W., “Blind Adaptive Multiuser Detection Based on Kalman Filtering”, *IEEE Transactions on Signal Processing*, v. 50, n. 1, pp. 87–95, Janeiro 2002.
- [27] LI, X., FAN, H., “Direct Blind Multiuser Detection for CDMA in Multipath without Channel Estimation”, *IEEE Transactions on Signal Processing*, v. 49, pp. 63–73, Janeiro 2001.
- [28] LIU, X., TEH, K. C., GUNAWAN, E., “Blind Adaptive Kalman Filter-based Multiuser Detector over a Multipath Fading Channel”, *IEEE Communications Letters*, v. 8, n. 6, pp. 342–345, Junho 2004.
- [29] KAPOOR, S., GOLLAMUDI, S., NAGARAJ, S., *et al.*, “Adaptive Multiuser Detection and Beamforming for Interference Suppression in CDMA Mobile Radio Systems”, *IEEE Transactions on Vehicular Technology*, v. 48, n. 5, pp. 1341–1355, Setembro 1999.



- [30] FAN, H., LI, X., “An Adaptive Linear Prediction Algorithm for Joint Blind Equalization and Blind Multiuser Detection in CDMA”, *Conference Record of the 33rd. Asilomar Conference on Signals, Systems and Computers*, , n. 1, pp. 24–27, Outubro 1999.
- [31] BENSLEY, S., AAZHANG, B., “Subspace-Based Channel Estimation for Code Division Multiple Access Communication Systems”, *IEEE Transactions on Communications*, v. 44, n. 8, pp. 1009–1020, Agosto 1996.
- [32] LIU, H., XU, G., “A Subspace Method for Signature Waveform Estimation in Synchronous CDMA Systems”, *IEEE Transactions on Communications*, v. 44, n. 10, pp. 1346–1353, Outubro 1996.
- [33] TSATSANIS, M., “Inverse Filtering Criteria for CDMA Systems”, *IEEE Transactions on Signal Processing*, v. 45, n. 1, pp. 102–112, Janeiro 1997.
- [34] ABED-MERAIM, K., MOULINES, E., LOUBATON, P., “Prediction Error Method for Second-Order Blind Identification”, *IEEE Transactions on Signal Processing*, v. 45, n. 3, pp. 694–705, Março 1997.
- [35] DASGUPTA, S., HUANG, Y. F., “Asymptotically Convergent Modified Recursive Least Squares with Data Dependent Updating and Forgetting Factor for Systems with Bounded Noise”, *IEEE Transactions on Information Theory*, v. IT-33, n. 3, pp. 383–392, Maio 1987.
- [36] GOLLAMUDI, S., NAGARAJ, S., KAPOOR, S., *et al.*, “Set-Membership Filtering and Set-Membership Normalized LMS Algorithm with Adaptive Step Size”, *IEEE Signal Processing Letters*, v. 5, n. 5, pp. 111–114, Maio 1998.
- [37] DINIZ, P. S. R., WERNER, S., “Set-Membership Binormalized Data-Reusing LMS Algorithms”, *IEEE Transactions on Signal Processing*, v. 51, n. 1, pp. 124–134, Janeiro 2003.
- [38] WERNER, S., DINIZ, P. S. R., “Set-Membership Affine Projection Algorithm”, *IEEE Signal Processing Letters*, v. 8, n. 8, pp. 231–235, Agosto 2001.

- [39] ZHAO, Q., TONG, L., “Adaptive Blind Channel Estimation by Least Squares Smoothing for CDMA”, *Proceedings of SPIE Conference on Advanced Signal Processing Algorithms*, v. 3461, pp. 55–66, Julho 1998.
- [40] TUGNAIT, J. K., “Blind Spatio-Temporal Equalization and Impulse Response Estimation for MIMO Channels Using a Godard Cost Function”, *IEEE Transactions on Signal Processing*, v. 45, n. 1, pp. 268–271, Janeiro 1997.
- [41] BALTAR, L. G., “Aplicação de Banco de Filtros em Sistemas de Comunicação Multiportadora”, Projeto Final de Eng. Eletrônica - UFRJ, Abril 2004.
- [42] WANG, X., POOR, V., *Wireless Communication Systems: Advanced Techniques for Signal Reception*. Upper Saddle River, Prentice Hall, 2003.