

ANÁLISES COMPARATIVAS DE ALGORITMOS ADAPTATIVOS PARA  
EQUALIZAÇÃO CEGA

André Felipe Costa Vliese

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. Sergio Lima Netto, Ph.D.

---

Prof. Paulo Sergio Ramirez Diniz, Ph.D.

---

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

---

Prof. Gláucio Lima Siqueira, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

Março de 2001

VLIESE, ANDRÉ FELIPE COSTA  
ANÁLISES COMPARATIVAS DE ALGORITMOS  
ADAPTATIVOS PARA EQUALIZAÇÃO CEGA [Rio  
de Janeiro] 2001  
IX,108 p. 29,7 cm (COPPE/UFRJ,  
M.Sc., Engenharia Elétrica, 2001)  
Tese - Universidade Federal do Rio de  
Janeiro, COPPE  
1. Identificação de sistemas  
2. Equalização cega de sistemas  
3. Análise de Sistemas de comunicações  
I. COPPE/UFRJ II. Título (série)

*Aos meus pais, Osmar e Vera, às  
minhas irmãs, Érica e Taísa, às min-  
has avós Etelvina e Maria e à minha  
querida namorada Kellen.*

## Agradecimentos

Meus sinceros agradecimentos:

- Aos professores Sergio Lima Netto e Paulo Sergio Ramirez Diniz, pela orientação durante todo o período de pesquisa;
- Aos funcionários, professores e colegas do Laboratório de Processamento de Sinais da COPPE/UFRJ, especialmente Fábio Freeland, Ranniery da Silva Maia e Rogério Caetano pelas importantes contribuições;
- À CAPES, pela bolsa de estudos fornecida.

André Felipe Costa Vliese

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

ANÁLISE COMPARATIVA DE ALGORITMOS ADAPTATIVOS DE  
EQUALIZAÇÃO CEGA

André Felipe Costa Vliese

Março/2001

Orientadores: Sergio Lima Netto  
Paulo Sergio Ramirez Diniz

Programa: Engenharia Elétrica

Neste trabalho, são apresentadas comparações entre diferentes algoritmos de identificação e equalização cega propostos recentemente. Serão comparados o algoritmo Bussgang, algoritmos baseados na decomposição de subespaços, o algoritmo de predição de erro e também o algoritmo LMS. A análise da eficiência é efetuada pelas comparações da taxa de erro do sinal equalizado no receptor e pela complexidade computacional dos algoritmos. As avaliações realizadas nesta tese indicam quais os métodos são mais apropriados na elaboração de um receptor com equalizador não supervisionado ou com treinamento curto.

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

COMPARATIVE ANALYSIS OF BLIND EQUALIZATION ADAPTIVE  
ALGORITHMS

André Felipe Costa Vliese

March/2001

Advisors: Sergio Lima Netto  
Paulo Sergio Ramirez Diniz

Department: Electrical Engineering

In this work, we present a detailed comparison among several adaptive algorithms for blind equalization. The algorithms considered in the analysis include the Bussgang algorithm, subspace-based algorithms, prediction error-based algorithms, and also the LMS algorithm. The performance of each algorithm is assessed by the bit error rate of the equalized signal, at the receiver end, as well as the computational complexity. The results indicate which methods are more appropriate to design receivers including blind and semi-blind equalizers.

# Sumário

<b>1</b>	<b>Equalização de Sistemas</b>	<b>1</b>
1.1	Introdução . . . . .	1
1.2	Objetivo da Tese . . . . .	3
<b>2</b>	<b>Equalização Cega</b>	<b>5</b>
2.1	Introdução . . . . .	5
2.2	Algoritmos de Equalização Cega . . . . .	5
2.3	Algoritmo Bussgang . . . . .	7
2.3.1	Convergência do Algoritmo Bussgang . . . . .	10
2.4	Algoritmos Baseados em Poliespectros . . . . .	12
2.5	Algoritmos Baseados em Estatísticas Cicloestacionárias . . . . .	14
2.5.1	Equalização Cega Fracionalmente Espaçada por Decom- posição de Subespaços Vetoriais . . . . .	15
2.5.2	Algoritmo Baseado na Decomposição de Subespaços . . . . .	18
2.6	Resultados Experimentais . . . . .	21
2.7	Conclusão . . . . .	22
<b>3</b>	<b>Equalização Cega por Decomposição em Subespaços com Suavização</b>	<b>23</b>
3.1	Introdução . . . . .	23
3.2	Características Gerais Requeridas para Equalização de Canal . . . . .	24
3.3	Modelo do Sistema . . . . .	24

3.4	Algoritmo de Detecção Conjunta de Ordem e Estimação Cega de Canal por Suavização Mínima Quadrática . . . . .	27
3.4.1	Definições e Relações entre os Subespaços . . . . .	27
3.4.2	Suavização Mínima Quadrática . . . . .	28
3.5	Implementação do Algoritmo J-LSS . . . . .	30
3.5.1	Formulação Geral do Método J-LSS . . . . .	30
3.5.2	Estruturas de Dados e Identificação de Canal . . . . .	31
3.5.3	Resumo do Algoritmo J-LSS . . . . .	34
3.6	Resultados Experimentais . . . . .	35
3.7	Conclusão . . . . .	42
<b>4</b>	<b>Equalização Cega Adaptativa por Decomposição em Subespaços com Suavização</b> . . . . .	<b>44</b>
4.1	Introdução . . . . .	44
4.2	Características para Operação em Tempo Real . . . . .	45
4.3	Modelo do Sistema . . . . .	46
4.4	Estimação de Canal via Suavização Mínima Quadrática Adaptativa . . . . .	46
4.4.1	Estrutura de Dados e Detecção de Ordem . . . . .	46
4.4.2	Suavização Mínima Quadrática Adaptativa . . . . .	49
4.5	Resultados Experimentais . . . . .	52
4.6	Conclusão . . . . .	54
<b>5</b>	<b>Algoritmo de Equalização Cega de Predição de Erro</b> . . . . .	<b>56</b>
5.1	Introdução . . . . .	56
5.2	Modelo do Sistema . . . . .	57
5.3	Método de Predição de Erro para a Equalização Cega . . . . .	57
5.3.1	Método de Predição de Erro com Ruído Aditivo . . . . .	58
5.4	Predição Linear Adaptativa Não Polarizada . . . . .	60
5.4.1	Predição Restrita . . . . .	60
5.4.2	Remoção de Ruído AWGN . . . . .	61



---

5.5	Algoritmo de Predição de Erro . . . . .	62
5.5.1	Implementação do Algoritmo . . . . .	62
5.6	Resultados Experimentais . . . . .	63
5.7	Conclusão . . . . .	69
<b>6</b>	<b>Detecção Multiusuário Cega por Decomposição de Subspaços</b>	<b>70</b>
6.1	Introdução . . . . .	70
6.2	Modelo do Sistema . . . . .	71
6.3	Detectores Multiusuário Cegos Baseados em Subespaços . . . . .	73
6.3.1	Determinação de Subespaços . . . . .	73
6.3.2	Detector Linear de Descorrelação . . . . .	75
6.3.3	Detector Linear MMSE . . . . .	75
6.3.4	Detecção Usando Estimativas da Matriz de Autocorrelação	77
6.4	Estimação de Canal Adaptativa Cega para Sinais CDMA . . . . .	78
6.5	Resultados Experimentais . . . . .	81
6.6	Conclusão . . . . .	87
<b>7</b>	<b>Comparação no Desempenho dos Algoritmos</b>	<b>88</b>
7.1	Introdução . . . . .	88
7.2	Resultados Experimentais . . . . .	89
7.3	Conclusão . . . . .	95
<b>8</b>	<b>Equalização Semicega</b>	<b>98</b>
8.1	Introdução . . . . .	98
8.2	Modelo de Sistema . . . . .	99
8.3	Resultados Experimentais . . . . .	99
8.4	Conclusão . . . . .	102
<b>9</b>	<b>Conclusão e Trabalhos Futuros</b>	<b>103</b>
9.1	Objetivos Alcançados por esta Tese . . . . .	103
9.2	Trabalhos Futuros . . . . .	105

# Capítulo 1

## Equalização de Sistemas

### 1.1 Introdução

Um problema bastante comum em sistemas de comunicações consiste na degradação do sinal transmitido após a passagem por um canal de transmissão não ideal. Canal ideal é aquele onde o sinal que chega ao receptor é um múltiplo atrasado do sinal original, ou seja, se fosse transmitido o sinal  $s(t)$ , no receptor deveria aparecer  $\alpha s(t - \tau)$ , onde  $\alpha$  é um ganho complexo e  $\tau$  um atraso de propagação.

Porém, devido às imperfeições de um canal, elementos como ruído aditivo e propagação por multipercursos provocam interferências e acabam por degradar a qualidade do sinal recebido, elevando a taxa de erro de bit em sistemas digitais, por exemplo.

Para resolver o problema, torna-se necessário o uso de filtros especiais, denominados equalizadores, cuja função é amenizar as distorções produzidas pelo canal, regenerando o sinal original.

De modo geral, pode-se produzir equalização através de dois métodos: identificação de canal usando topologia paralela, Figura 1.1.a, e modelagem inversa usando topologia em cascata, Figura 1.1.b. No método de identificação, o

processo de equalização se dá em dois estágios. No primeiro, o modelo é colocado em paralelo com o canal e a identificação do mesmo é realizada. Em um segundo estágio, o inverso do modelo é colocado em série com o canal, em modo direto de recepção, perfazendo a equalização do mesmo, como indicado na porção limitada pela linha pontilhada na Figura 1.1.b. Neste caso, a resposta do equalizador é o inverso da resposta do modelo. Já na modelagem inversa, o modelo é colocado diretamente em série com o canal, e a resposta impulsional do equalizador ao ser convoluída com a resposta ao impulso do canal de transmissão gera idealmente um impulso. Neste caso, cuja a resposta do equalizador é igual à do modelo, o método também é denominado equalização direta.

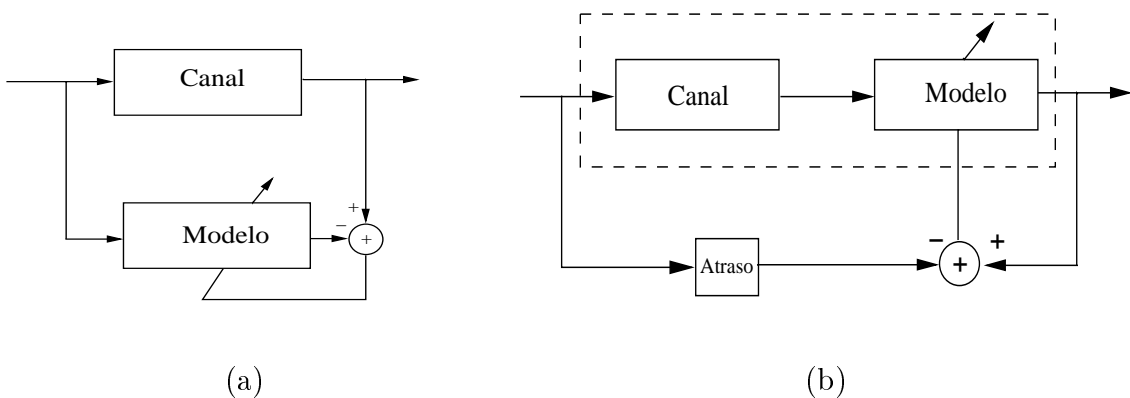


Figura 1.1: Topologias de equalização paralela e cascata.

Vale a pena ressaltar que, em geral, a implementação dos equalizadores nos dois métodos é do tipo FIR. Isto se dá porque a implementação dos mesmos com resposta IIR é bastante complexa e dependendo do canal a ser equalizado, como por exemplo canais de fase não mínima onde existem zeros fora do círculo de raio unitário no plano  $z$ , o resultado não é prático, já que nesse caso, o equalizador seria instável [1, 2].

Tipicamente, os equalizadores adaptativos usados em sistemas de comunicações requerem um período inicial de treinamento, durante o qual uma seqüência conhecida de dados é transmitida. Após o período de treinamento, o equalizador

é comutado para o modo direção de decisão e a partir daí a transmissão pode ser realizada. Desse modo, diz-se que o equalizador trabalha de maneira supervisionada. Contudo, há várias situações onde o período de treinamento é indesejável como nos exemplos abaixo [1]:

- Sistemas ponto-multiponto, onde uma unidade central é conectada a várias unidades independentes. Neste caso, cada unidade independente tem seu período dedicado de comunicação com a unidade central. Assim, a unidade central deve treinar cada unidade independente antes da transmissão dos dados o que eleva sensivelmente a vazão da rede, diminuindo bastante o desempenho da mesma.
- Sistemas de comunicações móveis: Neste caso, o custo envolvido na repetição da transmissão das seqüências de treinamento e também a inevitável presença de desvanescimento por multipercurso tornam todo o processo de treinamento caro.

Atualmente, estão sendo desenvolvidos métodos que permitem realizar a equalização sem o envio de seqüências de treinamento. A esses métodos dá-se a denominação genérica de equalização cega. Nestes métodos não é transmitida informação em excesso para o propósito de equalização. Todo o processo é realizado com próprio sinal a ser transmitido.

## 1.2 Objetivo da Tese

O objetivo desta tese é estudar os principais algoritmos de equalização cega e comparar seus desempenhos junto aos métodos de equalização que usam seqüências de treinamento. A estruturação da tese é a seguinte: No capítulo 2, será dada uma introdução aos principais métodos de equalização cega em relação à estatística do sinal de entrada. No capítulo 3, é apresentado o método de detecção conjunta de ordem e estimação de canal baseada na decomposição do subespaço

---

do sinal recebido. No capítulo 4, é apresentada uma versão adaptativa para o mesmo método do capítulo anterior. No capítulo 5, é apresentado o método de predição de erro para equalização direta de um canal de transmissão. No capítulo 6, um método de equalização para sinais com espalhamento espectral é avaliado. No capítulo 7, são apresentados resultados comparativos entre os métodos cegos apresentados. No capítulo 8, um modelo de equalização semicega, composto por um período curto de treinamento e seguido por equalização cega, é proposto. Neste capítulo, simulações envolvendo os casos cego e semi-cego também são avaliados. Por último, no Capítulo 9 são apresentadas as conclusões gerais da tese e sugestões para futuros trabalhos.

# Capítulo 2

## Equalização Cega

### 2.1 Introdução

Neste capítulo, é dada em linhas gerais uma introdução à equalização cega e seus principais métodos. Na seção 2.2, os métodos cegos são descritos a partir das estatísticas do sinal recebido. Na seção 2.3, será apresentado o algoritmo Bussgang para a equalização de canais com resposta ao impulso real, baseados na transmissão de sinais multiníveis modulados em amplitude [3]. Na seção 2.4, os métodos de equalização cega baseados em poliespectros são apresentados. Na seção 2.5, é apresentado o método de decomposição de subespaços vetoriais, baseado na cicloestacionaridade do sinal recebido. Na seção 2.6, o algoritmo de Bussgang é avaliado em uma simulação de equalização de canal. Por último, na seção 2.7, são apresentadas as conclusões do capítulo.

### 2.2 Algoritmos de Equalização Cega

Considere um sistema linear e invariante no domínio do tempo discreto, com entrada  $s(k)$ , saída  $y(k)$  e resposta ao impulso  $h(k)$  como visto na Figura 2.1. As amostras da seqüência de entrada são consideradas do tipo independente e identicamente distribuídas (iid). Na prática a única característica que se sabe

sobre a seqüência de entrada é sua função distribuição de probabilidade.

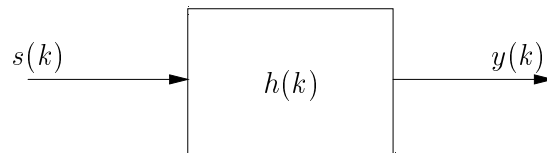


Figura 2.1: Sistema discreto no tempo genérico.

O problema considerado aqui é o de se regenerar  $s(k)$  ou, simplesmente, identificar o inverso do sistema  $h(k)$ , dada a seqüência observada  $y(k)$  na saída do sistema. Se o sistema  $h(k)$  for de fase mínima, então tanto o sistema quanto o seu inverso serão estáveis. Com isso, o problema da equalização cega é mais facilmente resolvido. Em várias situações, no entanto, os sistemas analisados são de fase não-mínima, como por exemplo linhas de transmissão de telefone e canais de rádio móveis [1]. Neste caso, a restauração da seqüência de entrada  $s(k)$ , dada a seqüência de saída do canal, não é facilmente obtida. A relação entrada-saída, no domínio da freqüência, é dada por

$$S_y(f) = |H(f)|^2 S_s(f) \quad (2.1)$$

onde  $S_y(f)$ ,  $S_s(f)$  e  $|H(f)|$  são respectivamente a densidade espectral de potência da saída, da entrada e o módulo da resposta em freqüência do sistema.

Podemos identificar duas grandes famílias de algoritmos de equalização cega, dependendo da forma da informação adicional usada pelo algoritmo para compensar o não conhecimento da seqüência de entrada do sistema:

1. Algoritmos baseados em estatísticas de ordem superior: Esta família ainda pode ser subdividida em dois grupos [1]:
  - Algoritmos baseados implicitamente em estatísticas de ordem superior, que exploram as estatísticas de alta ordem do sinal recebido em um

senso implícito. Neste grupo estão incluídos algoritmos da família Bussgang.

- Algoritmos baseados explicitamente em estatísticas de ordem superior, que usam cumulantes de alta ordem ou suas transformadas discretas de Fourier, também denominadas de poliespectro, que preservam a informação da fase do sinal transmitido.
2. Algoritmos que exploram estatísticas cicloestacionárias de segunda ordem do sinal recebido. A propriedade da cicloestacionaridade é importante por surgir em um sinal modulado que resulta em variação em amplitude, fase ou frequência da portadora senoidal, o qual é o princípio básico em sistemas de telecomunicações [1].

## 2.3 Algoritmo Bussgang

Considere um modelo de equalização em cascata como na Figura 2.2.

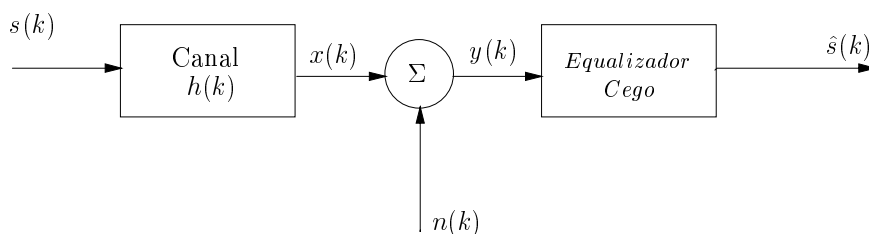


Figura 2.2: Equalização em cascata de um canal de comunicações e um equalizador cego.

O canal, que inclui os efeitos do filtro de transmissão, do meio de comunicação e também do filtro receptor, é caracterizado pela sua resposta ao impulso  $h(k)$  que a princípio é desconhecida podendo ser variante no tempo inclusive. Assim a relação entrada-saída do canal, onde assumimos  $h(k) = 0$  para  $k < 0$ , ou



seja, o canal cáusal, é dada por

$$y(k) = \sum_{m=0}^{\infty} s(k-m)h(m) + n(k) \quad (2.2)$$

Desejamos projetar um equalizador cego cuja resposta impulsional é o inverso da resposta do canal com seqüência de entrada desconhecida e sem um sinal de referência. Para resolver o problema, é necessário descrever o modelo probabilístico da seqüência de entrada  $s(k)$ . Para tanto, assume-se que  $s(k)$  tenha função densidade de probabilidade simétrica e uniforme e seja uma seqüência do tipo branca, isto é, os símbolos são variáveis aleatórias tipo iid, com média zero e variância unitária, ou seja

$$\begin{cases} E[s(k)] = 0 \\ E[s(k)s(i)] = \delta(i-k) \end{cases} \quad (2.3)$$

Seja  $w$  a resposta ao impulso do equalizador que é relacionado com a resposta  $h(k)$  do canal por

$$\sum_i w(i)h(k-i) = \delta(k) \quad (2.4)$$

O equalizador é definido como ideal, no sentido que ele reconstrói a seqüência  $s(k)$  corretamente. Para demonstrar isso, primeiro escrevemos com a ajuda da equação (2.2) na ausência do ruído,

$$\sum_i w(i)y(k-i) = \sum_i \sum_m w(i)h(m)s(k-i-m) \quad (2.5)$$

Fazendo a troca dos índices,  $m = l - i$ , no lado direito desta equação e alterando a ordem do somatório temos

$$\sum_i w(i)y(k-i) = \sum_l s(k-l) \sum_i w(i)h(l-i) \quad (2.6)$$

Usando a equação (2.4) na equação acima, chega-se a

$$\sum_i w(i)y(k-i) = \sum_l s(k-l)\delta(l) = s(k) \quad (2.7)$$

como resultado desejado.

Na prática, a resposta impulsional do canal,  $h(k)$ , é desconhecida. Assim, não podemos então usar a equação (2.4) para determinar os coeficientes do equalizador. Ao invés disso, pode-se usar um procedimento iterativo para calcular a resposta impulsional inversa aproximada,  $\hat{w}(i, k)$ , para o equalizador onde o índice  $i$  se refere ao  $i$ -ésimo coeficiente do filtro transversal na  $k$ -ésima iteração, como indicado na Figura 2.3. O cálculo é realizado até que a convolução da resposta  $\hat{w}(k)$  com o sinal recebido  $y(k)$  resulta em uma completa ou parcial remoção da interferência entre símbolos. Desse modo, na  $k$ -ésima iteração, tem-se a resposta aproximada da seqüência de saída do equalizador dada por

$$r(k) = \sum_{i=-L}^L \hat{w}(i, k)y(k-i) \quad (2.8)$$

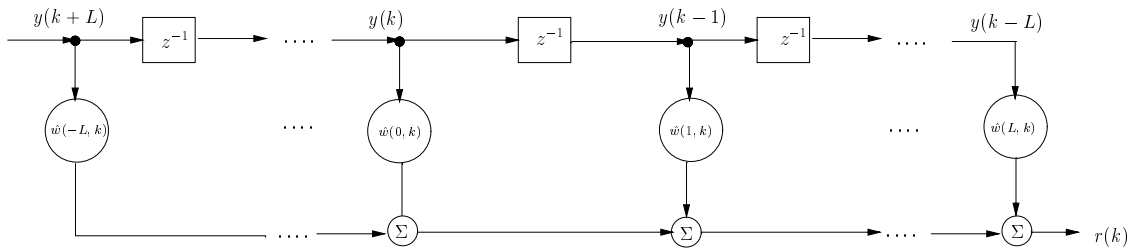


Figura 2.3: Realização transversal para o equalizador aproximado.

A saída do equalizador é então aplicada a um estimador não-linear sem memória, produzindo uma estimativa  $\hat{s}(k)$  para o símbolo de entrada  $s(k)$ , como visto na Figura 2.4. Logo podemos escrever

$$\hat{s}(k) = g(r(k)) \quad (2.9)$$

onde  $g(\bullet)$  é uma função não-linear. Um exemplo de função aplicada ao estimador em questão é

$$g(k) = \text{sgn}(k) = \begin{cases} +1, & k \geq 0 \\ -1, & k < 0 \end{cases}$$

Para se estimar os coeficientes  $\hat{w}(i, k)$ , têm-se uma variedade de algoritmos adaptativos que podem ser encontrados na literatura [1, 2]. Um exemplo é o algoritmo LMS que apresenta um baixo custo computacional. No algoritmo LMS, interpretando a estimativa não-linear  $\hat{s}(k)$  como a resposta desejada e reconhecendo  $r(k)$  como a resposta do filtro adaptativo, pode-se expressar o erro de estimação como

$$e(k) = \hat{s}(k) - r(k) \quad (2.10)$$

O valor atualizado para o  $i$ -ésimo coeficiente na iteração  $(k + 1)$  é então calculado por

$$\hat{w}(i, k + 1) = \hat{w}(i, k) + \mu y(k - i)e(k) \quad (2.11)$$

onde  $\mu$  é o valor do passo do algoritmo.

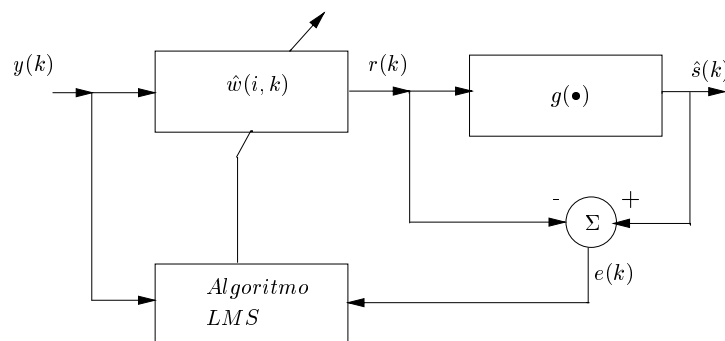


Figura 2.4: Diagrama em blocos de um equalizador cego.

### 2.3.1 Convergência do Algoritmo Bussgang

Para o algoritmo iterativo descrito na seção anterior convergir em valor médio, com respeito a solução não polarizada da equação (2.4), espera-se que o valor médio dos coeficientes  $\hat{w}(i, k)$  tenda a um valor constante conforme o

número de iterações  $k$  tenda para o infinito. Matematicamente, temos para  $k$  suficientemente grande

$$E[y(k-i)r(k)] = E[y(k-i)g(r(k))] \quad (2.12)$$

A partir da equação (2.8), notamos que

$$r(k-m) = \sum_{i=-L}^L \hat{w}(i, k-m)y(k-m-i) = \sum_{i=-L-k}^{L-k} \hat{w}(i-m, k)y(k-i) \quad (2.13)$$

Multiplicando ambos os lados da equação (2.12) por  $\hat{w}(i-m, k)$  e somando para  $i$ , chega-se a

$$E[r(k) \sum_{i=-L}^L \hat{w}(i-m, k)y(k-i)] = E[g(r(k)) \sum_{i=-L}^L \hat{w}(i-m, k)y(k-i)] \quad (2.14)$$

Considerando que  $L$  seja suficientemente grande para permitir uma equalização perfeita, pode-se aproximar a expressão de  $r(k-m)$  por

$$r(k-m) = \sum_{i=-L}^L \hat{w}(i-m, k)y(k-i) \quad (2.15)$$

Podemos usar a equação (2.15) para simplificar a equação (2.14)

$$E[r(k)r(k-m)] \approx E[g(r(k))r(k-m)] \quad (2.16)$$

Então um processo estocástico é dito ser um processo do tipo Bussgang, se satisfaz a seguinte condição

$$E[r(k)r(k-m)] = E[r(k)g(r(k-m))] \quad (2.17)$$

Na verdade um processo Bussgang tem como propriedade sua função autocorrelação ser igual a correlação cruzada entre o processo e a saída da não-linearidade produzida pelo processo e ambas com o mesmo atraso [1].

## 2.4 Algoritmos Baseados em Poliespectros

Existe uma classe de algoritmos de equalização cega que usa estatísticas de alta ordem do sinal recebido numa forma explícita. Para tal, é necessário um conhecimento sobre cumulantes.

Considere um processo estocástico  $y(k)$  com média zero

$$E[y(k)] = 0 \quad (2.18)$$

Sejam as variáveis aleatórias  $y(k), y(k + \tau_1), \dots, y(k + \tau_{m-1})$  obtidas pela observação desse processo nos instantes  $k, k + \tau_1, \dots, k + \tau_{m-1}$ , respectivamente. Forma-se então o vetor

$$\mathbf{y} = [y(k), y(k + \tau_1), \dots, y(k + \tau_{m-1})]^T \quad (2.19)$$

Considere agora a função característica da variável aleatória  $y$  definida por

$$\Theta(s) = E[e^{sy}] \quad (2.20)$$

a função característica segunda da mesma variável aleatória é definida por

$$\Psi(s) = \ln (\Theta(s)) \quad (2.21)$$

O cumulante  $c_n$  de  $n$ -ésima ordem é dado por

$$c_n = \frac{d^n \Psi(0)}{ds^n} \quad (2.22)$$

como exemplo, os cumulantes de segunda, terceira e quarta ordem são dados respectivamente por

$$c_2(\tau) = E[y(k)y(k + \tau)] \quad (2.23)$$

$$c_3(\tau_1, \tau_2) = E[y(k)y(k + \tau_1)y(k + \tau_2)] \quad (2.24)$$

$$\begin{aligned} c_4(\tau_1, \tau_2, \tau_3) &= E[y(k)y(k + \tau_1)y(k + \tau_2)y(k + \tau_3)] \\ &\quad - E[y(k)y(k + \tau_1)]E[y(k + \tau_2)y(k + \tau_3)] \\ &\quad - E[y(k)y(k + \tau_2)]E[y(k + \tau_1)y(k + \tau_3)] \\ &\quad - E[y(k)y(k + \tau_3)]E[y(k + \tau_1)y(k + \tau_2)] \end{aligned} \quad (2.25)$$

Considere agora o sistema linear invariante no tempo da Figura 2.1, excitado pela entrada  $s(k)$  consistindo de variáveis aleatórias iid. O cumulante de  $m$ -ésima ordem de  $y(k)$  é dado por

$$c_m^y(\tau_1, \tau_2, \dots, \tau_{m-1}) = c_m^s \sum_{i=-\infty}^{\infty} h(i)h(i + \tau_1) \dots h(i + \tau_{m-1}) \quad (2.26)$$

onde  $c_m^s$  é o cumulante de  $m$ -ésima ordem do processo de entrada  $s(k)$ . O poliespectro de  $m$ -ésima ordem é definido pela transformada discreta de Fourier do respectivo cumulante:

$$C_m(\omega_1, \dots, \omega_{m-1}) = \sum_{\tau_1=-\infty}^{\infty} \dots \sum_{\tau_{m-1}=-\infty}^{\infty} c_m^y(\tau_1, \dots, \tau_{m-1}) e^{-j(\omega_1 \tau_1 + \dots + \omega_{m-1} \tau_{m-1})} \quad (2.27)$$

Logo, tomando a transformada discreta de Fourier de  $m$ -ésima ordem em ambos os lados da equação (2.26) teremos

$$C_m^y(\omega_1, \dots, \omega_{m-1}) = c_m^s H(e^{j\omega_1}) \dots H(e^{j\omega_{m-1}}) H(e^{j\omega_1 + \dots + j\omega_{m-1}}) \quad (2.28)$$

Uma propriedade interessante dos poliespectros é que cada poliespectro de ordem maior do que 2 se anula quando o processo  $y(k)$  é gaussiano. Desse modo, os espectros de ordem maior provém uma medida da diferença de um processo estocástico para a característica da *gaussianidade*.

Enquanto a densidade espectral de potência de um processo estacionário é invariante à fase, os poliespectros de ordens mais altas são sensíveis à fase. Mais especificamente falando, a densidade espectral de potência é uma função real. Nós claramente vemos que ao passar um processo estocástico estacionário através de um sistema linear, a informação sobre a resposta da fase do sistema é eliminada se consideramos apenas o espectro de potência na saída. Em contraste, os poliespectros são funções complexas, o que significa que na situação similar acima, os poliespectros do sinal de saída do sistema preserva a informação de fase. Logo, como a equação (2.26) não suprime a informação da fase no sistema, é possível identificar canais de fase não mínima se as estatísticas de alta ordem

da entrada e da saída são conhecidas. Para maiores detalhes sobre equalização cega baseada em poliespectros ver [1].

## 2.5 Algoritmos Baseados em Estatísticas Cicloestacionárias

Uma alternativa aos métodos baseados nas estatísticas de alta ordem do sinal recebido é explorar outra característica intrínseca da saída do canal, denominada cicloestacionaridade. Dado um sinal  $y(t)$  com média igual a zero, sua função de autocorrelação é cicloestacionária se a relação

$$r_y(t_1, t_2) = E[y(t_1)y^*(t_2)] = r_y(t_1 + T, t_2 + T) \quad (2.29)$$

for verificada.

O que faz a cicloestacionaridade uma alternativa para a equalização cega é que ela é particularmente interessante pelo fato de usar apenas estatísticas de segunda ordem. O primeiro a reconhecer que as características cicloestacionárias de sinais modulados permitiam a recuperação da resposta completa do canal, em amplitude e fase, foi Gardner [4]. Contudo, a idéia de equalização de canal usando estatísticas cicloestacionárias é atribuída a Tong e Perreau [5]. A idéia original depende do uso da diversidade temporal, ou seja, sobreamostragem do sinal. Essa operação é realizada em sistemas digitais para o propósito de temporização e recuperação de fase. Entretanto, no contexto da equalização de canal, o uso de sobreamostragem nos leva à equalização fracionada, assim denominada porque os coeficientes do equalizador estão espaçados por um intervalo menor que  $T$ . Entre as várias técnicas existentes podemos mencionar o método de decomposição de subespaços descrito a seguir.

### 2.5.1 Equalização Cega Fracionalmente Espaçada por Decomposição de Subespaços Vetoriais

Suponha que um canal discreto seja modelado por um filtro FIR, e as medidas são realizadas a cada  $T$  segundos. Suponha também que sejam usadas antenas múltiplas na recepção, ou seja, o canal é subdividido em subcanais  $h^{[l]}(k)$ , tais que a relação entrada-saída para este sistema é dada por

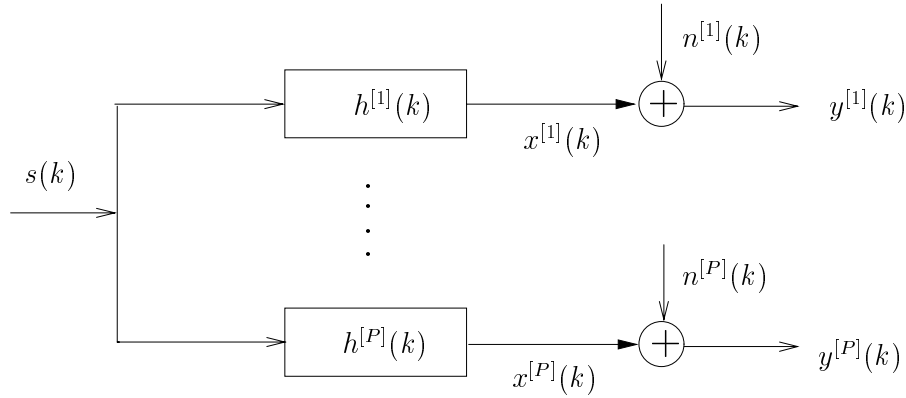


Figura 2.5: Sistema entrada única e múltiplas saídas.

$$y^{[l]}(k) = \sum_{m=-\infty}^{\infty} h^{[l]}(m)s(k-m) + n^{[l]}(k), \quad l = 1, \dots, P \quad (2.30)$$

$$h^{[l]}(m) = 0, \quad \text{para } m < 0 \text{ ou } m > L \text{ e para todo } l \quad (2.31)$$

onde  $L$  é a ordem de cada subcanal e  $P$  o número total de subcanais do sistema. Dependendo de como as amostras são agrupadas, podem-se distinguir duas representações matriciais diferentes para o sistema:

1. Representação modelo entrada única e saídas múltiplas (SIMO): Consiste de  $P$  subcanais e cada um é caracterizado pela observação de  $w$  amostras da saída do sistema

$$\mathbf{y}_w^{[l]}(k) = [y^{[l]}(k), y^{[l]}(k-1), \dots, y^{[l]}(k-w+1)]^T \quad (2.32)$$

$$\mathbf{n}_w^{[l]}(k) = [n^{[l]}(k), n^{[l]}(k-1), \dots, n^{[l]}(k-w+1)]^T \quad (2.33)$$



onde  $\mathbf{y}_w^{[l]}(k)$  e  $\mathbf{n}_w^{[l]}(k)$  são respectivamente, o sinal recebido e o ruído presente no  $l$ -ésimo subcanal,

$$\mathbf{h}^{[l]} = [h^{[l]}(0), \dots, h^{[l]}(L)]^T \quad (2.34)$$

é a resposta impulsional do  $l$ -ésimo subcanal. No instante  $k$ , o processamento envolve o uso de  $(L + w)$  símbolos da seqüência de entrada, ou seja

$$\mathbf{s}_{L+w}(k) = [s(k), s(k-1), \dots, s(k-L-w+1)]^T \quad (2.35)$$

Logo, pode-se representar a saída do sistema pela seguinte forma compacta

$$\mathbf{y}_w^{[l]}(k) = \mathbf{F}_w^{[l]}(\mathbf{h})\mathbf{s}_{L+w}(k) + \mathbf{n}_w^{[l]}(k) \quad (2.36)$$

onde a matriz  $\mathbf{F}_w^{[l]}(\mathbf{h})$ , de dimensão  $w \times (w + L)$ , é denominada matriz monocanal de filtragem de estrutura Toeplitz:

$$\mathbf{F}_w^{[l]}(\mathbf{h}) = \begin{bmatrix} h^{[l]}(0) & h^{[l]}(1) & \dots & h^{[l]}(L) & 0 & \dots & 0 \\ 0 & h^{[l]}(0) & \dots & h^{[l]}(L-1) & h^{[l]}(L) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & h^{[l]}(0) & h^{[l]}(1) & \dots & h^{[l]}(L) \end{bmatrix} \quad (2.37)$$

Combinando as  $P$  equações (2.36) na forma matricial teremos

$$\mathbf{y}'_w(k) = \mathcal{F}'_w(\mathbf{h})\mathbf{s}_{L+w}(k) + \mathbf{n}'_w(k) \quad (2.38)$$

onde  $\mathbf{y}'_w(k)$  é o vetor sinal recebido agrupado de todos os subcanais e  $\mathbf{n}'_w(k)$  é o vetor multicanal de ruído, isto é

$$\mathbf{y}'_w(k) = [\mathbf{y}_w^{[1]}(k)^T, \mathbf{y}_w^{[2]}(k)^T, \dots, \mathbf{y}_w^{[P]}(k)^T]^T \quad (2.39)$$

$$\mathbf{n}'_w(k) = [\mathbf{n}_w^{[1]}(k)^T, \mathbf{n}_w^{[2]}(k)^T, \dots, \mathbf{n}_w^{[P]}(k)^T]^T \quad (2.40)$$

e  $\mathcal{F}'_w(\mathbf{h})$ , de dimensão  $wP \times (w + L)$ , é a matriz multicanal de filtragem da forma

$$\mathcal{F}'_w(\mathbf{h}) = \begin{bmatrix} \mathbf{F}_w^{[1]}(\mathbf{h}) \\ \mathbf{F}_w^{[2]}(\mathbf{h}) \\ \vdots \\ \mathbf{F}_w^{[P]}(\mathbf{h}) \end{bmatrix} \quad (2.41)$$

cujas entradas estão descritas pela equação (2.37).

2. Representação pela matriz de Sylvester: Neste segundo modelo, os  $L$  coeficientes do canal são agrupados. Especificamente podemos escrever que

$$\mathbf{h}(m) = [h^{[1]}(m), h^{[2]}(m), \dots, h^{[P]}(m)]^T \quad m = 0, 1, \dots, L \quad (2.42)$$

Correspondentemente, definem-se o vetor de sinal recebido e o vetor de ruído

$$\mathbf{y}(k) = [y^{[1]}(k), y^{[2]}(k), \dots, y^{[P]}(k)]^T \quad (2.43)$$

$$\mathbf{n}(k) = [n^{[1]}(k), n^{[2]}(k), \dots, n^{[P]}(k)]^T \quad (2.44)$$

Agrupando as  $wP$  amostras do sinal recebido, temos

$$\mathbf{y}_w(k) = [\mathbf{y}^T(k), \mathbf{y}^T(k-1), \dots, \mathbf{y}^T(k-w+1)]^T = \mathcal{F}_w(\mathbf{h})\mathbf{s}_{L+w}(k) + \mathbf{n}_w(k) \quad (2.45)$$

onde  $\mathbf{n}_w(k)$  é o vetor de ruído multicanal definido por

$$\mathbf{n}_w(k) = [\mathbf{n}^T(k), \mathbf{n}^T(k-1), \dots, \mathbf{n}^T(k-w+1)]^T \quad (2.46)$$

o vetor  $\mathbf{s}_{L+w}(k)$  já foi definido em (2.35) e a matriz  $\mathcal{F}_w(\mathbf{h})$ , de dimensão  $wP \times (w + L)$ , é da forma

$$\mathcal{F}_w(\mathbf{h}) = \begin{bmatrix} \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(L) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{h}(0) & \dots & \mathbf{h}(L-1) & \mathbf{h}(L) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(L) \end{bmatrix} \quad (2.47)$$

Esta matriz, com estrutura Toeplitz em blocos, denomina-se matriz de filtragem de Sylvester, daí a denominação da segunda representação do modelo.

As matrizes  $\mathcal{F}_w(\mathbf{h})$  e  $\mathcal{F}'_w(\mathbf{h})$ , definidas pelas equações (2.47) e (2.41) respectivamente, diferem na maneira na qual suas colunas são organizadas, mas elas contêm a mesma informação sobre o canal de transmissão. Um fato importante diz que os espaços gerados pelas colunas de  $\mathcal{F}_w(\mathbf{h})$  e  $\mathcal{F}'_w(\mathbf{h})$  são canonicamente equivalentes [1]. Na equalização cega de canal, a matriz  $\mathcal{F}_w(\mathbf{h})$  representa um papel fundamental e o canal só pode ser equalizado corretamente se esta matriz tiver pôsto de colunas completo. Para isto ocorrer, devem ser satisfeitas as seguintes condições [1]:

- As funções de transferência dos subcanais

$$H^{[l]}(z) = \sum_{m=0}^L h^{[l]}(m)z^{-m} \quad l = 1, \dots, P \quad (2.48)$$

não devem possuir zeros em comum.

- Pelos menos um dos polinômios  $H^{[l]}(z)$  deve ter o grau máximo possível  $L$ .
- O número de amostras  $w$ , da observação do sinal recebido,  $\mathbf{y}^{[l]}(k)$  é maior que  $L$ , para cada subcanal.

Com as três condições acima satisfeitas, é possível descrever um método para a equalização cega baseada na decomposição de subespaços vetoriais, como será visto na próxima seção.

### 2.5.2 Algoritmo Baseado na Decomposição de Subespaços

Suponha que o sinal transmitido  $s(k)$  e o vetor ruído  $\mathbf{n}'(k)$  sejam processos estocásticos do tipo WSS (estacionário no sentido amplo, *wide sense stationary*) [1], e que também esses processos sejam estatisticamente independentes. Suponha que  $s(k)$  tenha média igual a zero e matriz de correlação

$$\mathbf{R}_s = E[\mathbf{s}_{L+w}(k)\mathbf{s}_{L+w}^H(k)] \quad (2.49)$$

O valor de  $w$  deve ser suficientemente grande tal que a matriz  $\mathbf{R}_s$  tenha pôsto<sup>1</sup> de colunas completo. O vetor  $\mathbf{n}'_w(k)$  também tem média igual a zero e matriz de correlação

$$\mathbf{R}_n = E[\mathbf{n}'_w(k)\mathbf{n}'_w(k)^H] = \sigma^2 \mathbf{I} \quad (2.50)$$

onde a variância do ruído  $\sigma^2$  é conhecida. Desse modo, o sinal recebido  $\mathbf{y}'_w(k)$  também tem média igual a zero e matriz de correlação

$$\begin{aligned} \mathbf{R}_y &= E[\mathbf{y}'_w(k)\mathbf{y}'_w(k)^H] \\ &= E[(\mathcal{F}'_w(\mathbf{h})\mathbf{s}_{L+w}(k) + \mathbf{n}'_w(k))(\mathcal{F}'_w(\mathbf{h})\mathbf{s}_{L+w}(k) + \mathbf{n}'_w(k))^H] \\ &= \mathcal{F}'_w(\mathbf{h})\mathbf{R}_s\mathcal{F}'_w(\mathbf{h})^H + \mathbf{R}_n \end{aligned} \quad (2.51)$$

A matriz de correlação  $\mathbf{R}_y$  pode ser escrita em função de seus autovalores e autovetores, como

$$\mathbf{R}_y = \sum_{m=0}^{wP-1} \lambda_m \mathbf{q}_m \mathbf{q}_m^H \quad (2.52)$$

onde  $\mathbf{q}_m$  é o  $m$ -ésimo autovetor associado ao  $m$ -ésimo autovalor  $\lambda_m$ . Os autovalores estão ordenados em ordem decrescente, ou seja

$$\lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_{wP-1} \quad (2.53)$$

Agora dividindo os autovalores em dois grupos tais que

$$\lambda_m > \sigma^2, \quad \text{para } m = 0, 1, \dots, (L + w - 1) \quad (2.54)$$

$$\lambda_m = \sigma^2, \quad \text{para } m = (L + w), \dots, (wP - 1) \quad (2.55)$$

podemos então identificar dois subespaços vetoriais distintos:

- Subespaço devido apenas ao sinal,  $\mathcal{S}$ , gerado pelos  $\mathbf{q}_m$  tais que os  $\lambda_m$  correspondentes satisfazem a equação (2.54).

---

<sup>1</sup>Pôsto de uma matriz é o número de linhas linearmente independentes não nulas.

- Subespaço devido apenas ao ruído,  $\mathcal{N}$ , gerado pelos  $\mathbf{q}_m$  tais que os  $\lambda_m$  correspondentes satisfazem a equação (2.55).

No subespaço do ruído fazemos

$$\mathbf{g}_i = \mathbf{q}_{L+w+i}, \quad \text{para } i = 0, 1, \dots, (wP - L - w - 1) \quad (2.56)$$

Por definição temos que

$$\mathbf{R}_y \mathbf{g}_i = \sigma^2 \mathbf{I} \mathbf{g}_i \quad (2.57)$$

Substituindo  $\mathbf{R}_n = \sigma^2 \mathbf{I}$  na equação (2.51), temos

$$\mathbf{R}_y = \mathcal{F}'_w(\mathbf{h}) \mathbf{R}_s \mathcal{F}'_w(\mathbf{h})^H + \sigma^2 \mathbf{I} \quad (2.58)$$

de forma que

$$\mathbf{R}_y \mathbf{g}_i = \mathcal{F}'_w(\mathbf{h}) \mathbf{R}_s \mathcal{F}'_w(\mathbf{h})^H \mathbf{g}_i + \sigma^2 \mathbf{I} \mathbf{g}_i = \sigma^2 \mathbf{I} \mathbf{g}_i \quad (2.59)$$

onde chegamos a

$$\mathcal{F}'_w(\mathbf{h}) \mathbf{R}_s \mathcal{F}'_w(\mathbf{h})^H \mathbf{g}_i = \mathbf{0} \quad (2.60)$$

Como as matrizes  $\mathcal{F}'_w(\mathbf{h})$  e  $\mathbf{R}_s$  têm posto de colunas completo, temos que

$$\mathcal{F}'_w(\mathbf{h})^H \mathbf{g}_i = \mathbf{0} \quad (2.61)$$

A partir da equação (2.61), podemos obter os coeficientes do canal de transmissão usando a decomposição do subespaço do sinal recebido. Como vimos, este método é baseado em:

- O conhecimento dos autovalores associados aos  $(wP - L - w)$  menores autovalores de  $\mathbf{R}_y$ .
- A ortogonalidade das colunas da matriz  $\mathcal{F}'_w(\mathbf{h})$  ao subespaço do ruído  $\mathcal{N}$  equação (2.61).

Ou seja, a cicloestacionaridade das estatísticas do sinal recebido,  $\mathbf{y}'_w(k)$  é suficiente para a identificação cega do canal a menos de uma constante de ganho. Para maiores detalhes sobre o método de decomposição de subespaços ver [1].

## 2.6 Resultados Experimentais

Nesta seção, é apresentada uma simulação onde desejamos usar o algoritmo Bussgang para a equalização de canal. Nos próximos capítulos, serão apresentadas simulações que envolvem algoritmos baseados na decomposição de subespaços.

Nesta experimento desejamos equalizar o seguinte canal:

$$\mathbf{h}_0 = \begin{bmatrix} -0.12 & 0.51 & 0.43 \end{bmatrix}^T \quad (2.62)$$

que representa uma inversão de fase no primeiro percurso. As figuras de mérito para desempenho foram o erro médio quadrático, (*mean squared error*, MSE) definido por

$$\text{MSE} = E[\mathbf{e}^H \mathbf{e}] \quad (2.63)$$

e a taxa de erro de símbolo do sinal equalizado na recepção, que quanto menor for, mais eficiente será o algoritmo. As medidas foram realizadas variando-se a razão sinal-ruído (SNR) na faixa entre 0 e 20 dB. Na Figura 2.6 são mostrados os gráficos de desempenho do algoritmo para o canal  $\mathbf{h}_0$ . Foram usadas neste caso 3000 amostras na seqüência de entrada para fazer a equalização do canal. Foi usada uma seqüência de entrada do tipo iid, composta de símbolos  $\pm 1$  gerados aleatoriamente, de modo a ter média zero. Foram realizados 50 experimentos para valores inteiros da SNR na faixa especificada, totalizando 150.000 símbolos transmitidos por valor fixo da SNR. O resultado final da taxa de erro, por SNR, foi obtido tirando-se a média dos valores encontrados nos 50 experimentos. Para o MSE, a curva de erro foi obtida com SNR = 20 dB.

Ao ver o desempenho do algoritmo Bussgang vemos que o mesmo não produz boa equalização, mesmo para bons valores de razão sinal-ruído, tais como 20 dB. Neste valor, a taxa de erro atinge o valor de  $6 \times 10^{-2}$  que é considerado alto. Porém a complexidade computacional do algoritmo é pequena: Para efetuar a equalização do canal deste exemplo, foram realizadas  $8 \times 10^4$  operações de

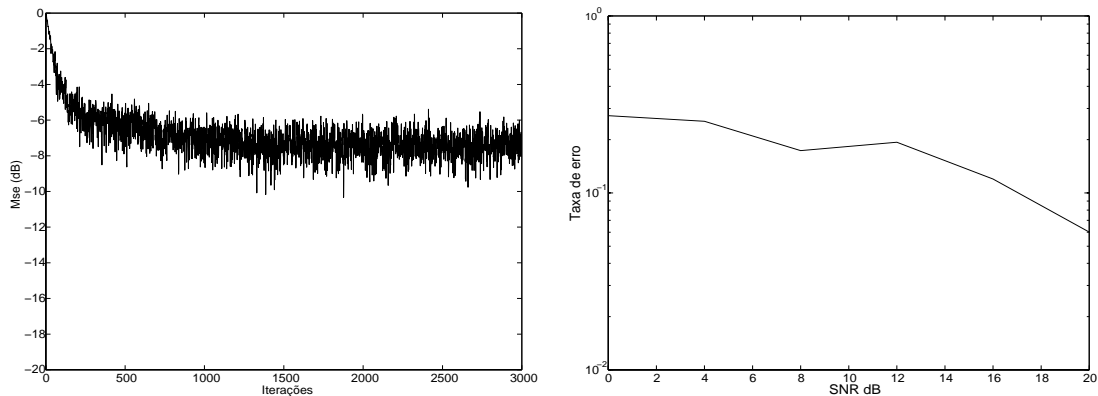


Figura 2.6: Desempenho do algoritmo Busgang para o canal  $h_0$ .

ponto flutuante. Conforme veremos no andamento deste trabalho, o algoritmo Busgang apresenta um custo computacional menor porém, seu desempenho, em relação à taxa de erro de bit do sinal equalizado, é inferior aos demais algoritmos de equalização cega avaliados, como os algoritmos baseados na decomposição de subespaços e algoritmo de predição de erro.

## 2.7 Conclusão

Neste capítulo, foram apresentadas duas grandes famílias de algoritmos para equalização cega divididas a partir das estatísticas do sinal recebido. Foi apresentado o algoritmo Busgang para a equalização direta de canais que, como vimos na seção 2.6, apresentou resultados ruins, se comparados aos resultados ainda a serem obtidos pelos demais algoritmos. Os métodos baseados em poliespectros e na decomposição de subespaços também foram apresentados, porém não avaliados. Os algoritmos pertencentes a este último grupo serão avaliados nos capítulos seguintes e como veremos, possuem desempenho superior ao algoritmo Busgang, no que se refere a taxa de erro do sinal equalizado.

# Capítulo 3

## Equalização Cega por Decomposição em Subespaços com Suavização

### 3.1 Introdução

Neste capítulo, é apresentado o algoritmo de detecção conjunta de ordem e estimação cega de canal por suavização mínima quadrática (*joint order detection and channel estimation by least squares smoothing*, J-LSS) [6]. O algoritmo J-LSS explora a relação isomórfica<sup>1</sup> entre os subespaços vetoriais gerados pela seqüência de entrada e pela seqüência de saída do canal, com a qual é possível determinar unicamente a ordem e a resposta impulsional do canal de transmissão. Na seção 3.2, a velocidade de convergência de um algoritmo de equalização é analisada, para este poder operar em tempo real. Na seção 3.3, o modelo de sistema para o algoritmo J-LSS é descrito. Na seção 3.4, suas propriedades gerais são descritas. Na seção 3.5, é dada a implementação do algoritmo. Na seção 3.6,

---

<sup>1</sup>Dois espaços vetoriais são isomórficos quando todas as transformações lineares definidas entre estes espaços forem do tipo injetora [7].



são apresentadas algumas simulações de equalização de canal, e na seção 3.7, são apresentadas as conclusões do capítulo.

## 3.2 Características Gerais Requeridas para Equalização de Canal

Uma das exigências mais importantes para a equalização de canal é a velocidade de convergência do algoritmo, especialmente no caso da transmissão realizada por pacotes quando existem poucas amostras de dados para processamento. Entre as técnicas recentes de equalização cega [5], aquelas que se baseiam em modelos determinísticos têm convergência mais rápida. Tal convergência rápida vem principalmente da exploração do uso da diversidade de canais, recepção por múltiplas antenas ou estrutura multicanais.

Os algoritmos existentes com a propriedade de convergência rápida a partir de um número finito de amostras têm uma dificuldade intrínseca: A determinação da ordem do canal. Enquanto muitos outros algoritmos de detecção podem ser aplicados, a detecção separada da ordem e estimação da resposta do canal pode não ser efetiva, especialmente quando a resposta impulsional do canal tem coeficientes na cauda, ou seja, os últimos coeficientes do canal são bem menores que os demais.

O algoritmo a ser apresentado neste capítulo possui um bom desempenho no que se refere à velocidade de convergência apresentando bom desempenho geral na equalização de canais. O algoritmo começa a ser descrito na próxima seção.

## 3.3 Modelo do Sistema

O sistema multicanal tipo SIMO com  $P$  saídas, como descrito pela Figura 2.5, será rerepresentado aqui para maior comodidade. Temos então, a seguinte

relação entrada-saída para este sistema

$$\begin{aligned}\mathbf{x}(k) &= \sum_{i=0}^L \mathbf{h}(i)s(k-i) \\ \mathbf{y}(k) &= \mathbf{x}(k) + \mathbf{n}(k), \quad k = 1, 2, \dots, N\end{aligned}\quad (3.1)$$

onde

$$\begin{aligned}\mathbf{x}(k) &= [x^{[1]}(k), \dots, x^{[P]}(k)]^T \\ \mathbf{n}(k) &= [n^{[1]}(k), \dots, n^{[P]}(k)]^T \\ \mathbf{y}(k) &= [y^{[1]}(k), \dots, y^{[P]}(k)]^T \\ \mathbf{h}(k) &= [h^{[1]}(k), \dots, h^{[P]}(k)]^T\end{aligned}\quad (3.2)$$

são, respectivamente, o sinal na saída do canal sem ruído, o ruído presente no sistema, o sinal recebido e a resposta impulsional do canal. Considerando um bloco de  $w$  amostras na observação da equação (3.1), temos os seguintes vetores

$$\begin{aligned}\mathbf{x}_w(k) &= [\mathbf{x}^T(k), \dots, \mathbf{x}^T(k-w+1)]^T \\ \mathbf{n}_w(k) &= [\mathbf{n}^T(k), \dots, \mathbf{n}^T(k-w+1)]^T \\ \mathbf{y}_w(k) &= [\mathbf{y}^T(k), \dots, \mathbf{y}^T(k-w+1)]^T \\ \mathbf{s}_{L+w}(k) &= [s(k), \dots, s(k-L-w+1)]^T\end{aligned}\quad (3.3)$$

Logo, podemos escrever

$$\begin{aligned}\mathbf{x}_w(k) &= \mathcal{F}_w(\mathbf{h})\mathbf{s}_{L+w}(k) \\ \mathbf{y}_w(k) &= \mathbf{x}_w(k) + \mathbf{n}_w(k)\end{aligned}\quad (3.4)$$

onde a matriz  $\mathcal{F}_w(\mathbf{h})$ , de dimensão  $wP \times (w+L)$ , é a matriz de filtragem de Sylvester

$$\mathcal{F}_w(\mathbf{h}) = \begin{bmatrix} \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(L) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{h}(0) & \dots & \mathbf{h}(L-1) & \mathbf{h}(L) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{h}(0) & \mathbf{h}(1) & \dots & \mathbf{h}(L) \end{bmatrix}\quad (3.5)$$

No problema de equalização a meta é estimar  $\mathbf{h} \triangleq [\mathbf{h}^H(L), \dots, \mathbf{h}^H(0)]^H$  a partir de  $\mathbf{y}_w(k)$ , para  $k = 1, \dots, N$ .

São feitas aqui duas suposições que serão consideradas no desenvolvimento que se segue [6]:

- S3.1) Existe um valor  $w_0$  tal que a matriz de filtragem  $\mathcal{F}_{w_0}(\mathbf{h})$  tenha pôsto de colunas completo.
- S3.2) A seqüência de entrada  $s(k)$  tem complexidade linear<sup>2</sup> maior que  $L_* = 2w_0 + 2L$ , ou seja,

$$\text{pôsto} \left\{ \left[ \begin{array}{ccc} s(L_* - L + 1) & \dots & s(N) \\ \vdots & \text{Toeplitz} & \\ s(1 - L) & & \end{array} \right] \right\} = L_* + 1. \quad (3.6)$$

A suposição S3.2 assegura que a seqüência de entrada possui informação suficiente para identificar o canal e está relacionada à excitação persistente.

Em decorrência da suposição S3.1 são estabelecidas as seguintes propriedades [6]:

- P3.1) As funções de transferência dos subcanais,  $H^{[l]}(z)$ , não possuem zeros em comum.
- P3.2)  $\mathcal{F}_w(\mathbf{h})$  tem pôsto de colunas completo para todo  $w \geq w_0$ .

Estas propriedades são uma decorrência direta da suposição S3.1 e o canal só pode ser corretamente identificado se as suposições S3.1 e S3.2 forem satisfeitas.

---

<sup>2</sup>A complexidade linear de um vetor  $\mathbf{s}$  com componentes  $s(0), \dots, s(k-1)$  é definida como o menor valor de  $c$  para qual a recursão na forma  $s(i) = \sum_{j=1}^c a(j)s(i-j)$ ,  $i = c, \dots, k-1$  existe, de modo a se gerar  $\mathbf{s}$  a partir dos  $c$  primeiros componentes [8].

## 3.4 Algoritmo de Detecção Conjunta de Ordem e Estimação Cega de Canal por Suavização Mínima Quadrática

A idéia essencial por trás da predição linear e suavização para estimação de canal, segue da relação direta entre os subespaços vetoriais da entrada e da saída. É essa relação entre os dois subespaços que nos permite evitar usar diretamente a seqüência de entrada, usando em seu lugar o subespaço vetorial obtido pela observação de  $\mathbf{x}(k)$ .

### 3.4.1 Definições e Relações entre os Subespaços

Da equação (3.1), sejam  $\mathbf{s}(k)$  um vetor linha dos símbolos de entrada e  $\mathbf{X}(k)$  a matriz de dados da observação sem ruído definidos, respectivamente, por

$$\mathbf{s}(k) = [s(k), s(k+1), \dots] \quad (3.7)$$

$$\mathbf{X}(k) = [\mathbf{x}(k), \mathbf{x}(k+1), \dots] \quad (3.8)$$

Considere os seguintes espaços vetoriais gerados pelas linhas dos  $p > 0$  vetores consecutivos de  $\mathbf{s}(k)$  e de  $\mathbf{x}(k)$

$$\mathcal{S}_{k,p} \triangleq sp\{\mathbf{s}(k), \dots, \mathbf{s}(k-p+1)\} = sp \left\{ \begin{bmatrix} s(k) & \dots \\ \vdots & \text{Toeplitz} \\ s(k-p+1) \end{bmatrix} \right\} \quad (3.9)$$

$$\mathcal{X}_{k,p} \triangleq sp\{\mathbf{X}(k), \dots, \mathbf{X}(k-p+1)\} = sp \left\{ \begin{bmatrix} \mathbf{x}(k) & \dots \\ \vdots & \text{Toeplitz} \\ \mathbf{x}(k-p+1) \end{bmatrix} \right\} \quad (3.10)$$

onde  $sp\{\mathcal{A}\}$  significa espaço vetorial gerado pelas linhas da matriz  $\mathcal{A}$ . As definições acima ainda se aplicam para  $p < 0$ , porém, nesse caso, nós temos o espaço gerado pelos  $|p|$  vetores de dados do futuro. Também é importante notar que

$$\begin{aligned}\mathcal{S}_{k,p} &= \mathcal{S}_{k-p+1,-p} \\ \mathcal{X}_{k,p} &= \mathcal{X}_{k-p+1,-p}\end{aligned}\tag{3.11}$$

Dado um subespaço linear  $\mathcal{S}$ , a projeção ortogonal  $\hat{\mathbf{s}}(k|\mathcal{S})$  de  $\mathbf{s}(k)$  em  $\mathcal{S}$  e seu erro de projeção  $\tilde{\mathbf{s}}(k|\mathcal{S})$  são definidos por

$$\begin{aligned}\hat{\mathbf{s}}(k|\mathcal{S}) &= \arg \min_{z \in \mathcal{S}} \|\mathbf{s}(k) - z\|^2 \\ \tilde{\mathbf{s}}(k|\mathcal{S}) &= \mathbf{s}(k) - \hat{\mathbf{s}}(k|\mathcal{S})\end{aligned}\tag{3.12}$$

e igualmente para o subespaço da saída temos que

$$\begin{aligned}\hat{\mathbf{x}}(k|\mathcal{X}) &= \arg \min_{z \in \mathcal{X}} \|\mathbf{x}(k) - z\|^2 \\ \tilde{\mathbf{x}}(k|\mathcal{X}) &= \mathbf{x}(k) - \hat{\mathbf{x}}(k|\mathcal{X})\end{aligned}\tag{3.13}$$

Ainda sobre a suposição S3.1 e pela equação (3.4) temos que

$$\mathcal{X}_{k,w} = \mathcal{S}_{k,L+w} \quad \text{para } w \geq w_0\tag{3.14}$$

ou seja,  $\mathcal{X}_{k,w}$  é isomórfico a  $\mathcal{S}_{k,L+w}$  com isomorfismo  $\mathcal{F}_w(\mathbf{h})$ .

Em geral, pela equação (3.14),  $\mathcal{X}_{k,w} \subseteq \mathcal{S}_{k,L+w}$  para algum  $w$ . Isso implica que para uma dada janela de observação fixa  $w$ , o subespaço de entrada  $\mathcal{S}_{k,L+w}$  pode não ser completamente visto pelo subespaço da saída  $\mathcal{X}_{k,w}$ . De outra maneira, por P3.1, toda a informação do subespaço de entrada está contida no subespaço de saída  $\mathcal{X}_{k,w}$  quando os subcanais não possuem zeros em comum e quando  $w$  é escolhido suficientemente grande.

### 3.4.2 Suavização Mínima Quadrática

O isomorfismo entre os subespaços da entrada e da saída permite que o canal seja identificado a partir de  $\mathcal{X}_{k,w}$  sem o conhecimento prévio da seqüência de

entrada. Para tal, precisamos construir subespaços que contenham tanto dados do passado quanto do futuro, esse processo é denominado suavização.

Vamos assumir por enquanto que  $N \rightarrow \infty$ . Seja  $\dot{\mathcal{S}}_k$  o subespaço que contenha todas as amostras do passado e do futuro, exceto  $\mathbf{s}(k)$ , isto é

$$\dot{\mathcal{S}}_k = sp\{\dots, \mathbf{s}(k-1)\} \cup sp\{\mathbf{s}(k+1), \dots\} = \mathcal{S}_{k-1, \infty} \cup \mathcal{S}_{k+1, -\infty} \quad (3.15)$$

Devido ao isomorfismo entre  $\mathcal{X}_{k,w}$  e  $\mathcal{S}_{k,L+w}$  temos que o erro de projeção do sinal recebido no subespaço da entrada é igual ao erro de projeção do sinal recebido no subespaço da saída [6], ou seja

$$\mathbf{E} = \begin{bmatrix} \tilde{\mathbf{x}}^H(k+L|\dot{\mathcal{S}}_k) \\ \vdots \\ \tilde{\mathbf{x}}^H(k|\dot{\mathcal{S}}_k) \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{x}}^H(k+L|\dot{\mathcal{X}}_k) \\ \vdots \\ \tilde{\mathbf{x}}^H(k|\dot{\mathcal{X}}_k) \end{bmatrix} = \mathbf{h} \tilde{\mathbf{s}}(k|\dot{\mathcal{X}}_k) \quad (3.16)$$

onde o subespaço da saída  $\dot{\mathcal{X}}_{k,L}$  é definido, devido ao isomorfismo com  $\dot{\mathcal{S}}_k$ , por

$$\dot{\mathcal{X}}_{k,L} = \mathcal{X}_{k-1, \infty} \cup \mathcal{X}_{k+L+1, -\infty} = \dot{\mathcal{S}}_k \quad (3.17)$$

A partir de  $\mathbf{E}$  existem várias maneiras para se encontrar  $\mathbf{h}$  até um fator de escalamento, isso porque os subespaços são invariantes com respeito ao escalamento. Uma solução seria pelo método do ajuste de mínimos quadrados do subespaço gerado pelas colunas da matriz  $\mathbf{E}$ , ou seja

$$\hat{\mathbf{h}} = \arg \max_{\|\mathbf{h}\|=1} \|\mathbf{h}^H \mathbf{E}\|^2 \quad (3.18)$$

A solução para essa otimização pode ser obtida pela decomposição de valores singulares [1] tanto de  $\mathbf{E}$  quanto da matriz de covariância da seqüência de erro de projeção definida por

$$\hat{\mathbf{R}}_E = (1/M) \mathbf{E} \mathbf{E}^H \quad (3.19)$$

onde  $M$  é o número de colunas de  $\mathbf{E}$ .

## 3.5 Implementação do Algoritmo J-LSS

Nesta seção, são dados maiores detalhes sobre o método de suavização mínima quadrática incluindo algumas de suas propriedades e implementações. Aqui é apresentada uma solução para a equação (3.16) em termos da observação dos dados obtidos no receptor do sistema.

### 3.5.1 Formulação Geral do Método J-LSS

Como o subespaço do sinal recebido  $\dot{\mathcal{X}}_{k,L}$  se baseia em uma estimativa da ordem do canal, consideraremos aqui um problema de formulação um pouco diferente no qual os subespaços de projeção são definidos ligeiramente diferentes. Isto nos permite realizar implementações do algoritmo baseadas em um número finito de amostras do sinal recebido e também a partir da ordem desconhecida do canal. Ao invés de usar o subespaço de projeção da equação (3.17), vamos considerar a suavização de  $l + 1$  observações de  $\mathbf{x}_{t+i}, i = 0, \dots, l$  através do uso de preditores progressivos e regressivos de ordem  $w > w_0$ . Este subespaço de projeção é dado por

$$\dot{\mathcal{X}}_{k,l} = \mathcal{X}_{k-1,w} \cup \mathcal{X}_{k+l+1,-w} = \mathcal{X}_{k-1,w} \cup \mathcal{X}_{k+l+w,w} \quad (3.20)$$

Notar que  $\dot{\mathcal{X}}_{k,l}$  é uma aproximação para a equação (3.17) onde aqui tratamos  $l$  como não sendo necessariamente igual a ordem do canal  $L$ . Devido a relação direta entre os subespaços da saída e da entrada, temos que

$$\dot{\mathcal{X}}_{k,l} = \mathcal{S}_{k-1,L+w} \cup \mathcal{S}_{k+l+w,L+w} = \dot{\mathcal{S}}_{k,l} \quad (3.21)$$

Desse modo, tem-se

$$\dot{\mathcal{S}}_{k,l} = \begin{cases} sp\{\mathbf{s}_{k-L-w}, \dots, \mathbf{s}_{k-1}, \dots, \mathbf{s}_{k+l+w}\} & l < L \\ sp\{\mathbf{s}_{k-L-w}, \dots, \mathbf{s}_{k-1}\} \cup sp\{\mathbf{s}_{k+l-L+1}, \dots, \mathbf{s}_{k+l+w}\} & l \geq L \end{cases} \quad (3.22)$$

A seguir, apresentamos as estruturas de dados para a implementação do algoritmo.

### 3.5.2 Estruturas de Dados e Identificação de Canal

Consideraremos a partir de agora o problema prático de se estimar o canal usando apenas um número finito de amostras do sinal recebido  $\mathbf{y}(k)$ ,  $k = 1, \dots, N$ . Para um valor fixo da ordem do preditor  $w$ ,  $w \geq w_0$ , definida como o número de símbolos de dados no passado, e da janela de suavização  $l$ ,  $l \geq 0$ , definida como o número de símbolos de dados no presente, definimos a matriz de dados do passado  $\mathbf{P}_{w,l}$ , matriz de dados do futuro  $\mathbf{F}_{w,l}$ , matriz de dados do presente  $\mathbf{Y}_{w,l}$  e a matriz de projeção de dados  $\mathbf{D}_{w,l}$  por

$$\begin{aligned}
 \mathbf{P}_{w,l} &\triangleq [\mathbf{y}_w(w), \dots, \mathbf{y}_w(N - w - l - 1)] \\
 \mathbf{F}_{w,l} &\triangleq [\mathbf{y}_w(2w + l + 1), \dots, \mathbf{y}_w(N)] \\
 \mathbf{Y}_{w,l} &\triangleq [\mathbf{y}_w(w + l + 1), \dots, \mathbf{y}_w(N - w)] \\
 \mathbf{D}_{w,l} &\triangleq \begin{bmatrix} \mathbf{F}_{w,l} \\ \mathbf{P}_{w,l} \end{bmatrix}
 \end{aligned} \tag{3.23}$$

ou seja

$$\mathbf{P}_{w,l} = \begin{bmatrix} y^{[1]}(w) & y^{[1]}(w+1) & \dots & y^{[1]}(N-w-l-1) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w) & y^{[P]}(w+1) & \dots & y^{[P]}(N-w-l-1) \\ \vdots & \vdots & \dots & \vdots \\ y^{[1]}(1) & y^{[1]}(2) & \dots & y^{[1]}(N-2w-l) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(1) & y^{[P]}(2) & \dots & y^{[P]}(N-2w-l) \end{bmatrix}$$

$$\mathbf{F}_{w,l} = \begin{bmatrix} y^{[1]}(2w+l+1) & y^{[1]}(2w+l+2) & \dots & y^{[1]}(N) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(2w+l+1) & y^{[P]}(2w+l+2) & \dots & y^{[P]}(N) \\ \vdots & \vdots & \dots & \vdots \\ y^{[1]}(w+l+2) & y^{[1]}(w+l+3) & \dots & y^{[1]}(N-w+1) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w+l+2) & y^{[P]}(w+l+3) & \dots & y^{[P]}(N-w+1) \end{bmatrix}$$



$$\mathbf{Y}_{w,l} = \begin{bmatrix} y^{[1]}(w+l+1) & y^{[1]}(w+l+2) & \dots & y^{[1]}(N-w) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w+l+1) & y^{[P]}(w+l+2) & \dots & y^{[P]}(N-w) \\ \vdots & \vdots & \dots & \vdots \\ y^{[1]}(w+1) & y^{[1]}(w+2) & \dots & y^{[1]}(N-2w+1) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w+1) & y^{[P]}(w+2) & \dots & y^{[P]}(N-2w+1) \end{bmatrix}$$
  

$$\mathbf{D}_{w,l} = \begin{bmatrix} y^{[1]}(2w+l+1) & y^{[1]}(2w+l+2) & \dots & y^{[1]}(N) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(2w+l+1) & y^{[P]}(2w+l+2) & \dots & y^{[P]}(N) \\ \vdots & \vdots & \dots & \vdots \\ y^{[1]}(w+l+2) & y^{[1]}(w+l+3) & \dots & y^{[1]}(N-w+1) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w+l+2) & y^{[P]}(w+l+3) & \dots & y^{[P]}(N-w+1) \\ \\ y^{[1]}(w) & y^{[1]}(w+1) & \dots & y^{[1]}(N-w-l-1) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(w) & y^{[P]}(w+1) & \dots & y^{[P]}(N-w-l-1) \\ \vdots & \vdots & \dots & \vdots \\ y^{[1]}(1) & y^{[1]}(2) & \dots & y^{[1]}(N-2w-l) \\ \vdots & \vdots & & \vdots \\ y^{[P]}(1) & y^{[P]}(2) & \dots & y^{[P]}(N-2w-l) \end{bmatrix}$$

Considere agora a matriz erro de suavização  $\mathbf{E}_{l,l}$  definida aqui como  $\mathbf{E}_{l,l} = \mathcal{P}_{\mathbf{D}_{l,l}}^\perp \{ \mathbf{Y}_{l,l} \}$  obtida pela projeção ortogonal de  $\mathbf{Y}_{l,l}$  sobre o espaço vetorial gerado pelas linhas da matriz  $\mathbf{D}_{l,l}$ , matematicamente

$$\mathbf{E}_{l,l} = [\mathbf{I} - \mathbf{D}_{l,l}(\mathbf{D}_{l,l}^H \mathbf{D}_{l,l})^{-1} \mathbf{D}_{l,l}^H] \mathbf{Y}_{l,l} \quad (3.24)$$

Sendo  $\mathbf{Q} = [\mathbf{Q}_0, \dots, \mathbf{Q}_l]$  uma matriz cujos vetores linha são ortogonais ao espaço gerado por  $\mathbf{E}_{l,l}$ , então teremos

$$[\mathbf{Q}_0, \dots, \mathbf{Q}_l] \mathbf{E}_{l,l} = 0 \quad (3.25)$$

A solução que permite encontrar o canal a partir da equação (3.24) é dada pelo seguinte teorema:

*Teorema 3.1:* Suponha que não há ruído presente no sistema, que os subcanais não possuam zeros em comum e que seja a matriz de covariância da seqüência de erro de suavização,  $\hat{\mathbf{R}}_E$ , definida aqui por [6]

$$\hat{\mathbf{R}}_E \triangleq (1/N - 3l) \mathbf{E}_{l,l} \mathbf{E}_{l,l}^H \quad (3.26)$$

Assumindo também que as linhas de  $\mathbf{Q}$  que são os vetores singulares associados com os  $[P(l+1) - l + m - 1]$  menores valores singulares de  $\hat{\mathbf{R}}_E$  e que  $\mathbf{Q}$  seja particionada em  $(l+1)$  submatrizes de dimensão  $[P(l+1) - l + m - 1] \times P$ , ou seja  $\mathbf{Q} \triangleq [\mathbf{Q}_0, \dots, \mathbf{Q}_l]$ , e definindo

$$\mathcal{T}_m(\mathbf{Q}) \triangleq \begin{bmatrix} \mathbf{Q}_0 & \dots & \mathbf{Q}_m \\ & \text{Hankel} & \vdots \\ & \text{em blocos} & \mathbf{Q}_l \end{bmatrix} \quad (3.27)$$

Então a equação linear homogênea

$$\mathcal{T}_m(\mathbf{Q})\mathbf{z} = \mathbf{0}, \quad \text{para } m = 1, \dots, (l-1) \quad (3.28)$$

tem solução única, não trivial,  $\mathbf{z} = \alpha \mathbf{h}$ , quando  $m = L$ , ou seja, a ordem do canal. Para a prova do teorema ver [6].

Assim o critério para detecção de ordem e estimação conjunta de canal fica estabelecido:

$$\{\hat{L}, \hat{\mathbf{h}}\} = \arg \min_{m, \|\mathbf{h}\|=1} \|\mathcal{T}_m(\mathbf{Q})\mathbf{h}\|^2 \quad (3.29)$$

Existem várias maneiras de se implementar o algoritmo descrito para a estimação de canal. A forma escolhida neste trabalho foi a seguinte:

Seleção da ordem dos preditores  $w$ : Na seleção da ordem  $w$  dos preditores progressivos e regressivos, nós devemos notar dois fatos: Primeiro para um número

fixo de dados, um  $w$  grande implica em um número pequeno de colunas nas matrizes de dados. Neste caso, é desejável escolher  $w$  o menor possível, esta sendo a razão pela qual escolhemos  $w = l$  no algoritmo. Certamente para  $P > 2$  um valor menor para  $w$  pode ser escolhido. Porém, grandes valores de  $w$  dão um certo grau de robustez ao algoritmo, especialmente quando os subcanais têm zeros em comum e próximos ao círculo de raio unitário no plano  $z$ .

### 3.5.3 Resumo do Algoritmo J-LSS

Aqui é apresentado um resumo do algoritmo J-LSS para implementação.

- Escolher  $l > L$  e formar as matrizes de dados  $\mathbf{Y}_{l,l}$  e  $\mathbf{D}_{l,l}$ .
- Obter as  $4l$  bases ortogonais  $\{\mathbf{u}_1, \dots, \mathbf{u}_{4l}\}$  que geram o espaço vetorial linha de  $\mathbf{D}_{l,l}$ .
- Obter o erro de projeção de  $\mathbf{Y}_{l,l}$  sobre o espaço gerado por  $\{\mathbf{u}_1, \dots, \mathbf{u}_{4l}\}$

$$\mathbf{E}_{l,l} \triangleq \mathbf{Y}_{l,l} - \mathbf{Y}_{l,l} \mathbf{U}^H \mathbf{U}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{4l} \end{bmatrix} \quad (3.30)$$

- Para cada  $1 \leq m < l$ , tratado como a ordem estimada do canal, seja  $\mathbf{Q} = [\mathbf{Q}_0, \dots, \mathbf{Q}_l]$  a matriz cujos vetores linha são os  $[P(l+1) - l + m - 1]$  vetores singulares à esquerda de  $\mathbf{E}_{l,l}$ , ou equivalentemente, da matriz de covariância de erro da seqüência de suavização,  $\hat{\mathbf{R}}_E$ . Formar

$$\mathcal{T}_m(\mathbf{Q}) \triangleq \begin{bmatrix} \mathbf{Q}_0 & \dots & \mathbf{Q}_m \\ & \text{Hankel} & \vdots \\ & \text{em blocos} & \mathbf{Q}_l \end{bmatrix} \quad (3.31)$$

- Fazer a detecção da ordem conjuntamente com a estimação do canal para cada  $1 \leq m < l$

$$\{\hat{L}, \hat{\mathbf{h}}\} = \arg \min_{m, \|\mathbf{h}\|=1} \|\mathcal{T}_m(\mathbf{Q})\mathbf{h}\|^2 \quad (3.32)$$

- A solução ótima para o último item pode ser dada por

$$\boldsymbol{\xi} = \mathbf{h}^H \mathcal{T}^H \mathcal{T} \mathbf{h} - \lambda(\mathbf{h}^H \mathbf{h} - 1)$$

derivando em relação a  $\mathbf{h}$  teremos

$$\frac{\partial \boldsymbol{\xi}}{\partial \mathbf{h}} = 2\mathcal{T}^H \mathcal{T} \mathbf{h} - 2\lambda \mathbf{h} = 0$$

logo

$$\mathcal{T}^H \mathcal{T} \mathbf{h} = \lambda \mathbf{h}$$

multiplicando à esquerda ambos os lados da equação acima por  $\mathbf{h}^H$  teremos

$$\mathbf{h}^H \mathcal{T}^H \mathcal{T} \mathbf{h} = \lambda \quad (3.33)$$

Assim,  $\mathbf{h}$  é autovetor associado ao menor autovalor da matriz quadrada  $\mathcal{T}_m^H(\mathbf{Q})\mathcal{T}_m(\mathbf{Q})$ , com a restrição de possuir norma unitária.

## 3.6 Resultados Experimentais

O algoritmo foi implementado no ambiente matlab e verificou-se que seu desempenho dependia fundamentalmente do tipo de canal a ser estimado. A figura de mérito para desempenho foi o erro médio quadrático normalizado, (*normalized root mean squared error*, NRMSE) definido por [6]

$$\text{NRMSE} \triangleq \sqrt{\frac{1}{N_j \|\mathbf{h}\|^2} \sum_{j=1}^{N_j} \|\hat{\mathbf{h}}^{(j)} - \mathbf{h}\|^2} \quad (3.34)$$

onde  $N_j$  é o número de tentativas para cada valor de razão sinal-ruído e  $\hat{\mathbf{h}}^{(j)}$  é a

resposta estimada do canal na  $j$ -ésima tentativa. A razão sinal-ruído é definida por

$$\text{SNR} \triangleq \frac{1}{P\sigma^2} E \left[ \sum_{m=1}^P |x^{[m]}(k)|^2 \right] \quad (3.35)$$

onde  $\sigma^2$  é a variância do ruído. A seqüência de entrada foi obtida através das amostras de um sinal modulado em BPSK. O ruído usado durante as simulações tinha média zero e função de autocorrelação indicada pela Figura 3.1.

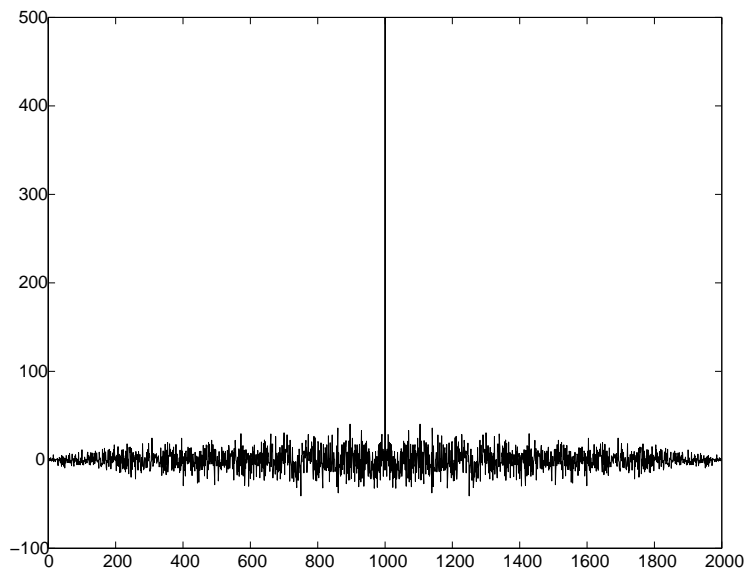


Figura 3.1: Função de autocorrelação do ruído usado durante as simulações.

Experimento 1: Primeiramente, realizamos a equalização do seguinte canal:

$$\mathbf{h}_1 = \begin{bmatrix} 0.345 & -0.016 & 0.209 & 0.253 & -0.213 & 0.254 & 0.118 & 0.483 & -0.034 & 0.462 & -0.111 & -0.285 \\ -0.715 & 0.690 & 0.120 & 0.388 & 0.132 & -0.120 & -0.388 & 0.451 & -0.204 & 0.56 & -0.675 & 0.147 \end{bmatrix}^T \quad (3.36)$$

Neste exemplo de canal, a matriz de filtragem era bem condicionada, pois tinha número de condicionamento<sup>3</sup> próximo de 100. Na Figura 3.2 é mostrado o

<sup>3</sup>O número de condicionamento de uma matriz quadrada é definido pela razão entre o maior e o menor autovalores desta matriz, respectivamente. Um canal onde os subcanais tenham zeros

diagrama de zeros do canal  $\mathbf{h}_1$ .

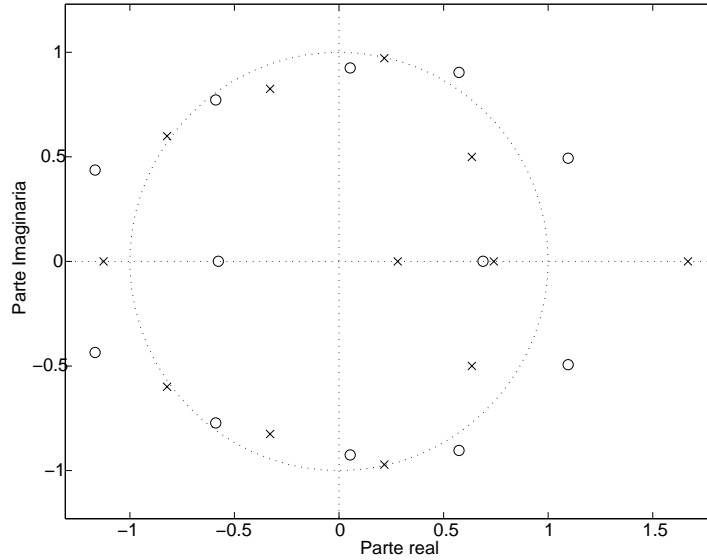


Figura 3.2: Diagrama de zeros do canal  $\mathbf{h}_1$ . Subcanal 1 'o', subcanal 2 'x'.

Para verificar o desempenho do algoritmo, calculamos o NRMSE e também a taxa de erro de símbolos na recepção. As medidas foram realizadas variando-se a razão sinal-ruído na faixa entre 0 e 18 dB. Na Figura 3.4 são mostrados os gráficos de desempenho do algoritmo para o canal  $\mathbf{h}_1$ . Foram usadas neste caso  $N = 180$  amostras no sinal de entrada para fazer a estimação do canal. Logo após a estimação do canal, a taxa de erro de símbolos (bits) foi determinada a partir da equalização do canal estimado, mantendo-se o canal,  $h^{[1]}(k)$ , e o equalizador,  $eq^{[1]}(k)$ , com respostas impulsivas constantes e usando como seqüência de entrada  $s(k)$ , de 1000 bits modulados em BPSK, vide Figura 3.3. Foram realizados 100 experimentos, para valores inteiros da razão sinal-ruído na faixa especificada, totalizando 100.000 bits transmitidos por SNR. O resultado final, por SNR, foi obtido tirando-se a média dos valores encontrados, tanto para o NRMSE quanto para a taxa de erro.

próximos, possui uma matriz de filtragem com número de condicionamento elevado, tipicamente maior que  $10^3$  e isso produzirá resultados ruins.

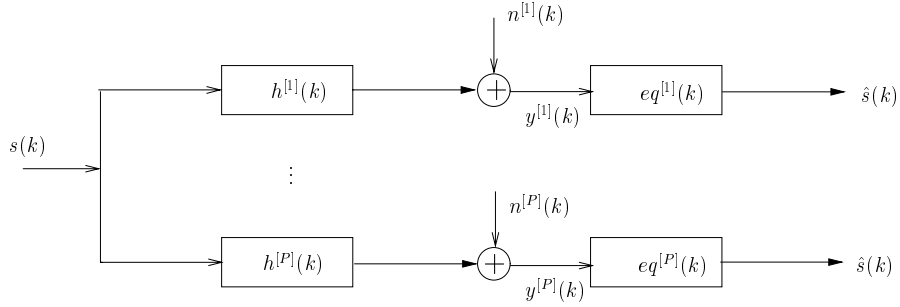
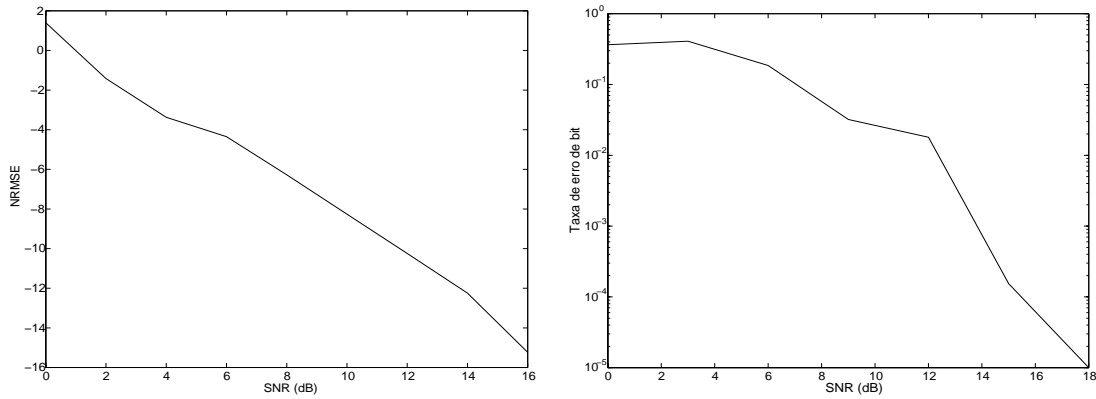


Figura 3.3: Sistema resultante equalizado.


 Figura 3.4: Desempenho do algoritmo J-LSS para o canal  $\mathbf{h}_1$ .

Para este canal, que possui grande interferência entre símbolos, constata-se o bom funcionamento do algoritmo principalmente no critério taxa de erro, pois vemos que quando a razão sinal-ruído atinge o valor de 18 dB, a taxa de erro é de apenas 1 bit a cada 100.000, ou seja,  $10^{-5}$ , valor que pode ser considerado bom para equalização de sistemas em tempo real.

Experimento 2: Agora o algoritmo J-LSS é usado para equalizar o canal  $\mathbf{h}_2$  cuja resposta impulsional produz uma matriz de filtragem que pode ser considerada mal condicionada com número de condicionamento próximo de  $3.6 \times 10^4$ .

$$\mathbf{h}_2 = \begin{bmatrix} 0.049 & -0.247 & 0.525 & -0.628 & 0.462 & -0.216 & 0.064 & -0.011 & 0.201 & -0.01 \\ 0.035 & -0.194 & 0.465 & -0.626 & 0.518 & -0.273 & 0.090 & -0.018 & 0.302 & -0.02 \end{bmatrix}^T \quad (3.37)$$

Pela Figura 3.5 pode-se constatar a proximidade dos zeros dos subcanais.

Para verificar o desempenho do algoritmo na equalização deste canal, computamos o NRMSE e a taxa de erro de bit do sinal equalizado, agora variando-se a SNR na faixa entre 0 e 45 dB. Na Figura 3.6 e são mostrados os gráficos de desempenho do algoritmo. Foram usadas as mesmas quantidades da simulação anterior.

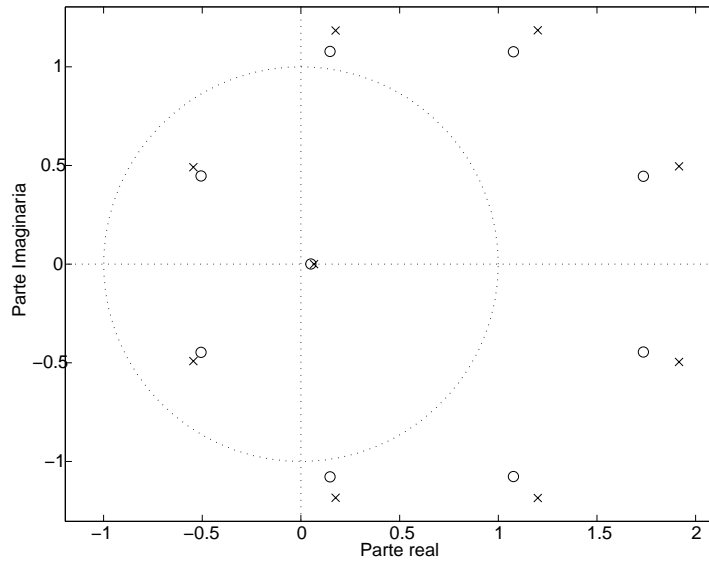


Figura 3.5: Diagrama de zeros do canal  $h_2$ . Subcanal 1 'o', subcanal 2 'x'.

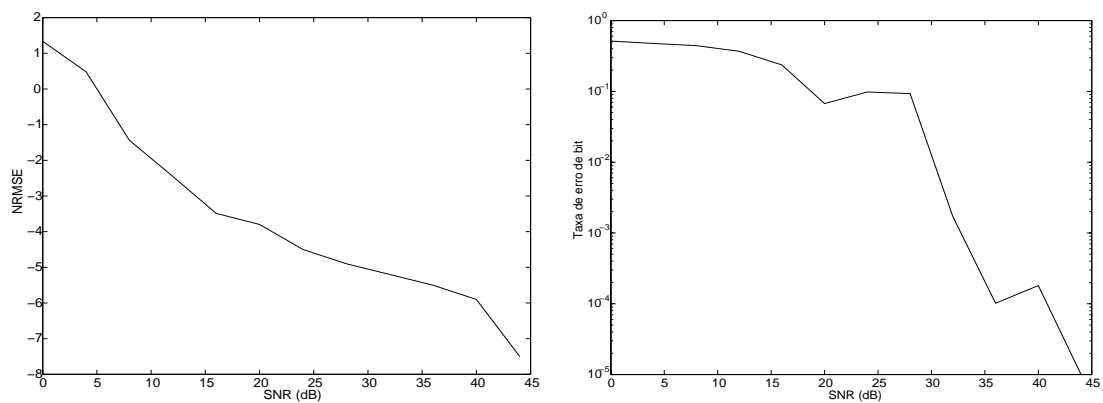


Figura 3.6: Desempenho do algoritmo J-LSS para o canal  $h_2$ .

Neste exemplo constata-se o desempenho ruim do algoritmo, especialmente



na faixa de 0 a 30 dB. A partir de 30 dB verifica-se uma melhoria sensível na taxa de erro.

Experimento 3: Aqui o algoritmo é usado para equalizar o canal  $\mathbf{h}_3$  cuja resposta impulsional produz uma matriz de filtragem que pode ser considerada um meio termo entre os dois experimentos anteriores.

$$\mathbf{h}_3 = \begin{bmatrix} 0.000 & 0.129 & 0.324 & 0.556 & 0.779 & 0.941 & 1.000 & 0.941 & 0.779 & 0.556 & 0.324 & 0.129 & 0.000 \\ 0.150 & 0.322 & 0.536 & 0.763 & 0.963 & 1.101 & 1.150 & 1.101 & 0.963 & 0.763 & 0.536 & 0.321 & 0.150 \end{bmatrix}^T \quad (3.38)$$

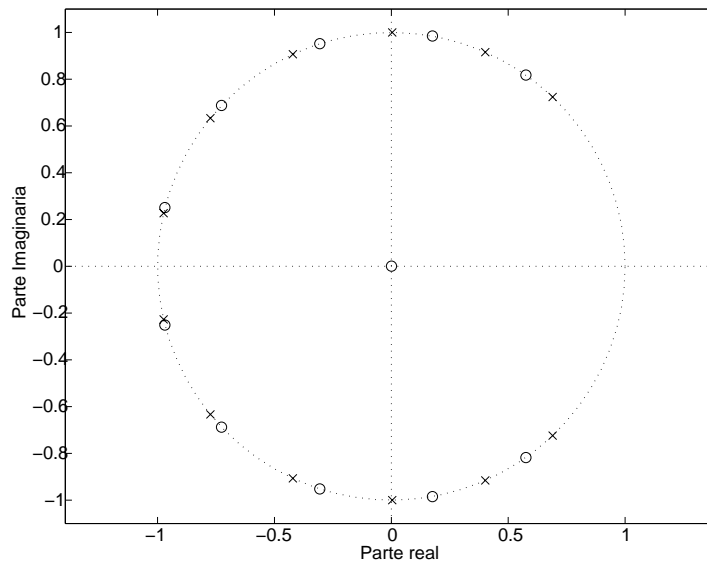


Figura 3.7: Diagrama de zeros do canal  $\mathbf{h}_3$ . Subcanal 1 'o', subcanal 2 'x'.

Pela Figura 3.7 pode-se constatar também a proximidade dos zeros dos subcanais neste exemplo, porém não tanto como no exemplo anterior. Para verificar o desempenho do algoritmo neste canal, computamos o NRMSE e a taxa de erro de bit do sinal equalizado, variando-se a SNR na faixa entre 0 e 35 dB. Foram usadas as mesmas quantidades das simulações anteriores. Na Figura 3.8 são mostrados os gráficos de desempenho do algoritmo.

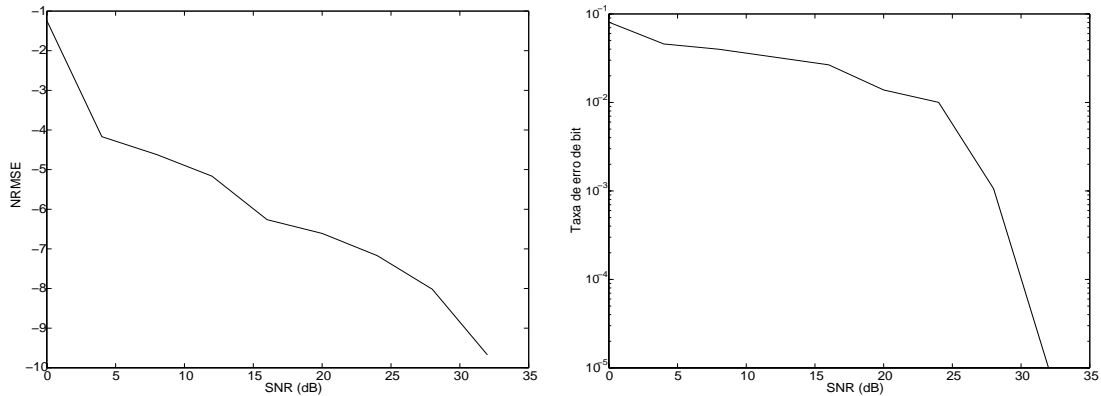


Figura 3.8: Desempenho do algoritmo J-LSS para o canal  $h_3$ .

Tabela 3.1: Tabela taxa de erro versus número de amostras do sinal de entrada.

Número de Amostras	Taxa de Erro
80	0.302
180	0.0248
300	0.0021
500	0.0008

Para avaliar a desempenho do algoritmo em função da variação do número de amostras do sinal de entrada, são apresentados neste exemplo simulações para valores distintos do número de amostras  $N$ . Constata-se que para um número maior de amostras o desempenho melhora bastante. Isso pode ser evidenciado pelos dados apresentados na Tabela 3.1. Os valores foram obtidos com  $\text{SNR} = 20$  dB.

Experimento 4: Em todos os três exemplos anteriores a ordem do preditor,  $w$ , foi escolhida precisamente, ou seja  $w = L + 1$ , de forma a não superestimar a ordem do canal estimado. Por último, apresentamos uma simulação onde a ordem do preditor não foi bem escolhida, sendo bastante superior a ordem real do canal.

Nesta situação, para  $w > L + 1$ , o desempenho do algoritmo cai sensivelmente, evidenciando o problema da detecção da ordem do canal como a principal dificuldade envolvida no processo. Neste caso, o desempenho só é satisfatório para valores elevados da razão sinal-ruído, onde, nestes casos, a detecção da ordem é precisa, vide Figura 3.9.

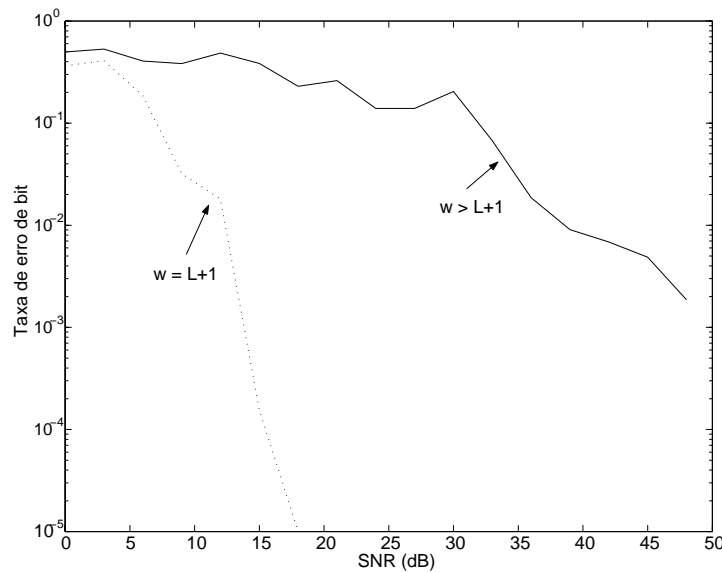


Figura 3.9: Desempenho do algoritmo J-LSS para o canal  $\mathbf{h}_1$  para diferentes valores da ordem de predição.

### 3.7 Conclusão

O algoritmo J-LSS mostrou-se eficiente na equalização de canais com diversidade  $P = 2$ . Para  $P > 2$  o algoritmo apresenta uma sensível melhora no desempenho, porém a um maior custo computacional.

A complexidade computacional do algoritmo é grande, mesmo para pequenas quantidades de amostras do sinal de entrada. Para realizar a identificação do canal  $\mathbf{h}_1$  foram realizadas  $1.08 \times 10^8$  operações de ponto flutuante contra apenas  $3.5 \times 10^4$  em modo supervisionado, ou seja, usando o algoritmo LMS com ordem igual a 12 e mesmo número de amostras na entrada. Isto porque são real-

izadas operações matriciais complexas como inversão e cálculo de autovalores de matrizes de ordem elevada. Isso pode ser considerado um problema em situações onde se deseja que o processo de equalização seja feito em tempo real.

Para uma boa estimativa e conseqüente boa equalização do canal, é preciso estimar bem sua ordem. A grande dificuldade inicial é a escolha da ordem de predição  $w$ . A escolha de  $w < L$  subestimaria a ordem do canal produzindo resultados não tão bons. Por outro lado uma escolha de  $w \gg L$  acaba por superestimar quase sempre a ordem do canal a menos que a SNR seja elevada, tipicamente maior que 40 dB.

O aumento do número de amostras na seqüência de entrada melhora bastante a eficiência do algoritmo, como visto na Tabela 3.1. Isso pode ser constatado pela redução da taxa de erro do sinal equalizado a medida que o número de amostras aumenta. Porém, um maior número de amostras na entrada eleva mais ainda a complexidade computacional, devido ao aumento nas dimensões das matrizes envolvidas.

Os canais cuja matriz de filtragem,  $\mathcal{F}_w(\mathbf{h})$ , é mal condicionada produzem resultados ruins. Isso tem a ver com a proximidade dos zeros dos subcanais que compõem o canal de transmissão. Como mostrado no experimento 2, subcanais com zeros em comum violam a propriedade P3.1, fazendo com que a matriz de filtragem não tenha posto de colunas completo e assim a relação entre os subespaços da entrada e da saída deixa de ser isomórfica, inviabilizando a equalização cega.

## Capítulo 4

# Equalização Cega Adaptativa por Decomposição em Subespaços com Suavização

### 4.1 Introdução

Neste capítulo, é apresentada uma versão adaptativa para o algoritmo J-LSS, o algoritmo A-LSS (*adaptive least squares smoothing*) [8] que se baseia na estrutura dos filtros *lattice* para calcular os erros de projeção e desse modo, determinar unicamente a ordem e a resposta impulsional do canal. Com os filtros *lattice*, os preditores progressivos e regressivos podem ser facilmente implementados em tempo, em ordem e recursivamente. Na seção 4.2, são apresentadas as características que um algoritmo de equalização cega deve satisfazer para operar em tempo real. Na seção 4.3, o modelo do sistema é descrito. Na seção 4.4, suas propriedades gerais mais a implementação do algoritmo são analisadas. Na seção 4.5, são apresentadas simulações de equalização de canal. Por último, na seção 4.6, são apresentadas as conclusões do capítulo.

## 4.2 Características para Operação em Tempo Real

Os atuais sistemas de comunicações digitais, principalmente o serviço móvel celular, dispõem de uma pequena faixa de frequência para a operação. Por isso, seus sinais devem conter o mínimo possível de informação redundante, de modo a aproveitar ao máximo esta pequena quantidade de recursos.

Neste contexto, os algoritmos de equalização cega contribuem em muito ao permitir que o processo de equalização seja realizado sem a necessidade da transmissão de sinais redundantes. Porém, para satisfazer as exigências dos sistemas que operam em tempo real, uma série de requerimentos deve ser satisfeita: O algoritmo de equalização deve ter convergência rápida; o algoritmo deve convergir também com um número finito de amostras; a adaptatividade deve ser rápida para acompanhar as possíveis variações na resposta do canal; e por último baixa complexidade em implementação é sempre o objetivo a ser alcançado.

O algoritmo J-LSS, descrito no capítulo anterior, possui as três primeiras características, porém infelizmente possui um elevado custo computacional, devido principalmente às decomposições matriciais.

Neste capítulo, o algoritmo A-LSS é avaliado. O A-LSS também preserva a propriedade da convergência por número finito de amostras com uma alta taxa de convergência e acomoda uma grande variação na resposta impulsional do canal, tanto variação dos parâmetros quanto a ordem do mesmo. O algoritmo é implementado com módulos básicos usados na literatura clássica dos filtros lattice, desse modo, a estrutura do A-LSS é altamente paralela e regular, usando apenas operações escalares.

### 4.3 Modelo do Sistema

Para descrevermos o algoritmo A-LSS usaremos o modelo de sistema multicanal tipo SIMO descrito pela Figura 2.5 e pelas equações (3.1)-(3.4) e pela matriz de filtragem de Sylvester descrita na equação (3.5).

A meta aqui também é novamente estimar  $\mathbf{h} \triangleq [\mathbf{h}^H(L), \dots, \mathbf{h}^H(0)]^H$  a partir de  $\mathbf{y}(k), k = 1, \dots, N$ .

As suposições S3.1 e S3.2 e as propriedades P3.1 e P3.2, que foram apresentadas para o algoritmo J-LSS, também devem ser satisfeitas aqui para permitir o pleno funcionamento do algoritmo.

### 4.4 Estimação de Canal via Suavização Mínima Quadrática Adaptativa

Nesta seção, será apresentada uma solução adaptativa para o algoritmo J-LSS com ordem desconhecida. A estrutura de dados para uma janela de suavização  $l$  variável, é primeiro especificada. Logo após, o algoritmo para a estimação de canal é apresentado.

#### 4.4.1 Estrutura de Dados e Detecção de Ordem

Em contraste com a estrutura de dados apresentada no capítulo anterior onde o comprimento da janela de suavização é fixo, a estrutura de dados baseada em uma janela de suavização de comprimento variável é agora discutida. Atribuindo um limite superior à ordem do canal,  $L_u$ , para uma ordem de suavização fixa  $w$  e para uma janela de suavização variável  $l$ , definimos a matriz geral de dados como

$$\mathbf{V}_l \triangleq \begin{bmatrix}
 \mathbf{y}(2w + L_u + 1) & \dots & \mathbf{y}(N) \\
 \vdots & \mathbf{Z}_f(l) & \\
 \mathbf{y}(w + l + 1) & & \mathbf{y}(N - w - L_u + l - 1) \\
 \dots & \dots & \dots \\
 \mathbf{y}(w + l) & \dots & \mathbf{y}(N - w - 1) \\
 \vdots & \mathbf{D}(l) & \\
 \mathbf{y}(w + 1) & & \mathbf{y}(N - 2w + 1) \\
 \dots & \dots & \dots \\
 \mathbf{y}(w) & \dots & \mathbf{y}(N - w - l - 1) \\
 \vdots & \mathbf{Z}_p & \\
 \mathbf{y}(1) & & \mathbf{y}(N - 2w - l)
 \end{bmatrix} \quad (4.1)$$

$$\mathbf{Z}(l) \triangleq \begin{bmatrix} \mathbf{Z}_f(l) \\ \mathbf{Z}_p \end{bmatrix} \quad (4.2)$$

$$\mathbf{Z}_c \triangleq \begin{bmatrix} \mathbf{Z}_f(l) \\ \mathbf{D}(l) \end{bmatrix} \quad (4.3)$$

Definindo  $\mathbf{Z}_f(l)$  como matriz de dados do futuro,  $\mathbf{D}(l)$  como matriz de dados do presente e  $\mathbf{Z}_p$  como matriz de dados do passado, temos a matriz de dados futuro-passado  $\mathbf{Z}(l)$  (equação (4.2)) e a matriz de dados futuro-presente  $\mathbf{Z}_c$  (equação (4.3)). Comparadas com as matrizes definidas na equação (3.23), pode-se ver que exceto por  $\mathbf{Z}_p$  e  $\mathbf{Z}_c$ , as matrizes de dados aqui são funções da variável  $l$ . Aqui enfatizamos nesta estrutura particular que  $\mathbf{Z}_p$  e  $\mathbf{Z}_c$  são independentes de  $l$ . Essa propriedade leva a uma implementação prática do algoritmo que será discutida adiante.

Para  $N \geq 4w + 2L_u + L + 1$  e  $s(k)$  tendo complexidade linear maior que  $2w + L_u + L$ , e supondo não haver ruído no sistema, temos as seguintes relações



entre as matrizes definidas em (4.1) e os vários espaços vetoriais:

$$\mathcal{V}_l \triangleq \mathcal{R}\{\mathbf{V}_l\} = \mathcal{X}_{2w+L_u+1, 2w+L_u+1} = \mathcal{S}_{2w+L_u+1, 2w+L_u+L+1} \quad (4.4)$$

$$\begin{aligned} \mathcal{Z}_l \triangleq \mathcal{R}\{\mathbf{Z}(l)\} &= \mathcal{X}_{w,w} \bigcup \mathcal{X}_{w+l+1, -(w+L_u-l+1)} \\ &= \mathcal{S}_{w,w+L} \bigcup \mathcal{S}_{w+l-L+1, -(w+L_u+L-l+1)} \end{aligned} \quad (4.5)$$

Definindo agora  $\mathcal{E}_A^X \triangleq \mathcal{P}_A^\perp\{\mathbf{X}\}$ , como a matriz erro de projeção de  $\mathbf{X}$  em  $\mathcal{R}\{\mathbf{A}\}$ , ver equação (3.24), considere a identificação do canal pela decomposição de pôsto unitário do erro de suavização  $\mathbf{E}(l)$  definido por

$$\mathbf{E}(l) \triangleq \mathcal{E}_{\mathbf{Z}(l)}^{D(l)} \quad (4.6)$$

E, analisando a equação (4.1) temos que

$$\mathcal{V}_l = \mathcal{Z}_l \cup \mathcal{R}\{\mathbf{D}(l)\} = \mathcal{Z}_l + \mathcal{R}\{\mathbf{E}(l)\} \quad (4.7)$$

A condição de pôsto unitário sobre  $\mathbf{E}(l)$  requer que o complemento ortogonal de  $\mathcal{Z}_l$  em  $\mathcal{V}_l$  tenha dimensão igual a 1 [8]. Comparando os subespaços de entrada nas equações (4.4) e (4.5), podemos ver que esse requerimento sobre  $\mathcal{Z}_l$  pode ser atingido somente quando  $l = L + 1$ . Quando  $l < L + 1$ ,  $\mathcal{Z}_l$  se iguala a  $\mathcal{V}_l$ , e não há erro de projeção. Para  $l > L + 1$ , tem-se que  $\mathbf{s}(w + 1), \dots, \mathbf{s}(w + l - L)$  não estão contidos em  $\mathcal{Z}_l$ , logo contribuindo para o erro de suavização [8].

*Teorema 4.1* [8]: Seja  $\mathbf{E}(l) \triangleq \mathcal{E}_{\mathbf{Z}(l)}^{D(l)}$  o erro mínimo quadrático de suavização, com a seqüência de dados especificada em (4.1). Tem-se, na ausência de ruído que

$$\mathbf{E}(l) = \begin{cases} 0, & l < L + 1 \\ \mathbf{h}\tilde{\mathbf{s}}(w + 1|\mathcal{Z}_l), & l = L + 1 \\ \begin{bmatrix} \mathbf{h}(L) & & & \\ \vdots & \ddots & & \\ \mathbf{h}(0) & & \mathbf{h}(L) & \\ & \ddots & \vdots & \\ & & & \mathbf{h}(0) \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{s}}(w + l - L|\mathcal{Z}_l) \\ \vdots \\ \tilde{\mathbf{s}}(w + 1|\mathcal{Z}_l) \end{bmatrix}, & L + 1 < l \leq L_u + 1 \end{cases} \quad (4.8)$$

para prova ver [8].

Considere agora a energia  $\epsilon(l)$  do erro de suavização definido como  $\epsilon(l) = \|\mathbf{E}(l)\|_F^2$ , onde  $\|\bullet\|_F$  denota a norma de Frobenius [1]. O teorema implica que o erro é zero para  $l < L + 1$ . Quando  $l = L + 1$ , o erro salta para um valor relacionado à potência do canal e aumenta assintoticamente com  $l$ .

#### 4.4.2 Suavização Mínima Quadrática Adaptativa

Quando a ordem do canal é desconhecida, existem três componentes envolvidas na estimação de canal no algoritmo A-LSS. Primeiro, os erros de suavização  $\mathbf{E}(l)$  para cada valor da janela de suavização,  $l = 1, \dots, L_u + 1$ , precisam ser calculados. Depois, baseado no teorema 4.1 as energias desses erros precisam ser comparadas a certo limiar para a detecção da ordem do canal. Finalmente, com a ordem estimada  $\hat{L}$ , o canal é obtido a partir de  $\mathbf{E}(\hat{L} + 1)$ .

O principal custo computacional do algoritmo A-LSS vem do cálculo dos erros de suavização  $\mathbf{E}(l)$ , para todo  $l$ . Por isso essa etapa deve ser realizada eficientemente, e para isso é realizada a decomposição do erro em previsões multipasso seguida por projeções lineares.

A implementação em previsões multipassos permite o uso de filtros lattice, que são bastante eficientes computacionalmente. A decomposição em previsões

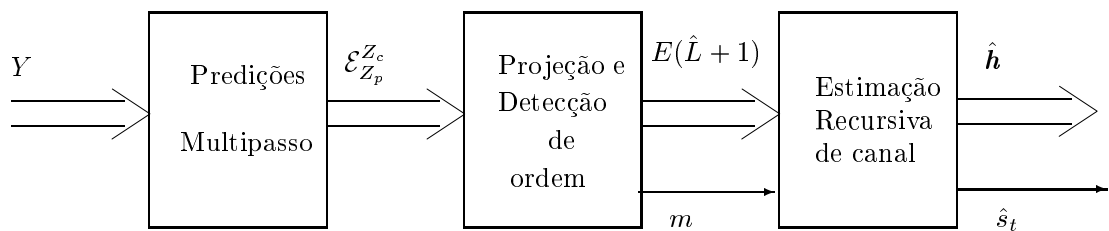


Figura 4.1: Algoritmo A-LSS em blocos.

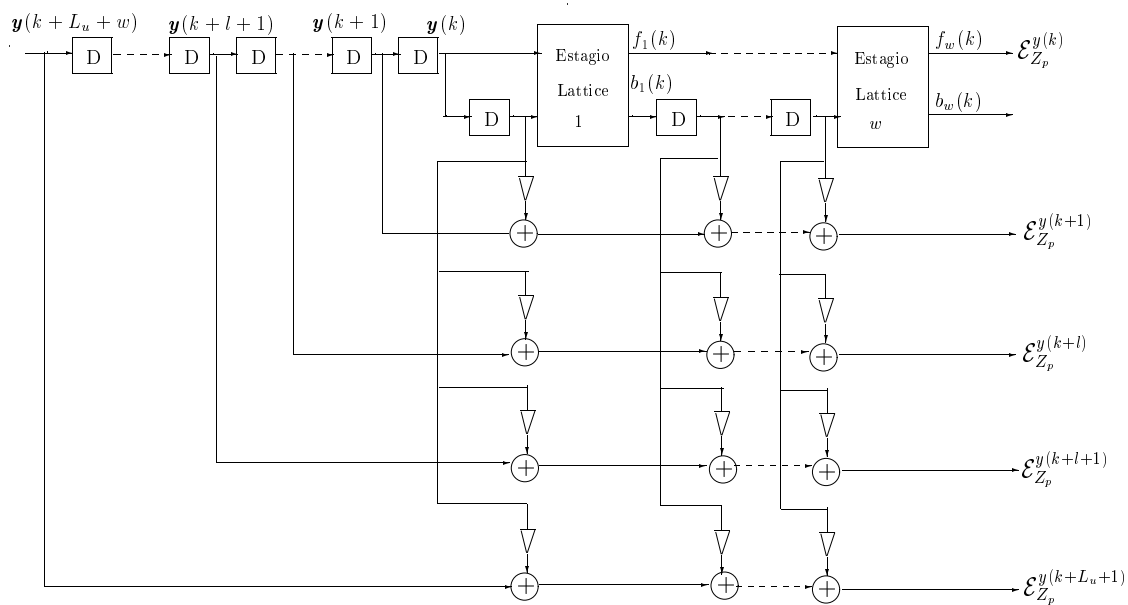


Figura 4.2: Predição multipasso.

multipasso é baseada no seguinte lema:

*Lema 4.1:* Para uma matriz  $\mathbf{U}$  e uma matriz  $\mathbf{Q} = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}$  com  $\mathcal{E}_B^A \neq \mathbf{0}$ , o erro de projeção mínimo quadrático de  $\mathbf{U}$  em  $\mathcal{R}\{\mathbf{Q}\}$  é dado por

$$\mathcal{E}_Q^U = \mathcal{E}_{\mathcal{E}_B^U}^{\mathcal{E}_B^A} \quad (4.9)$$

ou seja, o erro de projeção de  $\mathbf{U}$  em  $\mathcal{R}\{\mathbf{Q}\}$  é igual ao erro de projeção de  $\mathcal{E}_B^U$  em  $\mathcal{R}\{\mathcal{E}_B^A\}$ , onde  $\mathcal{E}_B^U$  e  $\mathcal{E}_B^A$  são os erros de projeção de  $\mathbf{U}$  e  $\mathbf{A}$  em  $\mathcal{R}\{\mathbf{B}\}$ . Para a demonstração ver [8].

Baseada na partição de  $\mathbf{Z}(l) \triangleq \begin{bmatrix} \mathbf{Z}_f(l) \\ \mathbf{Z}_p \end{bmatrix}$  aplica-se o lema 4.1 ao erro de predição  $\mathbf{E}(l)$  dado por

$$\mathbf{E}(l) \triangleq \mathcal{E}_{\mathbf{Z}(l)}^{\mathbf{D}(l)} = \mathcal{E}_{\begin{matrix} \mathcal{E}_{\mathbf{Z}_p}^{\mathbf{D}(l)} \\ \mathcal{E}_{\mathbf{Z}_f}^{\mathbf{Z}_p(l)} \end{matrix}} = \mathcal{E}_{\tilde{\mathbf{Z}}_f(l)}^{\tilde{\mathbf{D}}(l)} \quad (4.10)$$

onde  $\tilde{\mathbf{D}}(l)$  e  $\tilde{\mathbf{Z}}_f(l)$  são os erros de predição multipasso de  $\mathbf{D}(l)$  e  $\mathbf{Z}_f(l)$  respectivamente pela matriz de dados do passado  $\mathbf{Z}_p$ .

A estrutura baseada na decomposição em multipasso permite que o algoritmo seja implementado em três etapas, como na Figura 4.1. O preditor multipasso calcula os erros  $\mathcal{E}_{\mathbf{Z}_p}^{\mathbf{Z}_c}$ . No segundo passo, todos os erros de suavização para todos os valores de  $l$ , são obtidos recursivamente, e a ordem do canal é estimada. E, finalmente, com a ordem estimada  $\hat{L}$ , o canal é estimado a partir do erro de suavização  $\mathbf{E}(\hat{L} + 1)$  dado pela equação (4.8) mas que também é obtido adaptativamente.

Vários algoritmos adaptativos para filtros lattice podem ser usados para calcular  $\mathbf{E}(l)$  [8]. A propriedade da recursão em ordem dos filtros lattice pode ser muito útil no caso da ordem do canal ser desconhecida, devido à sua simplicidade computacional. Quando  $L$  é desconhecida, a ordem de predição tem de ser determinada baseada em  $L_u$ . Por exemplo, quando  $P = 2$  escolhemos  $w = L_u$ . Se  $L_u$  é um limite ruim para a ordem do canal, devemos escolher um  $w$  maior o que elevaria o custo computacional do sistema.

A partir dos erros de predição  $\mathcal{E}_{Z_p}^{Z_c}$ , os erros de suavização  $\mathbf{E}(l)$  para cada valor da janela de suavização  $l = 1, \dots, L_u + 1$  são obtidos para a detecção de ordem e a estimação de canal. Uma aproximação para se obter  $\mathbf{E}(l)$  recursivamente é aplicando o algoritmo recursivo modificado de Gram-Schmidt [8]. Uma vez que todos os  $\mathbf{E}(l)$  são obtidos, a ordem do canal é detectada por um detector de energia.

O último passo é a estimação da resposta impulsional do canal que é realizada recursivamente a partir dos erros de suavização  $\mathbf{E}(\hat{L} + 1)$ . Para detalhes deste processo ver [8].

## 4.5 Resultados Experimentais

O algoritmo A-LSS foi implementado também no ambiente matlab e verificou que seu desempenho foi inferior ao algoritmo J-LSS, em termos da taxa de erro do sinal equalizado, para a maioria dos canais testados. As figuras de mérito para desempenho foram o NRMSE, definido pela equação (3.34), e a taxa de erro para o sinal equalizado. Foram usadas neste caso  $N = 500$  amostras no sinal de entrada. A taxa de erro de bits, foi determinada a partir da equalização do canal estimado, usando uma seqüência de entrada de 1000 bits modulados em BPSK. Foram realizados 100 experimentos, para valores inteiros da razão sinal-ruído na faixa especificada, totalizando 100.000 bits transmitidos. O resultado final, por SNR, foi obtido tirando-se a média dos valores encontrados, tanto para o NRMSE quanto para a taxa de erro.

Experimento 1: Primeiramente, realizamos a estimação do canal  $\mathbf{h}_1$  dado pela equação (3.36). Como descrito no capítulo anterior, este canal possui a matriz de filtragem bem condicionada com número de condicionamento próximo de 100. Para verificar o desempenho do algoritmo, calculamos o NRMSE e também a taxa de erro de símbolos na recepção. As medidas foram realizadas variando-se

a razão sinal-ruído na faixa entre 0 e 20 dB. Na Figura 4.3 são mostrados os gráficos de desempenho do algoritmo para o canal dado na equação  $\mathbf{h}_1$ .

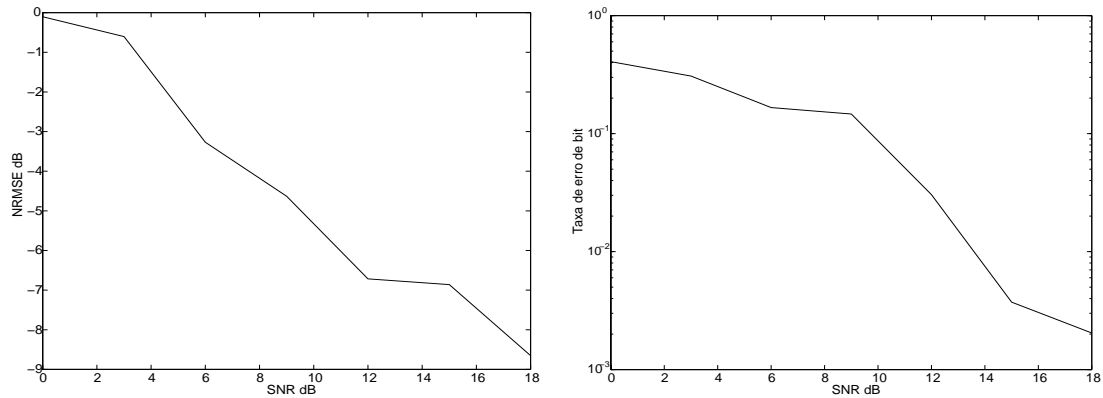


Figura 4.3: Desempenho do algoritmo A-LSS para o canal  $\mathbf{h}_1$ .

Para este canal constata-se que algoritmo teve o desempenho inferior ao J-LSS no critério da taxa de erro. Assim como o J-LSS, o A-LSS é um método de identificação de canais então, para perfazer a equalização do canal é necessário inverter a resposta do canal encontrado pelo algoritmo. Vemos que a taxa de erro apenas alcança valores satisfatórios onde a SNR é maior que 20 dB.

Experimento 2: Agora o algoritmo A-LSS é usado para equalizar, o canal  $\mathbf{h}_3$  dado pela equação (3.38) cuja resposta impulsional produz uma matriz de filtragem mal condicionada. Para verificar o desempenho do algoritmo na equalização deste canal, computamos o NRMSE e a taxa de erro de bit do sinal equalizado, agora variando-se a SNR na faixa entre 0 e 25 dB. Na Figura 4.4 são mostrados os gráficos de desempenho do algoritmo.

Neste exemplo constata-se que o desempenho do algoritmo foi superior ao do algoritmo J-LSS produzindo uma taxa de erro menor especialmente na faixa de 10 a 25 dB. Isto devido, fundamentalmente, à capacidade de ortogonalização dos filtros lattice. Quanto ao NRMSE constatamos também que o algoritmo tem boa

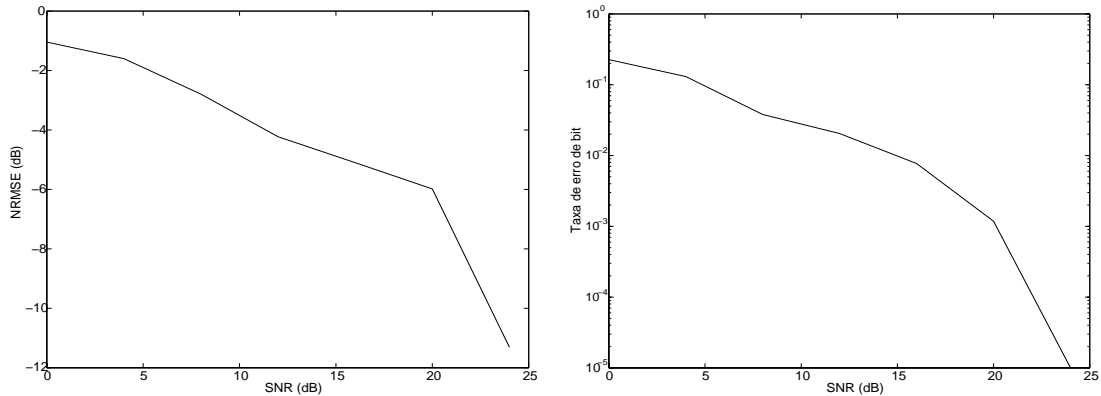


Figura 4.4: Desempenho do algoritmo A-LSS para o canal  $\mathbf{h}_3$ .

performance em vista da sua redução conforme o aumento da razão sinal-ruído.

Este experimento deveria apresentar um desempenho, em relação á taxa de erro, pior que o experimento 1. O que aconteceu, na realidade, foram muitos erro de inversão de canal no primeiro experimento, que não ocorreram com freqüência no experimento 2.

## 4.6 Conclusão

O algoritmo A-LSS mostrou-se eficiente na equalização de canais onde a SNR era alta, geralmente maior que 20 dB. Com o A-LSS, o processo de detecção da ordem do canal ainda é mais complexo e por ser assim é mais susceptível a erros. Para uma detecção precisa da ordem é preciso ter um número de iterações elevado, da ordem 2000 tipicamente, e além disso, não sobrestimar em muito o valor superior da ordem do canal  $L_u$ . Quando maior  $L_u$  maior será a dificuldade de se obter a ordem correta do canal.

A complexidade computacional do algoritmo é bem menor que no J-LSS. Para realizar a identificação do canal  $\mathbf{h}_1$  foram realizadas  $2.45 \times 10^5$  operações de ponto flutuante contra  $1.08 \times 10^8$  do J-LSS. Porém o A-LSS ainda requer mais esforço computacional que os métodos supervisionados, onde foram realizadas

apenas  $3.5 \times 10^4$  operações de ponto flutuante usando o algoritmo LMS com ordem igual a 12 e mesmo número de amostras na entrada. Toda a complexidade das operações matriciais realizadas no algoritmo J-LSS aqui são suprimidas o que torna o A-LSS bem cotado para processos de equalização em tempo real.

Assim como no J-LSS, os canais cuja matriz de filtragem,  $\mathcal{F}_w(\mathbf{h})$ , é mal condicionada produzem resultados ruins, porém melhores que no algoritmo J-LSS. Isso tem a ver com a capacidade de ortogonalização dos filtros lattice.



## Capítulo 5

# Algoritmo de Equalização Cega de Predição de Erro

### 5.1 Introdução

Os algoritmos de predição de erro (*prediction error*, PE) [9] oferecem uma alternativa para o problema da equalização envolvendo métodos cegos. Esses métodos exploram o fato da seqüência de entrada do sistema ser do tipo iid, e utiliza um filtro de predição de erro mínimo na recepção da seqüência transmitida.

As principais vantagens do algoritmo PE sobre os algoritmos avaliados nos capítulos anteriores são: a realização da equalização direta de canal usando a topologia em cascata; a não necessidade de se estimar a ordem do canal, que garante a robustez da técnica com respeito ao difícil problema de estimação da ordem; e por último a implementação adaptativa de custo muito baixo.

Sua principal desvantagem é grande sensibilidade ao ruído presente no canal. O ruído induz interferência entre símbolos residual na saída do filtro preditor, causando inconsistência na equalização do canal. Por essa razão, um procedimento de eliminação de ruído deve ser aplicado para remover a polarização do preditor.

Na seção 5.2, são apresentados o modelo de sistema e as características que um algoritmo deve satisfazer para produzir a equalização. Na seção 5.3, o método de predição de erro, no caso mais geral, é descrito. Na seção 5.4, o método de predição de erro não polarizado é apresentado. Na seção 5.5, a implementação do algoritmo é discutida. Na seção 5.6, são apresentadas simulações envolvendo o algoritmo. Na seção 5.7, são apresentadas as conclusões do capítulo.

## 5.2 Modelo do Sistema

Para descrevermos o algoritmo PE usaremos o modelo de sistema multicanal tipo SIMO descrito pela Figura 2.5 e pelas equações (3.1)-(3.4) e pela matriz de filtragem de Sylvester (equação (3.5)).

Assume-se que a ordem de predição  $w$  é grande o bastante, e que exista uma diversidade de canal suficiente tal que as seguintes suposições sejam estabelecidas:

- S5.1)  $wP > L + w$ , isto é,  $\mathcal{F}_w(\mathbf{h})$  tem mais linhas do que colunas, onde  $P$  é o número de canais,  $L$  é a ordem do canal e  $w$  é a ordem do preditor.
- S5.2)  $\mathcal{F}_w(\mathbf{h})$  tem posto de colunas completo, e igual a  $L + w$ .
- S5.3) A seqüência de entrada  $s(k)$  é branca, isto é, é composta por símbolos iid, com média zero e variância unitária.

## 5.3 Método de Predição de Erro para a Equalização Cega

A idéia dos métodos de predição de erro é encontrar a resposta inversa do canal de transmissão, usando a suposição S5.3 como restrição. O algoritmo de PE é motivado pelo seguinte lema [9]:

*Lema 5.1:* Seja  $\mathbf{p}$  o vetor com os coeficientes do filtro de predição de dimensão  $wP \times 1$ . Seja  $e^{[i]}(k)$  o erro de predição definido por

$$e^{[i]}(k) = y^{[i]}(k) - \mathbf{p}^H \mathbf{y}_w(k-1) \quad i = 1, \dots, P \quad (5.1)$$

Para uma transmissão sem ruído e considerando as suposições S5.1-S5.3, a variância do erro de predição dada por

$$J(\mathbf{p}) = E[e^{[i]}(k)]^2 \quad (5.2)$$

tem valor mínimo se e somente se

$$e^{[i]}(k) = [\underline{\mathbf{1}}, -\mathbf{p}^H] \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}_w(k-1) \end{bmatrix} = h^{[i]}(0)s(k) \quad (5.3)$$

onde  $\underline{\mathbf{1}} = (0, \dots, 0, 1, 0, \dots, 0)$  tem um elemento 1 na  $i$ -ésima posição. Diferentemente dos métodos baseados em decomposição de subespaços, a ordem  $L$  dos canais não precisa ser conhecida aqui. De fato, apenas a ordem de predição  $w$  precisa ser determinada de forma que a suposição S5.1, seja estabelecida.

Dessa forma, um novo método para equalização cega de canal é avaliado, o qual consiste em encontrar os coeficientes do filtro preditor correspondendo a minimização de  $J(\mathbf{p})$  e usando os coeficientes do filtro  $[\underline{\mathbf{1}}, -\mathbf{p}^H]$  como resposta do equalizador.

### 5.3.1 Método de Predição de Erro com Ruído Aditivo

A maior limitação do método de predição de erro para equalização cega está na presença de ruído aditivo. Na presença de ruído, a problema da minimização de  $J(\mathbf{p})$  motiva uma modificação nas equações de Yule-Walker[2]

$$\mathbf{R}_y \mathbf{p} = \mathbf{a} \quad (5.4)$$

com

$$\mathbf{R}_y = E[\mathbf{y}_w(k-1)\mathbf{y}_w^H(k-1)] \quad (5.5)$$

$$\mathbf{a} = E[\mathbf{y}_w(k-1)y^{[i]}(k)^*] \quad (5.6)$$

de forma que podemos separar  $\mathbf{R}_y$  e  $\mathbf{a}$  nas seguintes partes:

$$\mathbf{R}_y = \mathbf{R}_{sig} + \sigma^2 \mathbf{R}_n \quad (5.7)$$

$$\mathbf{a} = \mathbf{a}_{sig} + \sigma^2 \mathbf{a}_n \quad (5.8)$$

onde  $\mathbf{R}_{sig}$  indica a contribuição apenas do sinal na matriz de correlação,  $\mathbf{a}_{sig}$  indica a contribuição apenas do sinal no vetor de correlação.  $\mathbf{R}_n$  denota a matriz de correlação do ruído e  $\sigma^2$  a desconhecida variância do ruído. O vetor de correlação, de ruído  $\mathbf{a}_n$  é definido de maneira similar. Quando o ruído presente no sistema é branco, temos que

$$\mathbf{R}_n = E[\mathbf{n}_w(k)\mathbf{n}_w^H(k)] = \mathbf{I} \quad (5.9)$$

$$\mathbf{a}_n = (\sigma^2)^{-1} E[\mathbf{n}_w(k-1)n^{[i]}(k)^*] = \mathbf{0}$$

Seja  $\mathbf{p}_0$  um vetor que denota os coeficientes de um preditor livre de ruído definido por

$$(\mathbf{R}_y - \sigma^2 \mathbf{I})\mathbf{p}_0 = \mathbf{a} \quad (5.10)$$

Assim, o ruído introduz uma polarização em  $\mathbf{p}_0$  na forma de

$$\delta \mathbf{p} = \mathbf{p} - \mathbf{p}_0 = -\sigma^2 \mathbf{R}_y^{-1} \mathbf{p}_0 \quad (5.11)$$

Neste caso, o preditor polarizado  $[\underline{1}, -\mathbf{p}^H]$  não é mais a resposta inversa desejável para o canal. Isso causará inconsistência durante a estimação do canal o que torna necessário realizar uma remoção de ruído da seqüência  $y(k)$ , ou melhor, da matriz de correlação  $\mathbf{R}_y$ . Esse problema é resolvido tradicionalmente através da estimativa de  $\sigma^2$  e depois subtraindo a porção ruidosa da matriz  $\mathbf{R}_y$ . Dessa forma, o sistema estará trabalhando, em teoria, com estatísticas livre de ruído.

O problema é que a estimação do nível de ruído não é muito acurada e também o procedimento não tem em si uma implementação adaptativa fácil de ser realizada. No método PE, é analisado um novo esquema de predição permitindo a solução adaptativa de  $\mathbf{p}_0$  em um ambiente ruidoso.

## 5.4 Predição Linear Adaptativa Não Polarizada

Preditores não polarizados são definidos para vários tipos de ruído, como sendo a solução da equação de Yule-Walker na ausência de ruído

$$(\mathbf{R}_y - \sigma^2 \mathbf{R}_n) \mathbf{p}_0 = \mathbf{a} - \sigma^2 \mathbf{a}_n \quad (5.12)$$

O foco do algoritmo de PE está no problema da estimação adaptativa de  $\mathbf{p}$  com o conhecimento da matriz  $\mathbf{R}_n$ , porém sem o conhecimento de  $\sigma^2$ . Em [10], há um problema similar, onde propõe-se um algoritmo adaptativo denominado  $\gamma$ -LMS que tenta explicitamente remover a contribuição do ruído a partir das estatísticas instantâneas do erro de predição.

### 5.4.1 Predição Restrita

O critério de predição de erro  $J(\mathbf{p})$  no lema 5.1 consiste no caso do erro ser particionado na parte do sinal e na parte do ruído

$$\begin{cases} J(\mathbf{p}) = J_{sig}(\mathbf{p}) + \sigma^2 J_n(\mathbf{p}) \\ J_{sig}(\mathbf{p}) = \mathbf{R}_{sig}(i, i) + \mathbf{p}^H \mathbf{R}_{sig} \mathbf{p} - 2\Re[\mathbf{p}^H \mathbf{a}_{sig}] \\ J_n(\mathbf{p}) = 1 + \mathbf{p}^H \mathbf{R}_n \mathbf{p} - 2\Re[\mathbf{p}^H \mathbf{a}_n] \end{cases} \quad (5.13)$$

onde  $\Re[x]$  denota a parte real de  $x$ .

Vamos minimizar  $J(\mathbf{p})$  sujeito a restrição que  $J_n(\mathbf{p}) = \gamma^2$ , onde  $\gamma$  é um número real. Usando a técnica de multiplicadores de Lagrange, o problema resulta em um conjunto de pontos estacionários caracterizados por

$$\frac{\partial J(\mathbf{p})}{\partial \mathbf{p}} + \mu \frac{\partial J_n(\mathbf{p})}{\partial \mathbf{p}} = 0 \quad (5.14)$$

onde  $\mu$  é um multiplicador de Lagrange real cujo valor é determinado pela escolha da restrição. A idéia aqui é que os pontos estacionários da equação (5.14) são também aqueles obtidos no problema de predição não restrita com uma nova variância de ruído dada por  $(\sigma^2 + \mu)$ . Claro que estamos interessados em selecionar o valor da restrição  $\gamma$  para obter  $\mu = -\sigma^2$  de modo a alcançar a desejada remoção de ruído. A análise aqui é feita para o caso onde o ruído é do tipo AWGN, por simplicidade, mas esta mesma análise pode ser estendida para outros tipos de ruído [9].

### 5.4.2 Remoção de Ruído AWGN

Para o caso do ruído ser branco, a restrição se reduz a

$$J_n(\mathbf{p}) = \gamma^2 = 1 + \|\mathbf{p}\|^2 \quad (5.15)$$

Sem perda de generalidade, pode-se redefinir a restrição como  $\|\mathbf{p}\| = \gamma$  (restrição de norma), o que traz uma conexão entre a variância do ruído e a norma dos preditores.

No caso do ruído ser branco, os preditores não polarizados são definidos por

$$(\mathbf{R}_y - \sigma^2 \mathbf{I})\mathbf{p} = \mathbf{R}_{sig}\mathbf{p} = \mathbf{a}_{sig} \quad (5.16)$$

e ainda a matriz de correlação para o caso de não haver ruído é dada por

$$\mathbf{R}_{sig} = \mathcal{F}_w(\mathbf{h})\mathcal{F}_w^H(\mathbf{h}) \quad (5.17)$$

que sob a suposição S5.1 torna a matriz  $\mathbf{R}_{sig}$  uma matriz de posto não completo. Por essa razão, as soluções para o sistema de equações Yule-Walker da equação (5.16), na ausência de ruído, pertencem a um conjunto de infinitos termos. A singularidade do sistema de equações de Yule-Walker, na ausência de ruído, constitui uma propriedade notável. A idéia principal por trás do método de remoção

de ruído consiste em restaurar essa propriedade, no caso ruidoso, atuando sobre a norma do preditor.

As soluções para a equação (5.16) geram um subespaço que pode ser parametrizado por

$$\begin{aligned} \mathbf{p} &= \mathbf{p}_0 + \mathbf{q} & (5.18) \\ \mathbf{p}_0 &= (\mathbf{R}_{sig}^H \mathbf{R}_{sig})^{-1} \mathbf{R}_{sig}^H \mathbf{a} \\ \mathbf{q} &\in \text{núcleo}(\mathbf{R}_{sig}) \end{aligned}$$

onde  $\mathbf{p}_0$  denota um preditor particular não polarizado que é caracterizado pelo fato de possuir a menor norma para toda a classe de preditores não polarizados [9]. Seja  $\gamma_0$  a norma desse preditor

$$\gamma_0 = \|\mathbf{p}_0\| = \|(\mathbf{R}_{sig}^H \mathbf{R}_{sig})^{-1} \mathbf{R}_{sig}^H \mathbf{a}\| \quad (5.19)$$

Por construção de  $\gamma_0$ , todos os preditores não polarizados  $\mathbf{p}$  são tais que  $\|\mathbf{p}\| \geq \gamma_0$ . Acrescentando, para qualquer valor positivo de  $\gamma$  tal que  $\gamma_0 \leq \gamma \leq \infty$ , existe um outro preditor não polarizado  $\mathbf{p}$  tal que  $\gamma = \|\mathbf{p}\|$ . Em outras palavras, um preditor é não polarizado se e somente se sua norma é limitada inferiormente mas não superiormente devido à deficiência do posto da matriz  $\mathbf{R}_{sig}$  [9].

## 5.5 Algoritmo de Predição de Erro

Baseado no que foi dito nas seções anteriores, temos o seguinte critério para encontrar o preditor não polarizado:

$$\begin{cases} \text{minimizar } J(\mathbf{p}) = E[y^{[i]}(k) - \mathbf{p}^H \mathbf{y}_w(k-1)]^2 \\ \text{sujeito a } \|\mathbf{p}\| = \gamma \end{cases}$$

### 5.5.1 Implementação do Algoritmo

A implementação adaptativa de filtros restritos quadraticamente tem por solução direta a técnica de projeção de gradiente, usada extensivamente em toda

a literatura [11]. O algoritmo, que tem baixo custo computacional e mostra diversas propriedades de robustez, consiste em uma clássica forma do algoritmo LMS, onde a expressão para o gradiente de  $J(\mathbf{p})$ , seguido pela projeção e sobre a restrição de  $\|\mathbf{p}\| = \gamma$ . O algoritmo segue abaixo [9]:

para cada  $k$  computar

$$\begin{aligned} e^{[i]}(k) &= y^{[i]}(k) - \mathbf{p}(k-1)^H \mathbf{y}_w(k-1) \\ \mathbf{u}^{[i]}(k) &= \mathbf{y}_w(k-1) e^{[i]}(k)^* \\ \tilde{\mathbf{p}}(k) &= \mathbf{p}(k-1) + \alpha \mathbf{u}^{[i]}(k) \\ \mathbf{p}(k) &= \frac{\tilde{\mathbf{p}}(k) \gamma}{\|\tilde{\mathbf{p}}(k)\|} \\ \mathbf{r}(k) &= \lambda \mathbf{r}(k-1) + (1-\lambda) \mathbf{u}^{[i]}(k) \\ \sigma^2(k) &= \|\mathbf{r}(k)\|/\gamma \end{aligned}$$

onde  $\lambda$  é o fator de esquecimento ( $0 < \lambda < 1$ ).

A escolha do valor de  $\gamma \geq \gamma_0$  tem influência sobre o comportamento do erro médio quadrático do algoritmo. Por isso é bom não superestimar muito  $\gamma_0$ . Um aumento de  $\gamma$  dispara o efeito de remoção do ruído. Uma solução prática para realizar a estimativa de  $\gamma$  seria após cada transmissão e equalização sucessiva, incrementar  $\gamma$  a partir do valor de  $\gamma_0$  e escolher o valor que produzisse a menor taxa de erro. Esse critério porém requer o conhecimento de, pelo menos, parte da seqüência de entrada transmitida o que tornaria o sistema semicego.

## 5.6 Resultados Experimentais

O algoritmo foi implementado no ambiente matlab e diferentemente dos algoritmos J-LSS e A-LSS, o algoritmo PE realiza a equalização direta e não a identificação de canal. Isso é vantajoso para sistemas que operam em tempo real pois, desse modo, um menor número de operações é realizado aumentando a eficiência do algoritmo. As figuras de mérito para o algoritmo PE são o erro



médio quadrático definido por

$$\text{MSE} \triangleq E [e^H(k)e(k)] \quad (5.20)$$

onde  $e(k)$  foi definido na equação (5.3), e a taxa de erro como critério qualitativo para a equalização de canal. Nos vários experimentos procuramos variar os parâmetros do algoritmo em busca sempre do melhor desempenho.

Experimento 1: Primeiramente, realizamos a equalização do canal  $\mathbf{h}_1$  da equação (3.36). Este canal possui a matriz de filtragem bem condicionada. Para verificar o desempenho do algoritmo calculamos a taxa de erro de símbolos do sinal equalizado na recepção. As medidas foram realizadas variando-se a razão sinal-ruído na faixa entre 0 e 18 dB. Na Figura 5.1 é mostrado o gráfico da taxa de erro do algoritmo. Foram usadas as mesmas quantidades das simulações anteriores. A ordem do preditor foi igual a  $w = 18$ , o número de iterações foi igual a 2000 e o valor de  $\gamma$  foi igual a 0.75.

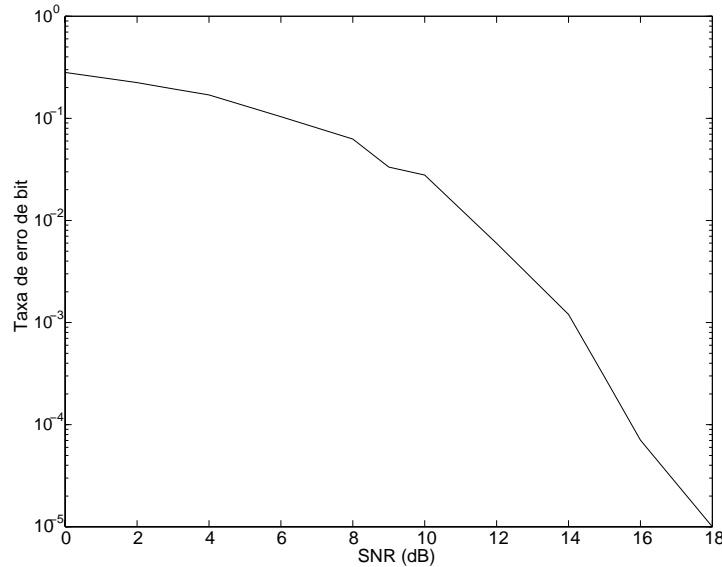


Figura 5.1: Desempenho do algoritmo PE para o canal  $\mathbf{h}_1$  critério taxa de erro.

Para este canal constata-se também o bom funcionamento do algoritmo

pois vemos que quando a razão sinal-ruído atinge o valor de 16 dB, a taxa de erro é menor que  $10^{-4}$ , valor considerado bom para equalização de sistemas em tempo real. Conforme pode ser observado comparando as Figuras 5.1 e 3.4, este resultado ainda é melhor que o obtido pelo algoritmo J-LSS e isso a um custo computacional bem menor.

Experimento 2: Agora realizamos a equalização do canal  $\mathbf{h}_2$  da equação (3.37). Como visto, o canal  $\mathbf{h}_2$  possui uma resposta impulsional que produz uma matriz de filtragem que pode ser considerada mal condicionada, induzindo a resultados ruins. Aqui, na Figura 5.2 é apresentada a taxa de erro de bit do sinal equalizado para dois valores distintos de  $\gamma$ . A ordem do preditor foi  $w = 18$  e o número de iterações foi igual a 2000.

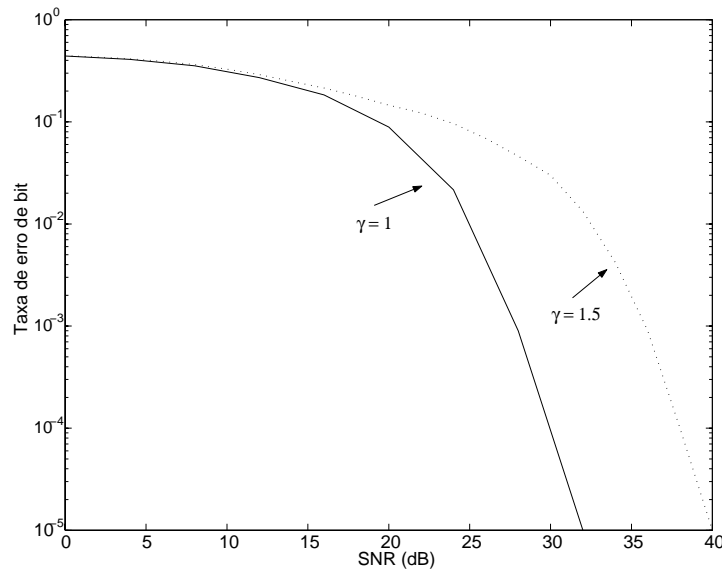


Figura 5.2: Desempenho do algoritmo PE para o canal  $\mathbf{h}_2$  critério taxa de erro.

Pode-se constatar o efeito de retirada de ruído para diferentes valores de  $\gamma$ . Para  $\gamma = 1$  a curva da taxa de erro de bit do sinal equalizado é melhor indicando que há menos ruído presente no sistema. Também nesse caso foi obtida um desempenho superior aos algoritmos J-LSS e A-LSS.

Experimento 3: Agora realizamos a equalização do canal  $\mathbf{h}_3$  da equação (3.38). A ordem do preditor foi igual a 18 e o valor de  $\gamma$  foi igual a 0.75. Os gráficos de desempenho do erro médio quadrático, da taxa de erro e também da variância retirada do preditor,  $\sigma^2$ , pelo número de iterações são apresentados nas Figuras 5.4, 5.3 e 5.5, para SNR = 20 dB.

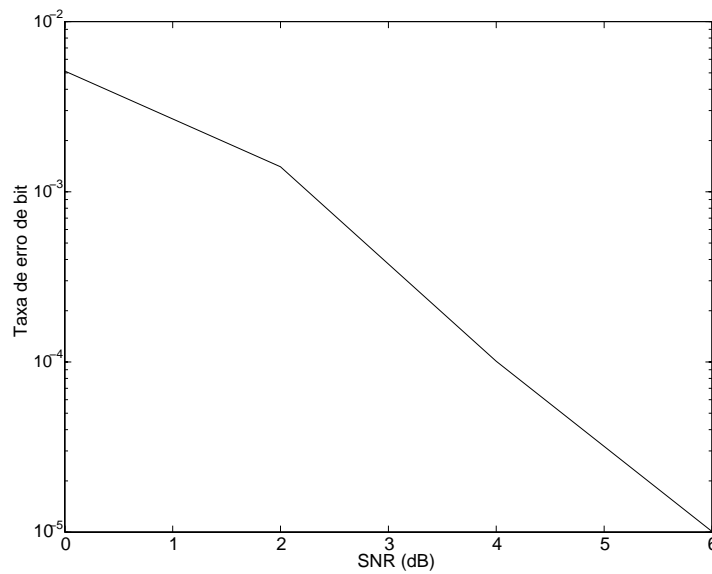


Figura 5.3: Desempenho, em termos da taxa de erro, do algoritmo PE para o canal  $\mathbf{h}_3$ .

A variância do ruído usado nas simulações foi de 0.004. Logo vemos que boa parte do ruído presente foi suprimida.

Experimento 4: Neste último exemplo o algoritmo PE é usado para equalizar um canal complexo com desvanecimento também de fase. O resultado é apresentado para dois valores distintos da ordem de predição, cujo o objetivo é obter uma comparação a partir da variação da mesma. Usamos  $\gamma = 1$ .

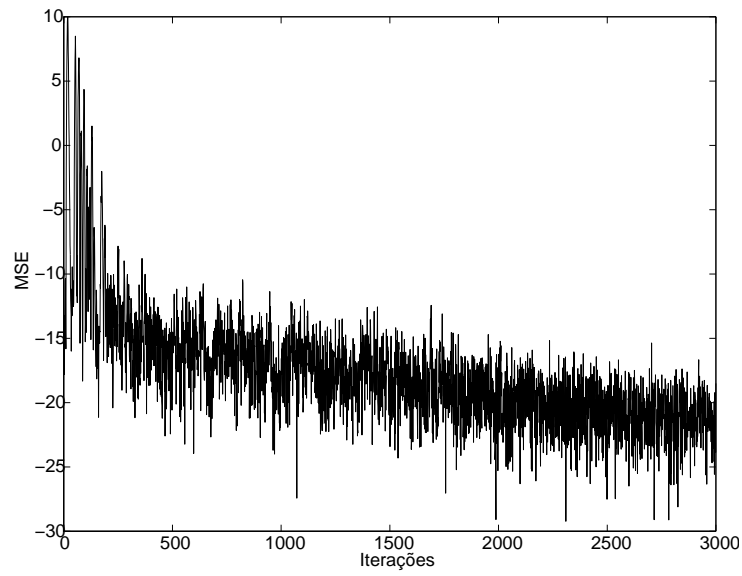


Figura 5.4: Desempenho, em termos do MSE, do algoritmo PE para o canal  $h_3$ .

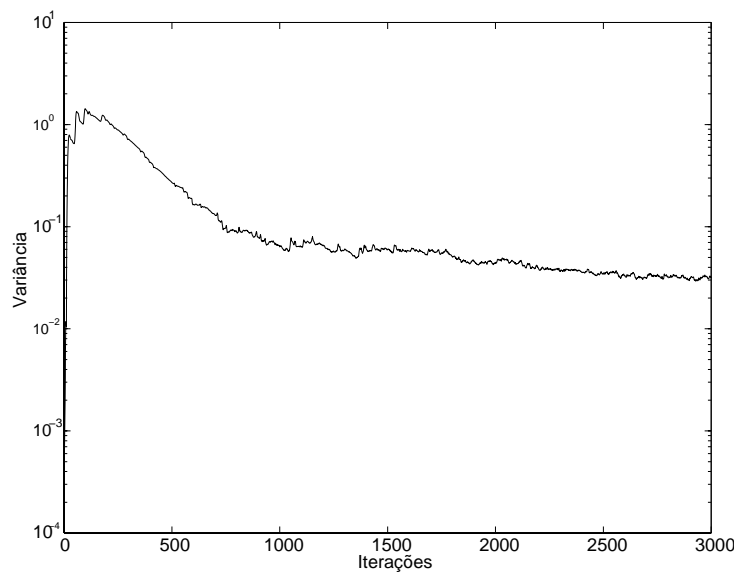


Figura 5.5: Estimativa de variância do ruído retirada do preditor no experimento

3.

$$\mathbf{h}_4 = \begin{bmatrix} 0.000 + 0.000i & 0.000 + 0.000i \\ 0.129 + 0.032i & 0.171 + 0.043i \\ 0.324 + 0.081i & 0.386 + 0.096i \\ 0.556 + 0.139i & 0.613 + 0.153i \\ 0.779 + 0.194i & 0.813 + 0.203i \\ 0.941 + 0.235i & 0.951 + 0.237i \\ 1.000 + 0.250i & 1.000 + 0.250i \\ 0.941 + 0.235i & 0.951 + 0.237i \\ 0.779 + 0.195i & 0.813 + 0.203i \\ 0.556 + 0.139i & 0.613 + 0.153i \\ 0.324 + 0.081i & 0.386 + 0.096i \\ 0.129 + 0.032i & 0.171 + 0.043i \\ 0.000 + 0.000i & 0.000 + 0.000i \end{bmatrix}$$

O gráfico da taxa de erro é exibido na Figura (5.6).

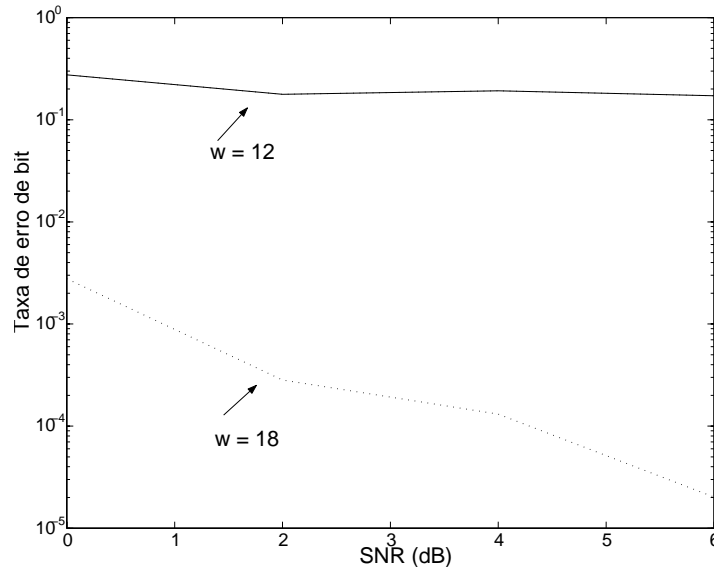


Figura 5.6: Desempenho do algoritmo PE para o canal  $\mathbf{h}_4$  segundo o critério taxa de erro.

No caso onde  $w = 12$  temos pela suposição S5.1 não é satisfeita, inviabilizando a equalização cega. Porém, no caso onde  $w = 18$  temos que S5.1 é satisfeita.

## 5.7 Conclusão

O algoritmo PE mostrou-se, até aqui, o mais eficiente na equalização de canais com diversidade e isso associado a um menor custo computacional. De fato, a complexidade computacional do algoritmo é a segunda menor atingida até aqui, perdendo apenas para o algoritmo Bussgang. Para realizar a identificação do canal  $\mathbf{h}_1$  foram realizadas  $1.05 \times 10^5$  operações de ponto flutuante contra  $1.08 \times 10^8$  do algoritmo J-LSS,  $2.45 \times 10^5$  do algoritmo A-LSS, chegando perto do algoritmo LMS com  $3.4 \times 10^4$  operações.

Para uma boa equalização do canal é preciso estimar bem seus parâmetros de entrada como a ordem dos preditores  $w$  e o valor da constante  $\gamma$ . A grande dificuldade inicial é a determinação do valor de  $\gamma_0$  que como vimos tem efeitos diferentes durante a equalização. Valores não ajustados deste parâmetro tornam o sistema mais susceptível ao ruído dificultando a equalização, isso para 500 amostras processadas.

Igualmente aos algoritmos J-LSS e A-LSS previamente implementados, os canais cuja matriz de filtragem,  $\mathcal{F}_w(\mathbf{h})$ , é mal condicionada produzem resultados ruins. Porém, devido a não necessidade de se estimar a ordem do canal e também por fazer equalização direta, o algoritmo de predição de erro torna-se vantajoso em relação os métodos baseados na decomposição se subespaços, como os algoritmos J-LSS e A-LSS.

## Capítulo 6

# Detecção Multiusuário Cega por Decomposição de Subspaços

### 6.1 Introdução

Sistemas de acesso múltiplo de divisão por código, no inglês CDMA (*code division multiple access*), que usam espalhamento espectral por seqüência direta, são hoje responsáveis por uma grande mudança no mundo dos serviços de comunicações como serviço móvel celular, sistemas de comunicações pessoais e comunicações via satélite. As técnicas de detecção multiusuário aumentam substancialmente a capacidade desses sistemas. Por isso, atualmente, muita pesquisa tem sido realizada na detecção multiusuário adaptativa. Como exemplo, têm-se os métodos para se obter soluções adaptativas para detectores de decorrelação e detectores lineares do tipo MMSE (*minimum mean square error*) que usam as seqüências de espalhamento no processo de adaptação [12, 13].

Nesta capítulo, é apresentado um algoritmo proposto por Wang e Poor [14] que consiste em um novo detector multiusuário cego baseado na estimação do subespaço vetorial do sinal recebido. Neste trabalho, demonstra-se que através da estimação do subespaço do sinal recebido, podem-se obter o detector de de-

scorrelação e o detector MMSE sem o uso da seqüência de entrada, ou seja, estes detectores podem ser obtidos por métodos cegos usando para tal a seqüência de espalhamento do usuário desejado.

Outra contribuição importante do trabalho é o desenvolvimento de um método adaptativo cego para estimar a resposta do canal de transmissão. Sobre certas condições, o problema da estimação da resposta do canal é equivalente ao *fitting* do subespaço estimado do sinal com o subespaço gerado pela seqüência de espalhamento, que é conhecida. É demonstrado que este problema pode ser resolvido transformando-se em um problema de minimização sem restrições, cujo mínimo local é único e desse modo pode ser resolvido por um método de otimização do tipo gradiente, por exemplo.

A estruturação do capítulo é a seguinte: Na seção 6.2, é dado o modelo de sistema a ser utilizado pelo algoritmo. Na seção 6.3, os detectores lineares descritos anteriormente são obtidos. Na seção 6.4, a estimação de canal cega é desenvolvida. Na seção 6.5, são apresentados resultados envolvendo a detecção de sinal e também a equalização de canal. Por último, na seção 6.6, as conclusões são apresentadas.

## 6.2 Modelo do Sistema

O modelo de sistema que será utilizado neste algoritmo consiste de um canal com apenas um subcanal porém com  $K$  usuários onde  $K \geq 1$  dividindo a mesma faixa de freqüência. Vamos considerar aqui um modelo de sistema digital em banda base e também em espalhamento espectral por seqüência direta [3]. O sinal transmitido pelo  $m$ -ésimo usuário, vide Figura 6.1, pode ser escrito como

$$x(m, k) = \sum_{i=-M}^M b(m, i)c(m, k - iT) \quad m = 1, \dots, K \quad (6.1)$$

onde  $2M + 1$  é o número de símbolos por quadro,  $T$  é a duração do símbolo,  $b(m, i)$  denota o  $i$ -ésimo bit de informação do  $m$ -ésimo usuário, consistindo de



amostras do tipo  $\pm 1$  e  $c(m, k)$  é a seqüência de espalhamento do  $m$ -ésimo usuário no instante  $k$ . A informação total transmitida para  $K$  usuários é dada por

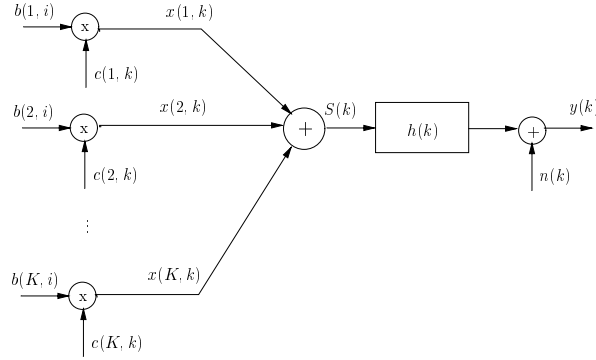


Figura 6.1: Modelo do sistema CDMA no enlace direto ou *downlink*.

$$S(k) = \sum_{m=1}^K x(m, k) \quad (6.2)$$

A seqüência de espalhamento é definida no intervalo  $[0, T]$  e tem energia igual a 1. No espalhamento espectral por seqüência direta temos que as formas de onda são do tipo

$$c(m, k) = \sum_{j=0}^{\zeta-1} \beta(m, j) \psi(k - jT_c) \quad k \in [0, T] \quad (6.3)$$

onde  $\zeta$  é o comprimento da seqüência de espalhamento,  $[\beta(m, 0), \dots, \beta(m, \zeta - 1)]$  corresponde a amostras do tipo  $\pm 1$  da seqüência de espalhamento do  $m$ -ésimo usuário,  $\psi(k)$  é a forma de onda do chip e  $T_c = T/\zeta$  é a duração do chip. Vale a pena ressaltar aqui que todos os usuários estão em sincronismo, ou seja, a transmissão dos dados se inicia e termina no mesmo instante.

Vamos supor por enquanto que não haja multipercurso no canal, ou seja,  $h(k)$  é igual a um impulso. No receptor, que é do tipo filtro casado, forma-se um vetor  $\mathbf{y}(k)$  de dimensão  $\zeta \times 1$  cujas componentes são as saídas do filtro nos instantes de amostragem

$$\mathbf{y}(k) = \sum_{m=1}^K A(m)b(m)\mathbf{c}(m) + \mathbf{n}(k) \quad (6.4)$$

onde  $A(m)$  corresponde a amplitude do sinal recebido de cada usuário e

$$\mathbf{c}(m) = \frac{1}{\sqrt{\zeta}}[\beta(m, 0), \dots, \beta(m, \zeta - 1)]^T \quad (6.5)$$

é a seqüência de espalhamento normalizada para o  $m$ -ésimo usuário e  $\mathbf{n}(k)$  é o vetor de ruído do tipo AWGN com variância  $\sigma^2$ .

## 6.3 Detectores Multiusuário Cegos Baseados em Subespaços

### 6.3.1 Determinação de Subespaços

Assumindo que as seqüências de espalhamento sejam ortogonais, ou seja,

$$\frac{1}{\zeta} \sum_{l=0}^{\zeta-1} c(m, l) \times c(i, l) = \begin{cases} 1, & m = i \\ 0, & m \neq i \end{cases} \quad (6.6)$$

e também definindo  $\mathbf{C}$  a matriz das seqüências de espalhamento

$$\mathbf{C} = [\mathbf{c}(1), \dots, \mathbf{c}(K)] \quad (6.7)$$

e a matriz diagonal

$$\mathbf{A} = \begin{bmatrix} A^2(1) & & \\ & \ddots & \\ & & A^2(K) \end{bmatrix} \quad (6.8)$$

então, a matriz de autocorrelação do sinal recebido,  $\mathbf{y}(k)$ , é dada por

$$\mathbf{R}_y = E[\mathbf{y}\mathbf{y}^T] \quad (6.9)$$

$$\begin{aligned} &= \sum_{m=1}^K A(m) \mathbf{c}(m) \mathbf{c}^T(m) + \sigma^2 \mathbf{I}_\zeta \\ &= \mathbf{C} \mathbf{A} \mathbf{C}^T + \sigma^2 \mathbf{I}_\zeta \end{aligned} \quad (6.10)$$

onde  $\mathbf{I}_\zeta$  é a matriz identidade de ordem  $\zeta$ . Realizando agora a decomposição dos autovalores e autovetores da matriz  $\mathbf{R}_y$ , temos que

$$\begin{aligned}\mathbf{R}_y &= \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \\ &= [\mathbf{U}_s \mathbf{U}_n] \begin{bmatrix} \mathbf{\Lambda}_s & \\ & \mathbf{\Lambda}_n \end{bmatrix} \begin{bmatrix} \mathbf{U}_s \\ \mathbf{U}_n \end{bmatrix}\end{aligned}\quad (6.11)$$

com

$$\mathbf{\Lambda}_s = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_K \end{bmatrix}\quad (6.12)$$

contendo os  $K$  maiores autovalores da matriz  $\mathbf{R}_y$  em ordem decrescente e a matriz  $\mathbf{U}_s$

$$\mathbf{U}_s = [\mathbf{u}(1), \dots, \mathbf{u}(K)]\quad (6.13)$$

contém os correspondentes autovetores ortonormais. Enquanto isto

$$\mathbf{\Lambda}_n = \sigma^2 \mathbf{I}_{\zeta-K}\quad (6.14)$$

e

$$\mathbf{U}_n = [\mathbf{u}(K+1), \dots, \mathbf{u}(\zeta)]\quad (6.15)$$

contém os  $(\zeta - K)$  autovetores ortonormais que correspondem ao autovalor  $\sigma^2$ .

O espaço gerado por  $\mathbf{U}_s$  é denominado subespaço do sinal recebido e o seu complemento ortogonal é o subespaço do ruído, gerado por  $\mathbf{U}_n$ . Um detector multiusuário linear utilizado para demodular o  $m$ -ésimo usuário está na forma de um correlator seguido por um limitador descontínuo

$$\hat{b}(m) = \text{sgn}(\mathbf{w}^T(m)\mathbf{y})\quad (6.16)$$

onde  $\mathbf{w}(m) \in \mathbb{R}^\zeta$ . A seguir, serão dadas as expressões para dois detectores lineares multiusuários, denominados respectivamente detector de descorrelação e detector linear MMSE.

### 6.3.2 Detector Linear de Descorrelação

A matriz de correlação da seqüência de espalhamento é definida por

$$\mathbf{R}_c = \mathbf{C}^T \mathbf{C} \quad (6.17)$$

Desde que o posto ( $\mathbf{C}$ ) =  $K$ , temos que  $\mathbf{R}_c$  é não singular.

O detector de descorrelação é projetado para eliminar a interferência de múltiplo acesso causada pelos demais usuários. Ele tem a forma  $\mathbf{w}(m) = \mathbf{d}(m)$  na equação (6.16), dado por

$$\mathbf{d}(m) = \sum_{j=1}^K [\mathbf{R}_c^{-1}]_{mj} \mathbf{c}(j) \quad (6.18)$$

onde  $[\mathbf{R}_c^{-1}]_{mj}$  denota o  $(m, j)$ -ésimo elemento da matriz  $\mathbf{R}_c^{-1}$ . O produto interno entre o detector de descorrelação, definido na equação (6.18), e a seqüência de espalhamento tem a seguinte importante característica, [14]:

$$\mathbf{d}^T(j) \mathbf{c}(i) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \quad (6.19)$$

O detector de descorrelação  $\mathbf{d}(m)$  é dado em termos dos parâmetros do subespaço do sinal por [14]

$$\mathbf{d}(m) = \frac{1}{\mathbf{c}^T(m) \mathbf{U}_s (\mathbf{\Lambda}_s - \sigma^2 \mathbf{I}_K)^{-1} \mathbf{U}_s^T \mathbf{c}(m)} \mathbf{U}_s (\mathbf{\Lambda}_s - \sigma^2 \mathbf{I}_K)^{-1} \mathbf{U}_s^T \mathbf{c}(m) \quad (6.20)$$

### 6.3.3 Detector Linear MMSE

Um detector linear que minimiza o erro médio quadrático, detector MMSE [15] que aplicado à equação (6.16) onde o vetor de coeficientes é dado por  $\mathbf{w}(m) = \mathbf{p}(m)$ , tal que  $\mathbf{p}(m) \in \mathbb{R}^{\zeta}$  e, cujo o MSE é dado por

$$\begin{cases} \text{MSE}(\mathbf{p}(m)) = E[(A(m)b(m) - \mathbf{p}^T(m)\mathbf{y})^2] \\ \text{sujeito a } \mathbf{p}^T(m) \mathbf{c}(m) = 1 \end{cases} \quad (6.21)$$

Para se obter o vetor de coeficientes  $\mathbf{p}^T(m)$  devemos usar as equações (6.4) e (6.21) e também o método dos multiplicadores de Lagrange. Então seja a função

$$\mathcal{L}(\mathbf{p}(m)) = \text{MSE}(\mathbf{p}(m)) - 2\mu(\mathbf{p}^T(m)\mathbf{c}(m) - 1) \quad (6.22)$$

onde  $\mu$  denota o multiplicador de Lagrange. Assim temos que

$$\begin{aligned} \mathcal{L}(\mathbf{p}(m)) &= \mathbf{p}^T(m)E[\mathbf{y}\mathbf{y}^T]\mathbf{p}(m) - 2A(m)\mathbf{p}^T(m)E[b(m)\mathbf{y}] \\ &\quad + A^2(m) - 2\mu(\mathbf{p}^T(m)\mathbf{c}(m) - 1) \\ &= \mathbf{p}^T(m)\mathbf{R}_y\mathbf{p}(m) - 2[A^2(m) + \mu]\mathbf{p}^T(m)\mathbf{c}(m) + [A^2(m) + 2\mu] \end{aligned} \quad (6.23)$$

onde usamos  $E[\mathbf{y}\mathbf{y}^T] = \mathbf{R}_y$  e  $E[b(m)\mathbf{y}] = \mathbf{c}(m)$ . Desde que  $\mathbf{R}_y$  é positiva definida,  $\mathcal{L}(\mathbf{p}(m))$  é uma função estritamente convexa de  $\mathbf{p}(m)$ . Logo, resolvendo para  $\mathbf{p}(m)$  temos

$$\begin{aligned} \nabla\mathcal{L}(\mathbf{p}(m)) &= 0 \\ 2\mathbf{R}_y\mathbf{p}(m) - 2[A^2(m) + \mu]\mathbf{c}(m) &= 0 \\ \mathbf{R}_y\mathbf{p}(m) &= [A^2(m) + \mu]\mathbf{c}(m) \end{aligned}$$

e finalmente

$$\mathbf{p}(m) = [A^2(m) + \mu]\mathbf{R}_y^{-1}\mathbf{c}(m) \quad (6.24)$$

Usando a equação (6.11) e sabendo que o subespaço do ruído é ortogonal a  $\mathbf{c}(m)$ , ou seja,  $\mathbf{U}_n^T\mathbf{c}(m) = \mathbf{0}$ , temos que

$$\begin{aligned} \mathbf{p}(m) &= [A^2(m) + \mu]\mathbf{R}_y^{-1}\mathbf{c}(m) \\ &= [A^2(m) + \mu](\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m) + [A^2(m) + \mu]\sigma^{-2}(\mathbf{U}_n\mathbf{U}_n^T)\mathbf{c}(m) \\ &= [A^2(m) + \mu](\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m) \end{aligned} \quad (6.25)$$

Finalmente, da restrição  $\mathbf{p}^T(m)\mathbf{c}(m) = 1$ , obtemos

$$\begin{aligned} \mathbf{p}^T(m) &= [A^2(m) + \mu]\mathbf{c}^T(m)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T) \\ \mathbf{p}^T(m)\mathbf{c}(m) &= [A^2(m) + \mu]\mathbf{c}^T(m)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m) \end{aligned} \quad (6.26)$$

e então

$$A^2(m) + \mu = \frac{1}{\mathbf{c}^T(m)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m)} \quad (6.27)$$

Substituindo este resultado na equação (6.25), chegamos ao resultado desejado

$$\mathbf{p}(m) = \frac{1}{\mathbf{c}^T(m)(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m)}(\mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T)\mathbf{c}(m) \quad (6.28)$$

### 6.3.4 Detecção Usando Estimativas da Matriz de Autocorrelação

Vamos agora usar uma estimativa para a matriz de autocorrelação do sinal recebido  $\mathbf{R}_y$  e suas autocomponentes ao invés de usar seu valor exato. Na prática, a matriz de autocorrelação pode ser obtida através de  $q$  vetores de dados recebidos

$$\hat{\mathbf{R}}_y(q) = \frac{1}{q} \sum_{i=1}^q \mathbf{y}(i)\mathbf{y}^T(i) \quad (6.29)$$

Iremos examinar aqui a consistência e a variância assintótica das estimativas para ambos os detectores descritos previamente. Desde que a regra de decisão na equação (6.16) é invariante a um multiplicador positivo nos detectores lineares [16], vamos considerar aqui uma versão escalada dos detectores lineares  $\mathbf{d}(m)$  e  $\mathbf{p}(m)$ , dados por

$$\mathbf{d}(m) = \mathbf{U}_s(\mathbf{\Lambda}_s - \sigma^2\mathbf{I}_K)^{-1}\mathbf{U}_s^T\mathbf{c}(m) \quad (6.30)$$

$$\mathbf{p}(m) = \mathbf{U}_s\mathbf{\Lambda}_s^{-1}\mathbf{U}_s^T\mathbf{c}(m) \quad (6.31)$$

e seus respectivos valores polarizados

$$\hat{\mathbf{d}}(q) = \hat{\mathbf{U}}_s(q)(\hat{\mathbf{\Lambda}}_s(q) - \hat{\sigma}^2(q)\mathbf{I}_K)^{-1}\hat{\mathbf{U}}_s^T(q)\mathbf{c}(m) \quad (6.32)$$

$$\hat{\mathbf{p}}(q) = \hat{\mathbf{U}}_s(q)\hat{\mathbf{\Lambda}}_s^{-1}(q)\hat{\mathbf{U}}_s^T(q)\mathbf{c}(m) \quad (6.33)$$

onde  $\hat{\mathbf{U}}_s(q)$ ,  $\hat{\mathbf{\Lambda}}_s(q)$  e  $\hat{\sigma}^2(q)$  são as autocomponentes de  $\hat{\mathbf{R}}_y(q)$ , isto é

$$\hat{\mathbf{R}}_y(q) = \hat{\mathbf{U}}_s(q)\hat{\mathbf{\Lambda}}_s^{-1}(q)\hat{\mathbf{U}}_s^T(q) + \hat{\sigma}^2(q)\hat{\mathbf{U}}_n(q)\hat{\mathbf{U}}_n^T(q) \quad (6.34)$$

Assume-se que as amostras do sinal recebido são do tipo iid. Segue-se que quando  $q \rightarrow \infty$ ,  $\hat{\lambda}(m, q) \rightarrow \lambda(m)$ , logo  $\hat{\mathbf{u}}(m, q) \rightarrow \mathbf{u}(m)$  para  $m = 1, \dots, K$ . Então teremos

$$\lim_{q \rightarrow \infty} \sum_{m=1}^K \frac{1}{\hat{\lambda}_m} \hat{\mathbf{u}}(m, q) \hat{\mathbf{u}}^T(m, q) \mathbf{c}(m) \rightarrow \sum_{m=1}^K \frac{1}{\lambda_m} \mathbf{u}(m) \mathbf{u}^T(m) \mathbf{c}(m) = \mathbf{p}(m) \quad (6.35)$$

Do mesmo modo o detector de decorrelação tem convergência garantida, ou seja, conforme  $q \rightarrow \infty$ , temos que  $\hat{\mathbf{d}}(m, q) \rightarrow \mathbf{d}(m)$ . Conseqüentemente, ambos os detectores lineares multiusuário baseados nas amostras do sinal recebido são fortemente consistentes. Contudo, eles são polarizados para um número finito de amostras. Pode ser provado [14], que o erro de estimação assintótico para ambos os detectores é tal que

$$\|\hat{\mathbf{d}}(m, q) - \mathbf{d}(m)\| = O\left(\sqrt{\frac{1}{q} \log(\log q)}\right) \quad (6.36)$$

$$\|\hat{\mathbf{p}}(m, q) - \mathbf{p}(m)\| = O\left(\sqrt{\frac{1}{q} \log(\log q)}\right) \quad (6.37)$$

onde  $O(x)$  significa da ordem de grandeza de  $x$ .

## 6.4 Estimação de Canal Adaptativa Cega para Sinais CDMA

Nesta seção, é apresentado um método para a estimação cega adaptativa de canal para sinais CDMA. Suponha que  $K$  usuários estejam transmitindo em sincronismo em um canal que possui múltiplos percursos de propagação. O número de caminhos resolvidos por usuário é, dado por, [17]

$$L = \lceil WT_m \rceil \quad (6.38)$$

onde  $\lceil x \rceil$  significa o maior inteiro próximo de  $x$ ,  $W$  é a banda do sinal e  $T_m$  é a dispersão temporal do canal. A resposta ao impulso do canal pode ser representada

por

$$h(k, m) = \sum_{l=1}^L h_{m,l} \delta(k - (l-1)T_c) \quad (6.39)$$

onde  $h_{m,l}$  são os coeficientes complexos do canal. Assim o vetor complexo de dimensão  $\zeta \times 1$  obtido na saída do filtro casado é dado por

$$\mathbf{y}(k) = \sum_{m=1}^K A(m)b(m) \sum_{l=1}^L h_{m,l} \mathbf{c}(m, l) + \mathbf{n}(k) \quad (6.40)$$

$$= \sum_{m=1}^K A(m)b(m) \tilde{\mathbf{c}}(m, l) + \mathbf{n}(k) \quad (6.41)$$

onde  $\mathbf{c}(m, l)$  é o vetor que representa a seqüência de espalhamento atrasada do  $m$ -ésimo usuário, ou seja

$$\mathbf{c}(m, l) = \mathbf{c}(m, k - (l-1)T_c) \quad (6.42)$$

e  $\mathbf{n}(k)$  é o vetor de ruído do tipo AWGN com variância  $\sigma^2$ . O sinal  $\tilde{\mathbf{c}}(m, l) = \mathbf{C}(m)\mathbf{h}(m)$  é a seqüência de espalhamento composta recebida do  $m$ -ésimo usuário onde

$$\mathbf{C}(m) = [\mathbf{c}(m, 1), \dots, \mathbf{c}(m, L)] \quad (6.43)$$

$$\mathbf{h}(m) = [h(m, 1), \dots, h(m, L)]^T \quad (6.44)$$

Suponha que o subespaço do sinal seja identificado por

$$\tilde{\mathbf{U}}_s = [\tilde{\mathbf{u}}(1), \dots, \tilde{\mathbf{u}}(K)] \quad (6.45)$$

Desde que  $\tilde{\mathbf{c}}(m)$  pertence ao espaço gerado por  $\tilde{\mathbf{U}}_s$ , existe um vetor  $\mathbf{f}(m) \in \mathbb{C}^K$  tal que

$$\tilde{\mathbf{c}}(m) = \tilde{\mathbf{U}}_s(m)\mathbf{f}(m) \quad m = 1, \dots, K \quad (6.46)$$

Porém, temos também que

$$\tilde{\mathbf{c}}(m) = \mathbf{C}(m)\mathbf{h}(m) \quad m = 1, \dots, K \quad (6.47)$$



Assim temos um sistema de  $K$  equações lineares onde

$$\mathbf{C}(m)\mathbf{h}(m) = \tilde{\mathbf{U}}_s(m)\mathbf{f}(m) \quad (6.48)$$

Obviamente  $\tilde{\mathbf{c}}(m)$  é unicamente determinada até um fator de escalamento se, e somente se

$$\text{pôsto} \{sp\{\mathbf{C}(m)\} \cap sp\{\tilde{\mathbf{U}}_s\}\} = 1 \quad (6.49)$$

Essa condição de unicidade é assumida daqui por diante durante a apresentação do método recursivo para a estimação do canal.

Como  $\tilde{\mathbf{U}}_s$  é uma estimativa polarizada para o subespaço do sinal, temos que resolver o sistema de equações (6.48) no senso dos mínimos quadrados. Definindo então

$$\mathbf{D}(m) = \begin{bmatrix} \tilde{\mathbf{U}}_s & \mathbf{C}(m) \end{bmatrix} \quad (6.50)$$

$$\mathbf{g}(m) = [\mathbf{f}^T(m) \quad -\mathbf{h}^T(m)]^T \quad (6.51)$$

então  $\mathbf{g}(m)$  é a solução para a seguinte otimização

$$\begin{cases} \hat{\mathbf{g}}(m) = \underset{\mathbf{g}(m) \in \mathbb{C}^{K+L}}{\text{arg min}} \|\mathbf{D}(m)\mathbf{g}(m)\|^2 \\ \|\mathbf{g}(m)\|^2 = 1 \end{cases} \quad (6.52)$$

A solução para a equação acima é dada pelo autovetor associado ao menor autovalor da matriz  $\mathbf{D}^H(m)\mathbf{D}(m)$  com a restrição de possuir norma unitária. Usando o método da função penalidade [18] este problema pode ser transformado em um outro problema de otimização sem restrição. Para isso define-se a seguinte função:

$$q(\mathbf{g}(m), \alpha) = \frac{1}{2}\mathbf{g}^H(m)\mathbf{D}^H(m)\mathbf{D}(m)\mathbf{g}(m) + \frac{\alpha}{4}(\mathbf{g}^H(m)\mathbf{g}(m) - 1)^2 \quad (6.53)$$

onde  $\alpha$  é uma constante positiva. O vetor  $\mathbf{g}(m)$  é um ponto estacionário de  $q(\mathbf{g}(m), \alpha)$  se e somente se  $\mathbf{g}(m)$  for um múltiplo de qualquer autovetor da matriz

$\mathbf{D}^H(m)\mathbf{D}(m)$  com norma, [14]

$$\|\mathbf{g}\| = \sqrt{1 - \frac{\nu}{\alpha}} \quad (6.54)$$

onde  $\nu$  é o autovalor correspondente. Todos os demais pontos estacionários de  $q(\mathbf{g}(m), \alpha)$  são mínimos locais exceto quando for um múltiplo do autovetor associado ao menor autovalor da matriz  $\mathbf{D}^H(m)\mathbf{D}(m)$ . Nesse caso, a função alcança seu mínimo global.

Pelo resultado acima, qualquer algoritmo do tipo gradiente descendente para a minimização de  $q(\mathbf{g}(m), \alpha)$  tem convergência garantida para autovetor associado ao menor autovalor da matriz  $\mathbf{D}^H(m)\mathbf{D}(m)$ . A constante  $\alpha$  deve ser maior que o menor autovalor da mesma matriz [14].

A partir do modelo acima, um algoritmo para a estimação conjunta de canal de detecção multiusuário é apresentado. A partir da obtenção de  $\tilde{\mathbf{U}}_s$  forma-se a matriz  $\mathbf{D}(m)$  e executamos a seguinte linha para a atualização do vetor  $\mathbf{g}(m)$ , de acordo com o método de tipo gradiente

$$\begin{aligned} \mathbf{g}(m, k) &= \mathbf{g}(m, k - 1) \\ &\quad - \mu [\mathbf{D}^H(m)\mathbf{D}(m)\mathbf{g}(m, k - 1) + \alpha (\|\mathbf{g}(m, k - 1)\|^2 - 1) \mathbf{g}(m, k - 1)] \end{aligned} \quad (6.55)$$

A resposta estimada do canal  $\mathbf{h}(m, k)$  é um subvetor de  $\mathbf{g}(m, k)$  na  $k$ -ésima iteração. Após normalização, a seqüência de espalhamento  $\tilde{\mathbf{c}}(m, k)$  efetiva é encontrada juntamente com os componentes  $\tilde{\mathbf{U}}_s(k)$  e  $\tilde{\mathbf{\Lambda}}_s(k)$  para a estimação dos detectores lineares. Essa implementação adaptativa permite uma estimação de canal em tempo real com menos operações aritméticas de ponto flutuante que a versão não adaptativa.

## 6.5 Resultados Experimentais

Nesta seção, serão apresentados alguns resultados experimentais do algoritmo de estimação conjunta de canal e detecção multiusuário apresentado neste

capítulo. O modelo de sistema utilizado nas simulações foi de um sistema CDMA em banda base com detecção por filtro casado.

Experimento 1: No primeiro experimento, o objetivo é avaliar o desempenho dos detectores lineares multiusuários apresentados neste capítulo. O sistema utilizado usava seqüências de espalhamento ortogonais de Hadamard [3], com comprimento  $\zeta = 8$ . Não foi incluído multipercurso no canal, apenas ruído tipo AWGN. Foram usadas também  $q = 100$  vetores para realizar a estimação da matriz de autocorrelação das seqüências de espalhamento. Obtivemos curvas da taxa de erro variando-se o número de usuários  $K$  e a razão sinal-ruído. Todos os usuários tinham a mesma potência na transmissão. Na Figura 6.2.a é mostrado o gráfico de desempenho para  $\zeta = 8$ .

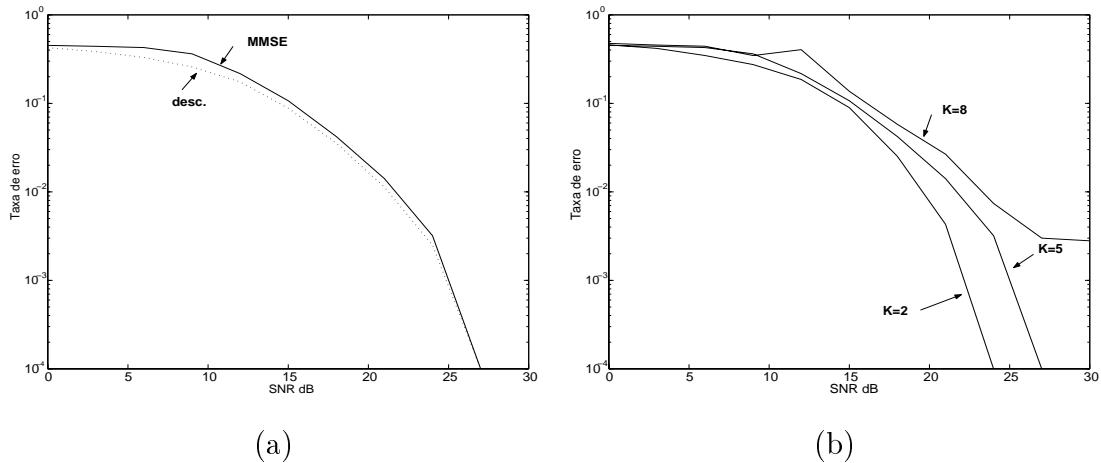


Figura 6.2: Desempenho dos detectores lineares para diferentes valores do número de usuários, para  $\zeta = 8$ .

Constata-se neste caso, como visto na Figura 6.2.a, que os detectores de decorrelação e MMSE têm desempenho bastante similar. Desse modo, apresentamos o resultado da taxa de erro apenas para um detector, o detector MMSE. Já na Figura 6.2.b, vemos que conforme o número de usuários aumenta temos um aumento na taxa de erro, principalmente na região de elevada SNR. Isto já

era de se esperar devido à maior interferência multiusuário.

Experimento 2: Neste segundo experimento, iremos avaliar o desempenho dos detectores lineares multiusuário apresentados agora com  $\zeta = 16$ . Também não consideramos multipercurso no canal, apenas ruído tipo AWGN. Neste caso, seus desempenhos não foram tão similares, sendo que o detector MMSE foi superior, ver na Figura 6.3.a. O sistema também usava seqüências de espalhamento ortogonais de Hadamard e as curvas foram obtidas variando-se o número de usuários  $K$  e a razão sinal-ruído. Todos os usuários tinham a mesma potência na transmissão. Na Figura 6.3.b é mostrado o gráfico de desempenho para  $\zeta = 16$ .

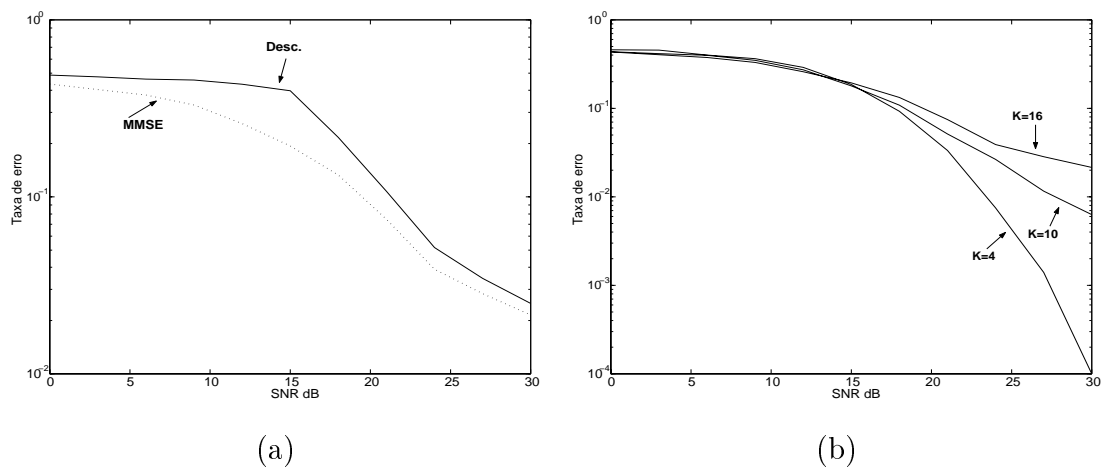


Figura 6.3: Desempenho dos detectores lineares para diferentes valores do número de usuários, para  $\zeta = 16$ .

Constata-se neste caso, como visto na Figura 6.3.a, que os detectores de descorrelação e MMSE têm desempenho não tão similar como no caso anterior. Desse modo, apresentamos o resultado da taxa de erro apenas para detector com melhor desempenho, o detector MMSE. Igualmente ao caso anterior, vemos na Figura 6.3.b que quando o número de usuários aumenta temos um aumento na taxa de erro, isto para SNR maior que 20 dB novamente isto já era de se esperar

devido a um acréscimo na interferência multiusuário.

Experimento 3: O primeiro exemplo no qual se realiza a estimação de canal, realizamos a identificação do canal  $\mathbf{h}_5$  da equação (6.56). Neste exemplo, temos um sistema no qual as seqüências de espalhamento tem comprimento  $\zeta = 8$  com  $K = 3$  usuários presentes, todos com a mesma potência na transmissão. Ao contrário dos algoritmos anteriores, neste caso não existe diversidade de canal para cada usuário. Para verificar o desempenho do algoritmo, realizamos a detecção do usuário  $m = 1$  e acompanhamos a variação da resposta estimada do canal, considerado aqui invariante no tempo ao decorrer de todo processo. O experimento foi realizado com SNR, considerando apenas o ruído tipo AWGN, igual a 15 dB. Na Figura 6.4 é mostrado o gráfico de desempenho do algoritmo para o canal  $\mathbf{h}_5$  dado pela equação (6.56).

$$\mathbf{h}_5 = \begin{bmatrix} 0.21 & 0.52 & -0.12 \end{bmatrix}^T \quad (6.56)$$

neste caso temos um canal com três coeficientes  $L = 3$ .

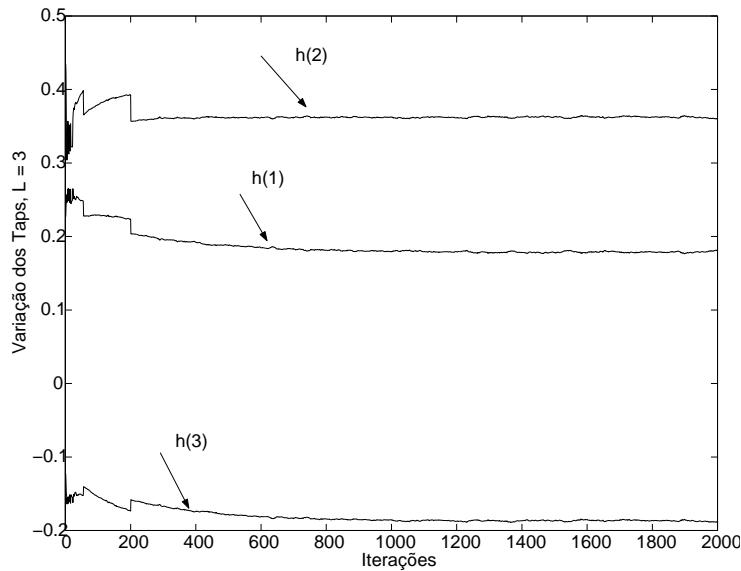


Figura 6.4: Desempenho do algoritmo Wang para o canal  $\mathbf{h}_5$ .

Como primeira observação, notamos que o valor médio para cada coefi-

ciente do canal estimado permanece praticamente constante o que assegura a convergência do algoritmo na prática, para este exemplo. Vemos também que os valores assintóticos dos coeficientes se aproximam bastante dos valores reais do canal, principalmente para o coeficiente  $\mathbf{h}_5(1)$ . Para os demais coeficientes também há convergência porém com maior erro relativo nesses casos.

Experimento 4: Neste outro exemplo, realizamos a identificação do canal  $\mathbf{h}_6$  da equação (6.57). Neste exemplo, temos um sistema com  $\zeta = 16$  e também com  $K = 3$  usuários presentes, todos também com a mesma potência na transmissão. Para verificar o desempenho do algoritmo, realizamos a detecção do primeiro usuário e acompanhamos a variação da resposta estimada do seu canal, considerado também invariante no tempo. O experimento foi realizado com a mesma razão sinal-ruído anterior, 15 dB. Na Figura 6.5 é mostrado o gráfico de desempenho do algoritmo.

$$\mathbf{h}_6 = \begin{bmatrix} 0.31 & 0.73 & 0.50 \end{bmatrix}^T \quad (6.57)$$

Neste caso temos também um canal com três coeficientes  $L = 3$ .

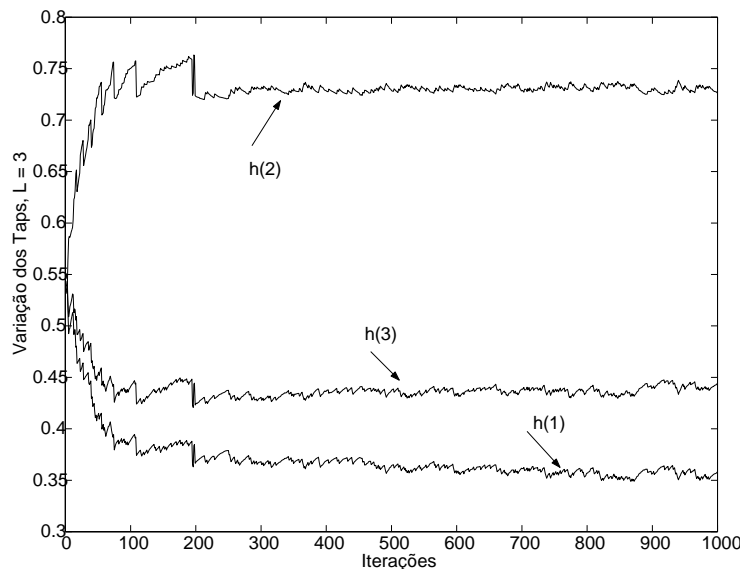


Figura 6.5: Desempenho do algoritmo Wang para o canal  $\mathbf{h}_6$ .

Também neste caso, observamos que o valor médio para cada coeficiente do canal estimado permanece constante, com os valores assintóticos dos coeficientes bastante próximos dos valores reais do canal, isto para todos os coeficientes.

Experimento 5: Neste quinto exemplo, realizamos a identificação e a equalização do canal  $\mathbf{h}_6$  da equação (6.57). Neste exemplo, temos um sistema com  $\zeta = 16$  e variamos o número de usuários presentes em  $K = 4$  e  $K = 8$ , todos também com a mesma potência na transmissão. Para verificar o desempenho do algoritmo, realizamos a detecção do primeiro usuário e computamos sua taxa de erro. O experimento foi realizado variando a razão sinal-ruído. Na Figura 6.6 é mostrado o gráfico de desempenho do algoritmo. Foram usadas 200 iterações.

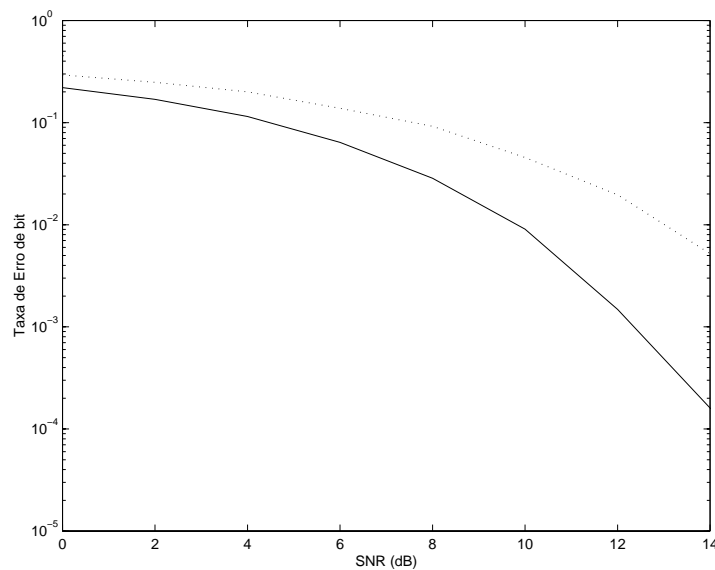


Figura 6.6: Desempenho do algoritmo Wang, na taxa de erro de bit, para o canal  $\mathbf{h}_6$ , com 4 usuários, (linha contínua), e 8 usuários, (linha pontilhada).

## 6.6 Conclusão

Dos detectores multiusuário analisados o detector do tipo MMSE teve o melhor desempenho porém quando o número de usuários é pequeno os desempenhos de ambos os detectores analisados são similares.

O algoritmo de Wang [14], assim como o algoritmo J-LSS [6], também é baseado na decomposição de subspaços e por isso possui um elevado custo computacional. Seu desempenho na identificação de canal é relativamente bom, pois ele depende, fundamentalmente, do número de coeficientes do canal a ser estimado, da SNR e do número de usuários presente no sistema. Para uma canal com poucos coeficientes e SNR maior que 10 dB a estimativa é em geral boa, produzindo boa equalização.

O algoritmo de Wang tem complexidade computacional alta. No experimento 5 foram realizadas  $1.8 \times 10^7$  operações de ponto flutuante para realizar a estimação de canal. Tanta complexidade é devido à decomposição matricial de autovalores que é usada extensivamente em todo o algoritmo.

O algoritmo não usa diversidade de canais em sua estrutura e isso torna a identificação do canal independente de sua resposta impulsional o que não acontece nos algoritmos previamente implementados, onde vimos que o desempenho depende fundamentalmente do canal a ser identificado, mais especificamente do número de condicionamento da matriz de filtragem. Isso é vantajoso para o algoritmo de Wang nos casos onde a matriz de filtragem daqueles algoritmos é mal condicionada.



# Capítulo 7

## Comparação no Desempenho dos Algoritmos

### 7.1 Introdução

Nesta capítulo, apresentaremos o resultado comparativo entre os diferentes algoritmos de identificação-equalização cega apresentados até aqui. Além dos algoritmos apresentados neste trabalho, também fazem parte da comparação um modelo de canal onde apenas o ruído se faz presente, cuja resposta impulsional é um impulso, o modelo AWGN, e também outro onde a equalização é supervisionada usando o algoritmo LMS. Nos dois primeiros experimentos são avaliados e comparados os algoritmos LMS, J-LSS, A-LSS e o algoritmo PE. Os parâmetros para cada algoritmo foram escolhidos de forma a otimizar seu desempenho. O número de amostras na entrada foi o mesmo,  $N = 500$ , para todos os algoritmos. Foram realizados 100 experimentos para valores inteiros da razão sinal-ruído na faixa indicada. Em cada experimento a seqüência a ser equalizado era composto de 1000 bits modulados em BPSK, usando pulso retangular como forma de onda. O resultado final obtido, para cada valor da SNR, foi a média dos valores da taxa de erro em todos os 100 experimentos. Os experimentos 3 e 4 são realizados

em ambiente de espalhamento espectral sendo assim, possível a comparação do algoritmo de Wang com os demais. E por último, no experimento 5, o algoritmo Bussgang é incluído para a comparação geral entre os métodos cegos.

## 7.2 Resultados Experimentais

Experimento 1: Realizar a equalização para o canal  $\mathbf{h}_1$  da equação (3.36).

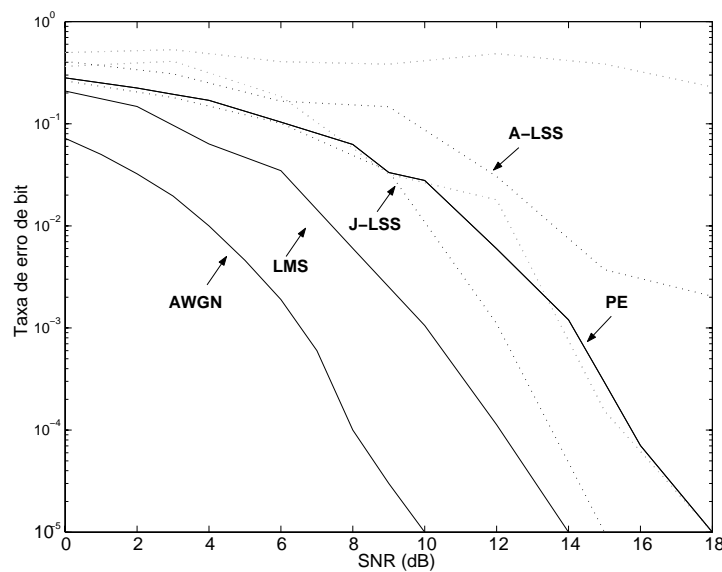


Figura 7.1: Desempenho comparativo dos algoritmos para o canal  $\mathbf{h}_1$ .

Na Figura 7.1 podemos ver o desempenho dos algoritmos LMS, J-LSS, A-LSS, PE e ainda o modelo de canal AWGN. Vemos que os métodos supervisionados ainda têm uma vantagem na equalização de sistemas. Neste exemplo, o algoritmo J-LSS supera o algoritmo PE em desempenho. Isso devido principalmente a quantidade de amostras usada no processo, 500 amostras, o que torna o J-LSS bastante eficiente. O algoritmo A-LSS tem o pior desempenho de todos. Logo a seguir, é dada na Tabela 7.1 o custo computacional de cada algoritmo.

Experimento 2: Realizar a equalização para o canal  $\mathbf{h}_2$  da equação (3.37).

Tabela 7.1: Comparativo de custo computacional experimento 1.

Algoritmo	op. ponto flutuante
LMS	$3.58 \times 10^4$
J-LSS	$1.15 \times 10^8$
A-LSS	$2.45 \times 10^5$
PE	$1.05 \times 10^5$

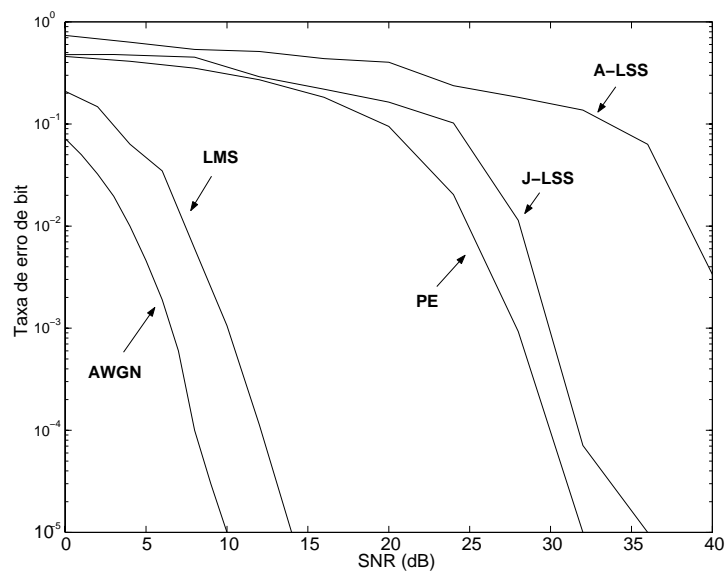
Figura 7.2: Desempenho comparativo dos algoritmos para o canal  $h_2$ .

Tabela 7.2: Comparativo de custo computacional experimento 2.

Algoritmo	op. de ponto flutuante
LMS	$3.58 \times 10^4$
J-LSS	$4.34 \times 10^8$
A-LSS	$2.58 \times 10^5$
PE	$3.45 \times 10^5$

Neste caso, como podemos ver na Figura 7.2, vemos que o algoritmo PE supera o algoritmo J-LSS, mais seus desempenhos ainda são inferiores ao LMS. O desempenho superior do algoritmo PE nesta caso em relação ao J-LSS pode ser explicado pela sua robustez ao difícil problema da identificação da ordem de canal, que se agrava ainda mais, quando a matriz de filtragem é mal condicionada. O algoritmo LMS que é insensível a esse problema, pois não usa diversidade de canais em sua estrutura, continua tendo desempenho superior tanto na taxa de erro de bit. Já o A-LSS continua tendo desempenho ruim devido principalmente aos erros na estimação da ordem do canal. Na Tabela 7.2 são dadas informações sobre o custo computacional de cada algoritmo, para este experimento.

Experimento 3: Neste experimento, é apresentada a comparação entre os algoritmos J-LSS, PE e o algoritmo de Wang. Neste caso, os dois primeiros algoritmos foram adaptados para funcionar com sinais em espalhamento espectral. Os algoritmos J-LSS e PE ainda faziam uso da diversidade de canais, com 2 subcanais por canal. Já o algoritmo de Wang faz uso de um canal com apenas um subcanal, a primeira da linha da equação (7.1). Em ambos os casos a transmissão foi realizada no sentido estação rádio-base estação móvel, ou seja, no *downlink*. Dessa forma, os sinais de todos os usuários estão sincronizados minimizando a interferência entre símbolos. Da mesma forma que nos experimentos anteriores,

os parâmetros para cada algoritmo foram escolhidos de forma a otimizar seu desempenho. O número de amostras na entrada foi o mesmo,  $N = 500$ , para todos os algoritmos. Foram realizados 100 experimentos para valores inteiros da razão sinal-ruído na faixa indicada. Em cada experimento a seqüência de entrada era um sinal em banda de base composta por 1000 bits espalhados no espectro, em seqüência direta, usando códigos de Hadamard de comprimento igual a 16. O número de usuários foi variado em 4, 8 e 16 respectivamente. O resultado final obtido, para cada valor da SNR, foi a média dos valores da taxa de erro em todos os 100 experimentos. Os gráficos de desempenho são mostrados nas Figuras 7.3, 7.4 e 7.5.

$$\mathbf{h}_7 = \begin{bmatrix} -0.12 & 0.51 & 0.43 \\ 0.34 & 0.98 & -0.12 \end{bmatrix}^T \quad (7.1)$$

Aqui vemos que o algoritmo de Wang sofre um perda de eficiência conforme

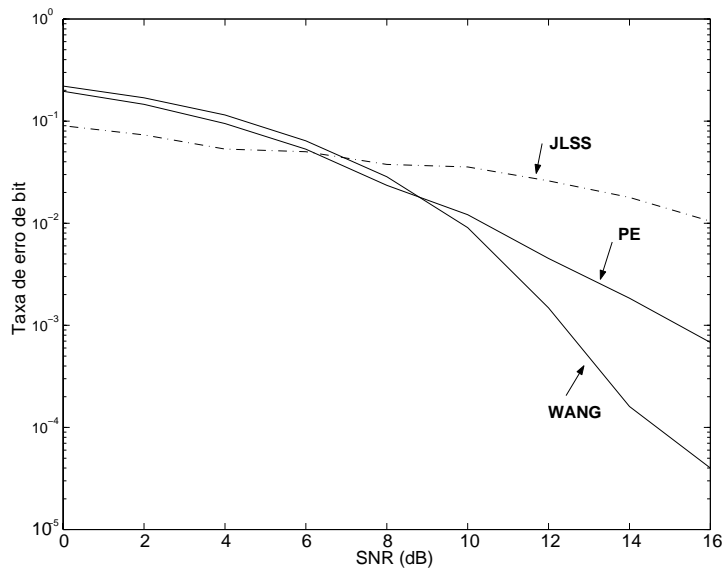


Figura 7.3: Desempenho comparativo dos algoritmos para o canal  $\mathbf{h}_7$  usando sinais CDMA com 4 usuários.

o número de usuários aumenta, isto devido principalmente ao aumento da inter-

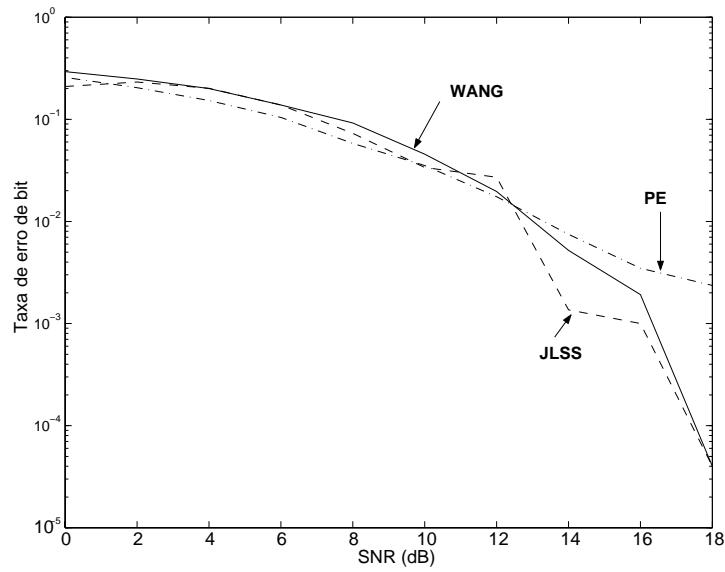


Figura 7.4: Desempenho comparativo dos algoritmos para o canal  $h_7$  usando sinais CDMA com 8 usuários.

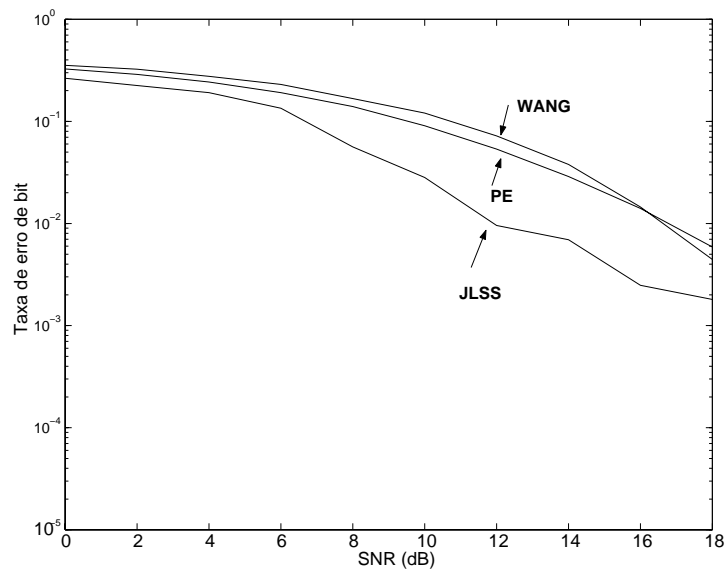


Figura 7.5: Desempenho comparativo dos algoritmos para o canal  $h_7$  usando sinais CDMA com 16 usuários.

Tabela 7.3: Comparativo de custo computacional no experimento 3.

Algoritmo	op. de ponto flutuante
J-LSS	$4.99 \times 10^7$
PE	$5.58 \times 10^4$
WANG	$1.83 \times 10^7$

ferência multiusuários no sistema. Conforme se aumenta o número de usuários no sistema, o algoritmo de melhor desempenho passa a ser o J-LSS.

Experimento 4: Neste último experimento, iremos realizar a equalização de um canal onde o sinal transmitido também é espalhado no espectro, porém agora são realizadas as transmissões assíncrona, transmissão no sentido estação-móvel para estação de rádio-base, e síncrona transmissão no sentido estação de rádio-base para estação-móvel. As quantidades foram as mesmas do experimento 3, porém usando apenas 8 usuários. Foi usada transmissão de sinal em BPSK e, dessa forma, apenas os algoritmos J-LSS e PE foram comparados. O canal usado foi da equação (3.36). Os gráficos de desempenho são mostrados nas Figuras 7.6 e 7.7.

Neste experimento, vemos que a transmissão assíncrona leva a uma perda de eficiência no sistema, isto devido a propriedade da não ortogonalidade total dos códigos de espalhamento, em ambientes deste tipo. Com isso, aumenta-se a correlação entre os sinais de usuários diferentes e dificulta-se a recepção dos mesmos.

Experimento 5: Realizar a equalização para o canal  $\mathbf{h}_7$ , equação (7.1), porém com a seqüência de entrada em banda de base.

Neste caso o algoritmo J-LSS supera o algoritmo PE e o algoritmo Buss-

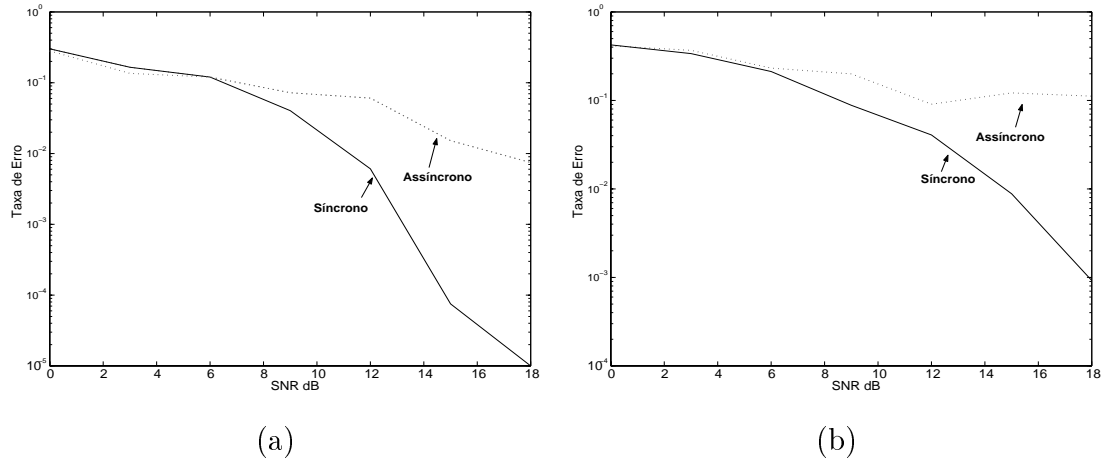


Figura 7.6: Desempenho dos algoritmos J-LSS (a) e PE (b) nos casos síncrono e assíncrono.

Tabela 7.4: Comparativo de custo computacional experimento 5.

Algoritmo	op. de ponto flutuante
J-LSS	$2.68 \times 10^6$
Bussgang	$2.01 \times 10^4$
PE	$5.85 \times 10^4$

gang tem o pior desempenho de todos. Logo a seguir, na Tabela 7.4 são dadas informações sobre o custo computacional de cada algoritmo, para este experimento.

### 7.3 Conclusão

A principal conclusão que chegamos ao fim deste capítulo, no que se refere a taxa de erro, é a que os algoritmos de equalização cega ainda têm um desempenho muito inferior aos métodos supervisionados, o que pode ser concluído observando



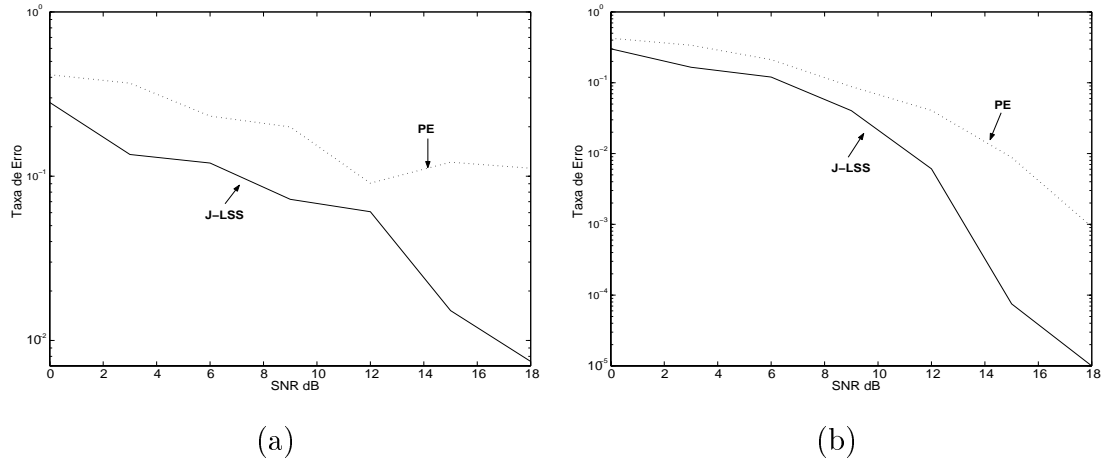


Figura 7.7: Comparação no desempenho dos algoritmos J-LSS e PE, nos casos assíncrono, (a) e síncrono, (b).

as Figuras 7.1 e 7.2. Contudo, seu desempenho, que é dependente do canal a ser equalizado, melhora sensivelmente com o aumento da SNR.

Os algoritmos J-LSS e PE têm desempenho similar, porém quando o número de amostras disponíveis para processamento é grande, o algoritmo J-LSS tem desempenho superior, devido a construção mais precisa dos subespaços da entrada e da saída. Porém, como o custo computacional do J-LSS é muitas vezes maior que o custo do algoritmo de PE, sua utilização é recomendada apenas em situações onde o processamento é feito em tempo não real.

Para sinais em espalhamento espectral, o algoritmo de Wang tem o melhor desempenho para um pequeno número de usuários presentes no sistema. Com o seu aumento destas quantidades, os algoritmos analisados neste caso, J-LSS, PE e Wang, tem desempenho similar.

Vimos também que para a transmissão assíncrona que a taxa de erro na recepção é maior na transmissão síncrona, e isto é devido a maior correlação entre as seqüências de espalhamento dos diferentes usuários não sincronizados.

O algoritmo Bussgang tem o pior desempenho constatado entre os algoritmos analisados. Seu uso deve ser limitado apenas a sistemas de banda de base

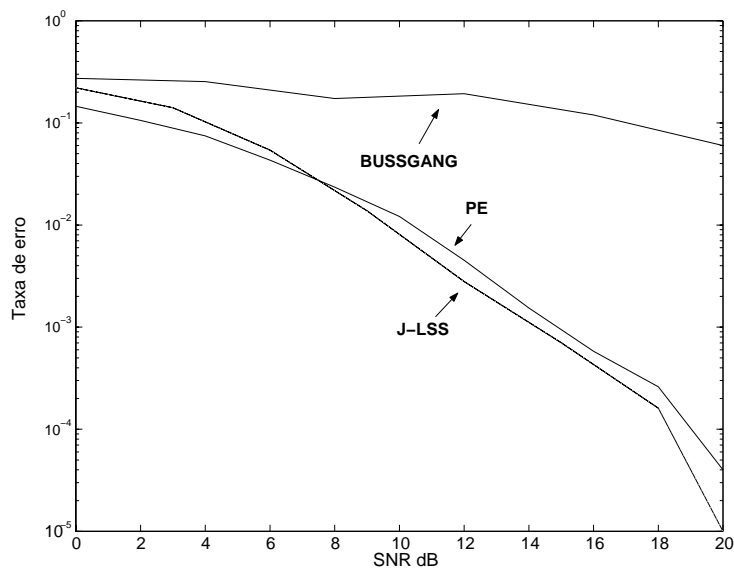


Figura 7.8: Desempenho comparativo dos algoritmos para o canal  $h_7$  em banda de base.

onde algum tipo de código de correção de erro é utilizada. Porém, como foi visto, seu custo computacional é baixo, ideal para sistemas que operam em tempo real.

# Capítulo 8

## Equalização Semicega

### 8.1 Introdução

Nos capítulos anteriores, constatamos que os métodos supervisionados, possuem desempenho melhor na equalização, no que se refere à taxa de erro do sinal equalizado. E sabemos também que nos atuais sistemas digitais de comunicações é necessária a transmissão de informação redundante que é conhecida também pelo receptor para diversos propósitos como sincronização entre outros. Desse modo, essa informação, conhecida a priori, pode ser usada para produzir um novo esquema de equalização, combinando os métodos supervisionados e os métodos cegos de modo a produzir uma equalização que seja um meio termo entre essas duas técnicas. A esse esquema dá-se o nome de equalização semicega.

Neste esquema, uma pequena quantidade de dados conhecidos no receptor é usada para produzir a equalização supervisionada e, então, usa-se a resposta do modelo obtida com este breve período de treinamento como resposta inicial para o modo cego, esperamos assim uma convergência mais rápida do processo e também um desempenho melhor que no método puramente cego.

Um esquema alternativo também é possível onde primeiramente realizamos a equalização cega e utilizamos a resposta do modelo obtida como resposta inicial

para o modo supervisionado. Ambas os esquemas serão tratados neste capítulo.

## 8.2 Modelo de Sistema

Como modelo de sistema apresentamos as configurações vistas na Figura 8.1.

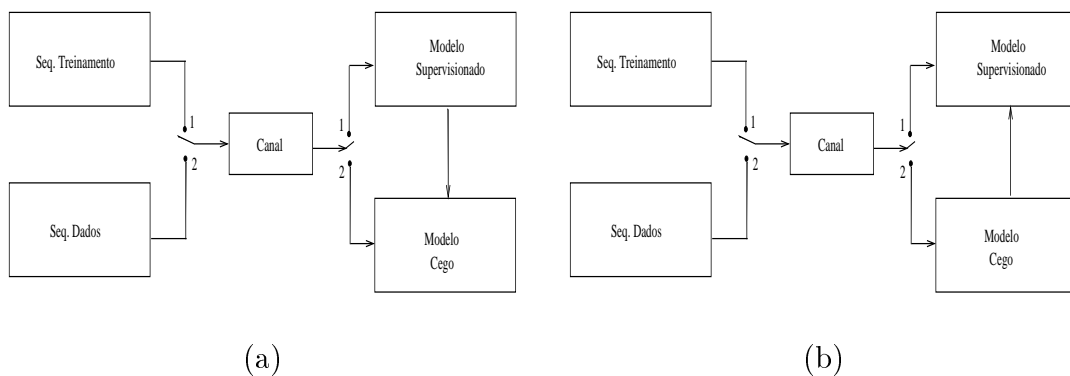


Figura 8.1: Configurações para equalização semicega: Supervisionado como resposta inicial (a) e cega como resposta inicial (b).

A Figura 8.1.a corresponde ao caso onde primeiro é realizado o treinamento supervisionado e depois a equalização cega. Já na Figura 8.1.b ocorre o inverso, primeiro a modelagem cega e depois a supervisionada.

## 8.3 Resultados Experimentais

Neste seção, fizemos uso de ambos os esquemas das Figuras 8.1.a e 8.1.b para verificar se houve melhoria na resposta do sinal equalizado, que deve produzir menor taxa de erro na equalização. No esquema (a), o treinamento supervisionado é realizado usando o algoritmo LMS com apenas 100 amostras na entrada. A resposta obtida com o LMS era usada como resposta inicial para o método

cego subsequente, que usava um número maior de amostras, 500 amostras, para a equalização. Já no esquema (b), primeiro era realizada a modelagem cega com 500 amostras e depois o treinamento supervisionado, também com 100 amostras, e tendo como resposta inicial a resposta obtida pela modelagem cega. Os gráficos de desempenho são mostrados nas Figuras 8.2, 8.3 e 8.4.

Experimento 1: Usando o algoritmo A-LSS na equalização do canal  $\mathbf{h}_1$  da equação (3.36).

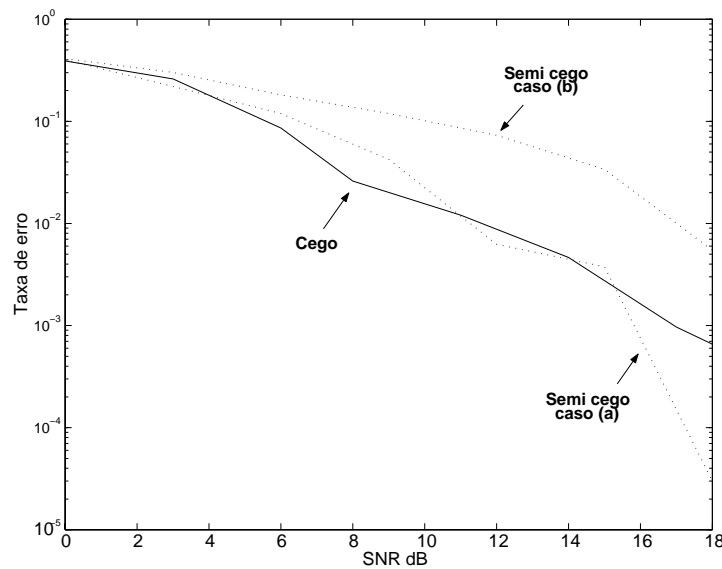


Figura 8.2: Desempenho comparativo do algoritmo A-LSS nos modos semicego e cego.

Experimento 2: Usando o algoritmo PE na equalização do canal  $\mathbf{h}_1$  da equação (3.36).

Experimento 3: Usando o algoritmo WANG na equalização do canal  $\mathbf{h}_5$  da equação (6.56).

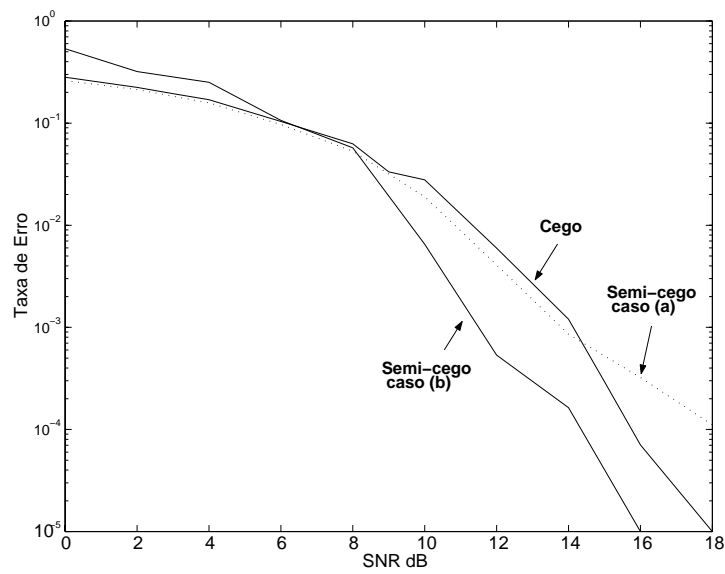


Figura 8.3: Desempenho comparativo do algoritmo PE nos modos semicego e cego.

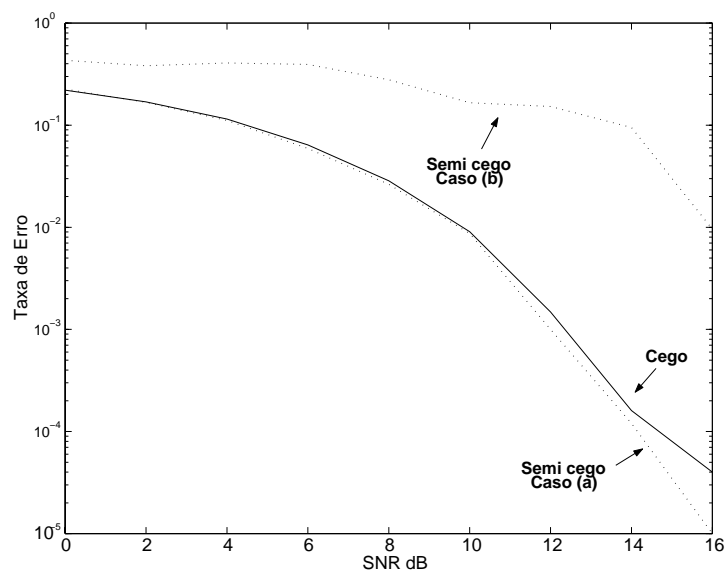


Figura 8.4: Desempenho comparativo do algoritmo WANG nos modos semicego e cego.

## 8.4 Conclusão

Nos três experimentos exibidos neste capítulo, vimos que todos os algoritmos apresentaram alguma melhoria em algum caso de equalização semicega, porém não nos mesmos casos.

Os algoritmos A-LSS e WANG apresentaram melhoria na taxa de erro usando a topologia ou caso (a), e mesmo assim para valores mais elevados da SNR. Essa melhoria pode ser explicada, em parte, devido a esses dois algoritmos realizarem a identificação de canal ao invés da equalização direta. A identificação de canal é realizada mais precisamente, no caso de poucas amostras no sinal de entrada, utilizando métodos supervisionados e usando a resposta obtida aí como resposta inicial para a modelagem cega. Desse modo, produz-se melhores resultados.

Porém, no caso do algoritmo PE, que realiza equalização direta, o curto período de treinamento inicial é insuficiente para produzir uma boa resposta inicial a ser usada pelo modelo cego no caso (a). Por isso, seu desempenho neste caso é inferior ao caso (b) onde a modelagem cega inicial produz sozinho bons resultados.

A modelagem semicega pode ser usada ainda de modo diferente como: Identificação da constante de ganho do canal no caso dos métodos baseados na decomposição de subespaços e também na determinação do parâmetro  $\gamma$  do algoritmo de PE. Em ambos os casos, a transmissão de informação redundante e conhecida no receptor é de fundamental importância para a determinação destes parâmetros. No caso dos algoritmos baseados na decomposição de subespaços, a constante de ganho pode ter valor negativo o que inverteria a informação recebida após a equalização, ou seja, bit 0 seria identificado como bit 1 e vice versa. E no algoritmo PE pode-se otimizar o valor de  $\gamma$  a partir do seu valor inicial  $\gamma_0$ , medindo a taxa de erro de bit para diferentes valores deste parâmetro e escolhendo aquele que produz menor taxa de erro.

# Capítulo 9

## Conclusão e Trabalhos Futuros

Os algoritmos de equalização cega ainda têm desempenho inferior aos algoritmos que fazem a equalização a partir do uso de seqüências de treinamento. Isto pode ser evidenciado pelos experimentos realizados onde a taxa de erro do sinal equalizado é menor para algoritmos supervisionados. Isso não tira o mérito dos algoritmos de equalização cega, apenas propõem mudanças que melhorem seu desempenho.

### 9.1 Objetivos Alcançados por esta Tese

Nesta tese, fizemos uma análise comparativa abrangente e atual de alguns dos mais importantes algoritmos de equalização cega [19].

O algoritmo J-LSS mostrou-se o mais eficiente na equalização de canais com diversidade, isso se considerarmos poucas amostras no processamento, produzindo menor taxa de erro de símbolos. Porém seu desempenho é ofuscado pelo seu elevado custo computacional que foi o maior entre todos os algoritmos implementados. Sua versão adaptativa oferece a solução para o problema da complexidade nos cálculos porém seu método de detecção da ordem do canal é complexo e bastante susceptível a erros. Uma modelagem semicega deve ser aplicada para se obter com exatidão a resposta do canal de transmissão, pois, como



vimos, a resposta obtida é um múltiplo do canal na realidade.

O algoritmo de predição de erro realizou, na maioria dos experimentos, uma boa equalização cega e entre todos os algoritmos analisados é o que possui a segunda menor complexidade computacional, perdendo apenas para o algoritmo Bussgang. Além disto, por ser um método de equalização direta o algoritmo PE leva vantagem sobre os demais por não precisar inverter a resposta do canal estimado. Seu principal problema está na determinação do parâmetro  $\gamma$  que é responsável pela remoção de ruído no sistema. Uma determinação não ajustada deste parâmetro torna a performance do algoritmo ruim. Para a determinação correta deste parâmetro, a modelagem semicega torna-se necessária.

O algoritmo de aproximação de subespaços de Wang, que possui elevado custo computacional devido às operações matriciais complexas, tem o melhor desempenho para sinais em espalhamento espectral [20], onde o número de usuários é pequeno e a relação sinal ruído é maior que 20 dB. Contudo para valores abaixo deste, todos os algoritmos tiveram desempenho equivalente, no critério taxa de erro. Sendo assim, a escolha daquele com menor custo computacional é a mais indicada.

Todos os algoritmos aqui implementados funcionam a partir de um número finito de amostras. Isso os torna mais próximos para a solução de problemas em tempo real onde poucas amostras para processamento são disponíveis.

O problema da diversidade de canais pode ser contornado pela sobreamostragem do sinal recebido. Fica constatado que para canais longos, com resposta impulsional com muitos coeficientes, a equalização se torna mais difícil pois, neste caso, há um número maior de zeros na resposta impulsional e o risco de haver zeros em comum é maior.

## 9.2 Trabalhos Futuros

Como proposta para trabalhos futuros, podemos citar o desenvolvimento de métodos semi-cegos mais eficientes que possam unir a melhor capacidade de equalização dos métodos supervisionados com a versatilidade dos métodos cegos.

Os algoritmos cegos são uma realidade e seu modo de operação desonera os sistemas digitais atuais de várias deficiências como: Desperdício de banda passante para a transmissão das seqüências de treinamento e perdas de desempenho por desvanecimento da resposta do canal. Com isso, seu uso deve ser encorajado.

# Referências Bibliográficas

- [1] HAYKIN, S., *Adaptative Filter Theory*. Prentice-Hall, 1996.
- [2] DINIZ, P. S. R., *Adaptative Filtering: Algorithms and Practical Implementation*. Kluwer Academic Publishers, 1997.
- [3] SKYLAR, B., *Digital Communications: Fundamentals and Application*. Prentice-Hall, 1988.
- [4] GARDNER, W. A., “A New Method of Channel Identification”, *IEEE Transactions on Communications*, v. 39, n. 8, pp. 813–817, 1991.
- [5] TONG, L., e PERREAU, S., “Multichannel Blind Channel Estimation: from Subspace to Maximum Likelihood Methods”, *Proc. IEEE*, v. 86, n. 10, pp. 1951–1968, Outubro 1998.
- [6] TONG, L., e ZHAO Q., “Joint Order Detection and Blind Channel Estimation by Least Squares Smoothing”, *IEEE Transactions on Signal Processing*, v. 47, n. 9, pp. 2345–2355, Setembro 1999.
- [7] STRANG, G., *Linear Algebra and Its Applications*. Harcourt Brace Jovanovich, 1988.
- [8] ZHAO Q., e TONG, L., “Adaptative Blind Channel Estimation by Least Squares Smoothing”, *IEEE Transactions on Signal Processing*, v. 47, n. 11, pp. 3000–3012, Novembro 1999.

- 
- [9] GESBERT, D., e DUHAMEL, P., “Unbiased Blind Adaptive Channel Identification and Equalization”, *IEEE Transactions on Signal Processing*, v. 48, n. 1, pp. 148–157, Janeiro 2000.
- [10] TREICHLER, J. R., “ $\gamma$ -LMS and its Use in a Noise Compensating Adaptive Spectral Analysis Technique”. In: *Proc. ICASSP*, 1979.
- [11] MACCHI, O., e HACHICHA, A., “Self Adaptive Equalization Based on a Prediction Principle”. In: *Proc. GLOBECOM*, 1986.
- [12] CHEN, D., e ROY, S., “An Adaptive Multiuser Receiver for CDMA Systems”, *IEEE J. Select Areas Communications*, v. 12, 1994.
- [13] MITRA, U., e POOR, V., “Adaptive Receiver Algorithms for Near-far Resistant CDMA”, *IEEE Trans. Communications*, v. 43, 1995.
- [14] WANG, X., e POOR, H. V., “Blind Multiuser Detection: A Subspace Approach”, *IEEE Transactions on Information Theory*, v. 44, n. 2, pp. 677–689, Março 1998.
- [15] HONIG, M., MADHOW, U., e VERDÚ S., “Blind Multiuser Detection”, *IEEE Transactions on Information Theory*, v. 41, 1995.
- [16] SHANMUGAN, S., e BREIPOHL, A., *Random Signals: Detection, Estimation and Data Analysis*. John Wiley and Sons, 1988.
- [17] PROAKIS, J., *Digital Communications*. McGraw-Hill, 1995.
- [18] LUENBERGER, D. G., *Linear and Nonlinear Programming*. Addison-Wesley, 1989.
- [19] VLIESE, A. F. C., NETTO, S. L., CAMPOS, M. L. R. e DINIZ, P. S. R., “Performance of adaptation algorithms in multipath channel equalization for CDMA systems”. In: *Proc. ICECS*, 1999.

- 
- [20] VLIESE, A. F. C., NETTO, S. L., CAMPOS, e DINIZ, P. S. R., “Performance comparisons of adaptive algorithms for blind equalization”. In: *Proc. 1st SAWCAS*, 2000.