

**Francisco José Alves de Aquino**

**PROCESSAMENTO LARGAMENTE LINEAR  
APLICADO AO PROBLEMA DE EQUALIZAÇÃO DE  
CANAL DE COMUNICAÇÃO DIGITAL**

**Florianópolis  
2008**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**

**PROGRAMA DE PÓS-GRADUAÇÃO  
EM ENGENHARIA ELÉTRICA  
GRUPO DE PESQUISA EM COMUNICAÇÕES (GPqCom)**

**PROCESSAMENTO LARGAMENTE LINEAR  
APLICADO AO PROBLEMA DE EQUALIZAÇÃO DE  
CANAL DE COMUNICAÇÃO DIGITAL**

Tese submetida à  
Universidade Federal de Santa Catarina  
como requisito parcial à obtenção do grau de  
Doutor em Engenharia Elétrica.

**FRANCISCO JOSÉ ALVES DE AQUINO**

Florianópolis, setembro de 2008.

# PROCESSAMENTO LARGAMENTE LINEAR APLICADO AO PROBLEMA DE EQUALIZAÇÃO DE CANAL DE COMUNICAÇÃO DIGITAL

Francisco José Alves de Aquino

Esta tese foi julgada adequada para a obtenção do título de Doutor em Engenharia Elétrica, área de concentração *Comunicações e Processamento de Sinais*, sendo aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.

---

Prof. Carlos Aurélio Faria da Rocha, Dr.  
Orientador

---

Prof. Leonardo Silva Resende, Dr.  
Co-orientador

---

Prof.<sup>a</sup> XXXXXXXXXXXX, Dra.  
Coordenadora do Curso de Pós-Graduação em Engenharia Elétrica  
da Universidade Federal de Santa Catarina

Banca Examinadora:

---

Prof. Leonardo Silva Resende, Dr.  
Presidente

---

Prof. Bartolomeu Uchôa-Filho, Ph.D.

---

Prof. José Carlos Moreira Bermudez, Ph.D.

---

Prof. XXXXXXXXXXXXXXXX, Dr.

---

Prof. XXXXXXXXXXXXXXXX, Dr.  
Relator

**SUBSTITUIR ESSA PÁGINA!**  
**Pela folha de assinaturas**

*À minha esposa, Ana Vlândia,  
à minha mãe, Francisca Tenório,  
à minha sogra, Maria Fátima.*

*Aos meus filhos, Catherine, Caroline e Bruno.*

*Porventura agradecerá ao servo, porque este fez o que lhe foi mandado? Assim também vós, quando fizerdes tudo o que vos for mandado, dizei: 'Somos servos inúteis; fizemos somente o que devíamos fazer'.*

*Lucas, 17, 9-10.*

*Jesus respondeu-lhes: "Por causa da vossa pouca fé; pois em verdade vos digo que, se tiverdes fé como um grão de mostarda direis a este monte:transporta-te daqui para lá, e ele se transportará, e nada vos será impossível.*

*Mateus, 17, 20.*

*O excesso de luz cega a vista.  
O excesso de som ensurdece o ouvido.  
Condimentos em demasia estragam o gosto.  
O ímpeto das paixões perturba o coração.  
A cobiça do impossível destrói a ética.  
Por isso, o sábio em sua alma  
Determina a medida para cada coisa.  
Todas as coisas visíveis lhe são apenas  
Setas que apontam para o Invisível.*

*Lao-Tsé,  
Tao Te Ching.*

## ***Agradecimentos***

*Meu reconhecimento à minha família, especialmente à minha esposa Ana Vlândia pelo suporte emocional, sem o qual este trabalho não teria sido possível, à minha mãe Francisca Tenório que sempre me estimulou a estudar seriamente desde o início e à minha sogra Maria Fátima, que muitas vezes foi uma segunda mãe para mim. Ao meu irmão, Dr. Jakson Aquino, que teve a paciência de ler um rascunho desta tese e ajudou a encontrar alguns erros de redação.*

*O meu muito obrigado a todos aqueles que contribuíram direta ou indiretamente para a realização deste trabalho. Em especial aos meus professores orientadores Carlos Aurélio e Leonardo Resende e aos professores Charles C. Cavalcante, João M. T. Romano, José C. Bermudez, Márcio H. Costa e Bartolomeu Uchôa-Filho, que participaram da banca e contribuíram para eliminar erros e imprecisões no documento original.*

*Aos meus colegas do GPqCom que participaram do dia-a-dia deste e de outros projetos: Andrei Legg, Bruno Chang, Cesar Vidal, Iuri Baran, João Rebelatto, Guilherme Marra, Gustavo Lima, Mário de N. Neto, Renato Machado, Ricardo Seiti, Ricardo Trentin, Roberto Nóbrega e Wilson Lopez.*

*Também não posso esquecer dos meus colegas professores do CEFET-Ce por me concederem mais de quatro anos de afastamento e algum auxílio financeiro nos momentos mais críticos (antes de começar a receber a bolsa de doutorado).*

*O meu agradecimento “institucional” ao CEFET-Ce, à CAPES pelo auxílio material e à UFSC pelos recursos que possibilitaram a realização deste trabalho.*

*Francisco José Alves de Aquino*

Resumo da tese apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Doutor em Engenharia Elétrica.

# PROCESSAMENTO LARGAMENTE LINEAR APLICADO AO PROBLEMA DE EQUALIZAÇÃO DE CANAL DE COMUNICAÇÃO DIGITAL

**Francisco José Alves de Aquino**

Setembro/2008

**Orientador:** Carlos Aurélio Faria da Rocha, Dr.

**Co-orientador:** Leonardo Silva Resende, Dr.

**Área de Concentração:** comunicações e processamento de sinais.

**Palavras-chave:** circularidade, processamento largamente linear, filtro de erro de predição largamente linear, equalização cega, algoritmo LMS, interferência entre símbolos.

**Número de Páginas:** 199.

## RESUMO:

Esta tese aborda o desenvolvimento e a implementação de técnicas de processamento largamente linear (LL) aplicadas aos problemas de equalização treinada e equalização cega do canal de comunicação. O processamento largamente linear vem se mostrando uma técnica de grande interesse de pesquisa, pois, sob algumas condições, leva a um desempenho muito superior às técnicas lineares convencionais de equalização, predição, formatação de feixe entre outras aplicações, com custo computacional reduzido.

Inicialmente, este documento apresenta a fundamentação teórica do processamento largamente linear onde revisamos os conceitos de sinal impróprio, ruído duplamente branco, circularidade e outros. Em seguida, com base nestes conceitos, apresentamos exemplos de sinais reais e complexos que são não-circulares onde o processamento largamente linear pode, portanto, ser aplicado vantajosamente.

Investigamos as propriedades dos equalizadores largamente lineares cujo desempenho é superior ao dos equalizadores lineares, tanto em relação à complexidade computacional quanto à compensação dos efeitos do canal. Nesse estudo, incluímos o cálculo do atraso ótimo de equalização e aliamos a técnica *multi-split* aos equalizadores largamente lineares com o objetivo de obter uma maior taxa de convergência e um menor erro de convergência sem elevar muito o custo computacional.

Um dos principais resultados obtidos é o desenvolvimento de um novo equalizador cego baseado em um filtro de erro de predição largamente linear (FEPLL). Este novo equalizador, em contraste com a equalização usando um filtro de erro de predição linear (FEPL), é capaz de equalizar canais de fase não-mínima, inclusive aqueles com nulos espectrais. Adaptamos os algoritmos LMS (com passo fixo ou variável) e RLS para o uso com este equalizador LL. Usando um conjunto de FEPLL obtivemos uma solução de equalização cega mais robusta e de melhor desempenho que a conseguida por um único FEPLL.

Parte do trabalho teórico desenvolvido (equalizador largamente linear e FEPLL de passo fixo) foi implementado em FPGA. Todas as propostas desta tese foram validadas por meio de simulações de Monte Carlo.

Abstract of Ph.D. dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

# **WIDELY LINEAR PROCESSING APPLIED TO EQUALIZATION PROBLEM OF THE DIGITAL COMMUNICATION CHANNEL**

**Francisco José Alves de Aquino**

September/2008

**Advisor:** Carlos Aurélio Faria da Rocha, Dr.

**Co- Advisor:** Leonardo Silva Resende, Dr.

**Area of Concentration:** Communications and Signal Processing.

**Keywords:** widely linear processing, circularity, equalization, blind equalization, widely linear prediction error filter, LMS algorithm, intersymbol interference (ISI).

**Number of Pages:** 199.

## **ABSTRACT:**

This Ph.D. dissertation is about the development and implementation of widely linear processing techniques applied to trained and blind equalization of a communication channel. Widely linear processing is an active research topic because, under certain conditions, it can perform much better than conventional techniques for equalization, prediction, beamforming and other applications with a reduced computational load.

Initially, this document presents a theoretical base for widely linear processing, where we revise concepts such as improper signals, doubly white noise, circularity, among others. Following, we present examples of real and complex signals that are non-circular, where widely linear processing can be applied with advantages.

We investigate the properties of widely linear equalizers, which performance is better than the one from linear equalizers, both the computational complexity and channel effects compensation. In this study, we include the calculation of the optimal delay for equalization, and we use the multi-split technique with widely linear equalizers to obtain both a better convergence rate and a smaller convergence error without a great increase in computational load.

One of the main obtained results is the development of a new blind equalizer based on a widely linear error prediction filter (WLEPF). This new equalizer, in contrast to equalization using a linear error prediction filter (LEPF), is capable of equalizing non-minimal phase channels, including channels with spectral nulls. We adapt the LMS (with fixed or variable step-size) and RLS algorithms for use with this equalizer. Using a set of WLEPFs, we can reach a solution for blind equalization that is more robust and has better performance than the one with a single WLEPF.

Part of the theoretical work presented (widely linear equalizer and fixed step WLEPF) was implemented on a FPGA. All proposals in this thesis were validated using Monte Carlo simulations.



# Sumário

<b>Sumário .....</b>	<b>ix</b>
<b>Lista de Figuras .....</b>	<b>xii</b>
<b>Lista de Tabelas .....</b>	<b>xvi</b>
<b>Lista de Símbolos e Abreviaturas.....</b>	<b>xvii</b>
<b>1 Introdução.....</b>	<b>1</b>
1.1 Sistema de comunicação digital.....	1
1.2 Motivação e processamento largamente linear.....	6
1.3 Canais discretos equivalentes .....	9
1.4 Objetivos.....	12
1.5 Contribuições na forma de artigos publicados.....	14
1.6 Organização da tese.....	17
<b>2 Fundamentação Teórica .....</b>	<b>19</b>
2.1 Circularidade.....	19
2.1.1 Conceito de circularidade .....	19
2.1.2 Definições de circularidade.....	21
2.2 Função de relação ou pseudo-covariância .....	23
2.2.1 Conceito de estacionaridade.....	24
2.2.2 Ruído branco e duplamente branco .....	26
2.3 Processamento largamente linear.....	28
2.3.1 Caso conjuntamente circular .....	30
2.3.2 O vetor de observação é circular .....	30
2.3.3 O sinal transmitido é real .....	30
2.3.4 Detecção e estimação: comparação entre processamento LL e estritamente linear	31
2.4 Exemplos de sinais complexos impróprios.....	31
2.5 Processamento LL e códigos espaço-temporais de bloco (STBC) .....	35
2.5.1 Esquema de Alamouti.....	35
2.5.2 Processamento LL e o esquema de Alamouti .....	37
2.6 Revisão bibliográfica.....	38
2.6.1 Sistemas SISO.....	38
2.6.2 Sistemas MIMO .....	39
2.6.3 Teses consultadas .....	39
2.7 Conclusão do capítulo .....	40
<b>3 Equalização Largamente Linear em Sistemas SISO .....</b>	<b>42</b>
3.1 Equalização linear .....	42
3.2 Equalizador largamente linear .....	46
3.3 Equalizador LL ótimo.....	48
3.4 Ordem mínima de um equalizador LL e atraso ótimo.....	49
3.5 Algoritmo LMS-LL.....	54
3.6 Uma análise estatística do algoritmo LMS-LL.....	56
3.6.1 Comportamento médio do vetor de coeficientes $w$ .....	56
3.6.2 Comportamento do MSE .....	58
3.7 Algoritmo LMS-LL de passo variável .....	60

3.8 Equalizador <i>Multi-Split</i> largamente linear .....	61
3.8.1 Transformada <i>Multi-Split</i> e algoritmo LMS .....	61
3.8.2 Análise da matriz de correlação largamente linear transformada .....	65
3.9 Implementação em FPGA do equalizador largamente linear .....	67
3.9.1 A tecnologia FPGA .....	67
3.9.2 Implementação de filtros FIR em FPGA .....	70
3.10 Resultados de simulação .....	73
3.10.1 Equalização LL ótima .....	73
3.10.2 Equalização adaptativa (LMS-LL) .....	75
3.10.3 Análise estatística: exemplos .....	78
3.10.4 Algoritmo LMS-LL-MS: exemplos .....	81
3.10.5 Implementação em FPGA: comparações e exemplos .....	84
3.11 Conclusão do capítulo .....	86
<b>4 Filtro de Erro de Predição Largamente Linear .....</b>	<b>88</b>
4.1 Predição linear e filtro de erro de predição .....	89
4.1.1. Filtro de erro de predição .....	90
4.1.2. Propriedades do filtro de erro de predição .....	91
4.1.3. Predição linear e modelamento autoregressivo .....	92
4.2 Filtro de erro de predição largamente linear (FEPLL) .....	94
4.2.1 Filtro de erro de predição LL ótimo – critério MSE .....	95
4.2.2 Critério ZF para o filtro de predição LL .....	97
4.3 Análise polifásica do filtro de erro de predição LL .....	99
4.3.1 Equalização ZF para o FEPLL .....	101
4.3.2 FEPLL como um equalizador cego .....	102
4.4 FEPLL adaptativo .....	105
4.5 FEPLL com a convergência acelerada .....	106
4.5.1 FEPLL de passo variável .....	106
4.5.2 FEPLL com algoritmo RLS .....	108
4.6 Equalização com atraso usando um conjunto de FEPLL's .....	110
4.6.1 Predição multi-atraso .....	110
4.6.2 Esquema proposto .....	111
4.7 Implementação do FEPLL em FPGA .....	115
4.8 Resultados de Simulação .....	117
4.8.1 FEPLL ótimos .....	117
4.8.2 FEPLL adaptativo com passo fixo .....	119
4.8.3 FEPLL adaptativo com passo variável .....	123
4.8.4 FEPLL usando o algoritmo RLS .....	128
4.8.5 Equalização com atraso usando um conjunto de FEPLL's .....	129
4.9 Conclusão do capítulo .....	131
<b>5 Conclusões e Trabalhos Futuros .....</b>	<b>132</b>
5.1 Contribuições desta tese .....	132
5.2 Continuidade do trabalho .....	133
<b>Apêndice A – Complementos do Capítulo 2 .....</b>	<b>135</b>
A.1 Definições iniciais .....	135
A.2 Variável aleatória discreta .....	137
A.3 Covariância e <i>pseudo-covariância</i> .....	137
A.4 Comparação de desempenho entre o processamento linear e largamente linear para os casos de estimação e detecção .....	139

A.4.1 Definições necessárias para a comparação .....	140
A.4.2 Ganho na estimação .....	142
A.4.3 Ganho na detecção .....	143
<b>Apêndice B – Complementos do Capítulo 3.....</b>	<b>145</b>
B.1 Sistema de equações (3.6) e atraso ótimo .....	145
B.1.1 Dedução do sistema de equações (3.6) .....	145
B.1.2 Atraso ótimo .....	147
B.2 Comparação das estimativas de variância de um sinal aleatório usando as equações (3.53) e (3.58) e outras estimativas .....	150
B.3 Número de coeficientes <i>versus</i> $J_{min}$ para o equalizador LL .....	152
B.4 Efeito da latência: algoritmo DLMS.....	156
<b>Apêndice C – Complementos do Capítulo 4 .....</b>	<b>159</b>
C.1 Algoritmos cegos.....	159
C.1.1 Algoritmo de Sato .....	160
C.1.2 Algoritmos de Godard e CMA .....	161
C.1.3 Algoritmo BGR (Benveniste-Goursat-Ruget).....	161
C.1.4 Algoritmo Benveniste-Goursat.....	162
C.1.5 Algoritmo “Stop-and-Go” .....	162
C.1.6 Outros algoritmos cegos.....	163
C.1.7 Modo de decisão direta (DD) .....	163
C.2 Comparação das soluções ZF e MSE para o FEPLL.....	165
C.3 Detalhamento da implementação do FEPLL com passo fixo .....	167
<b>Referências Bibliográficas .....</b>	<b>170</b>
<b>Nota biográfica.....</b>	<b>181</b>

# Lista de Figuras

Figura 1.1: Modelo de um sistema comunicação digital.....	2
Figura 1.2: Exemplos de constelações, todas com energia média unitária. ....	3
Figura 1.3: Meio de propagação com múltiplos percursos (canal seletivo em frequência). ....	4
Figura 1.4: Comparação de taxas de erros de bits em uma transmissão com e sem código. ....	6
Figura 1.5: Modelo simplificado de um sistema de comunicação digital.....	6
Figura 1.6: Exemplo de resposta ao impulso de um canal seletivo em frequência e variante no tempo devido ao efeito dos múltiplos percursos.....	13
Figura 1.7: “Fotografia instantânea” das magnitudes dos espectros correspondentes às respostas ao impulso do canal mostrado na Figura 1.6. ....	13
Figura 2.1. Exemplo de um sinal próprio e de um sinal impróprio. ....	21
Figura 2.2. Transformada de Fourier (a) da função de relação do sinal complexo $s[k]$ , (b) da covariância do sinal complexo $s[k]$ e (c) da covariância do sinal real $p[k]$ . ....	25
Figura 2.3: Amplitudes (a) covariância do sinal $w[k]$ , (b) função de relação de $w[k]$ , (c) covariância de $z_{db}[k]$ , (d) função de relação de $z_{db}[k]$ . ....	27
Figura 2.4: (a) Diagrama de blocos de um transmissor OQAM, (b) sinais $I$ e $Q$ defasados de $T_b$ , (c) transição entre os símbolos de uma constelação OQAM de quatro pontos, (d) constelação 16-OQAM mostrando as transições, (e) constelação 16-QAM. ....	33
Figura 2.5. Esquema Alamouti: diversidade na transmissão.....	36
Figura 3.1. Exemplo dos efeitos de um canal de sobre uma constelação 4-QAM e sua equalização: (a) constelação transmitida, (b) constelação na saída do canal, (c) constelação equalizada. ....	43
Figura 3.2: Diagrama de blocos (a) equalizador transversal (b) DFE. ....	45
Figura 3.3: Modelo LL para filtragem inversa (equalização).....	47
Figura 3.4: Modelo multicanal (polifásico) equivalente. ....	47
Figura 3.5: (a) Filtro transversal, (b) Filtragem <i>multi-split</i> , (c) Estrutura em borboleta para uma implementação <i>multi-split</i> eficiente.....	63
Figura 3.6: Desempenho exigido em função do tempo, adaptado de XILINX Inc. (2005b)...	68
Figura 3.7: Exemplo de um projeto em FPGA usando <i>System Generator</i> em ambiente Simulink.....	69
Figura 3.8: Fluxo de projeto em <i>Xilinx System Generator</i> , adaptado de XILINX Inc. (2005b). ....	69
Figura 3.9: Comparação das estruturas dos filtros em relação à taxa de amostragem e número de coeficientes, adaptado de XILINX Inc. (2005b). ....	70
Figura 3.10 Demanda de recursos com número de coeficientes para as estruturas (a) seqüencial e (b) semi-paralela.....	72
Figura 3.11. Canal simulado: (a) zeros, (b) resposta em frequência, (c) constelação após o canal, (d) constelação equalizada. ....	75
Figura 3.12: Variância do erro em função de $k_0$ . ....	75
Figura 3.13: Comparação das curvas de aprendizado dos algoritmos LMS e LMS-LL. (a) SNR de 25 dB, (b) SNR de 30 dB, (c) SNR de 35 dB, (d) SNR de 40 dB. ....	76
Figura 3.14. Canal do Exemplo 2: (a) zeros, (b) resposta em frequência. ....	77
Figura 3.15: Constelação 16-OQAM mostrando as transições (a) antes do canal, (b) equalizada. Relação sinal-ruído de 30 dB. ....	78
Figura 3.16: Curva de aprendizado do Exemplo 2. ....	78
Figura 3.17: Comportamento do MSE, Exemplo 1. ....	79

Figura 3.18 Comportamento teórico e simulado (média de apenas 3 realizações) para os coeficientes do filtro $\mathbf{f}$ , Exemplo 1.....	80
Figura 3.19 Comportamento do MSE, Exemplo 2.....	80
Figura 3.20 Comportamento teórico e simulado (média de apenas 3 realizações) para os coeficientes do filtro $\mathbf{f}$ , Exemplo 2.....	81
Figura 3.21 Canal $\mathbf{h}_1$ (a) zeros, (b) resposta em frequência, (c) saída do canal, (d) constelação equalizada (igual para os três equalizadores).....	82
Figura 3.22 Curvas de aprendizado para o canal $\mathbf{h}_1$ .....	82
Figura 2.23 Canal $\mathbf{h}_2$ : (a) zeros, (b) resposta em frequência, (c) sinal após o canal.....	83
Figura 3.24 Curvas de aprendizado para o canal $\mathbf{h}_2$ .....	83
Figura 3.25 Comparação das estruturas paralela, semi-paralela e seqüencial: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado.....	84
Figura 3.26 Simulação I: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado dos algoritmos para o canal A.....	85
Figura 3.27 Simulação II: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado dos algoritmos para o canal B.....	86
Figura 4.1: (a) Preditor linear e (b) filtro de erro de predição linear (FEPL).....	91
Figura 4.2: (a) Síntese de um processo AR com um filtro IIR, (b) análise de um processo AR usando um filtro de predição.....	94
Figura 4.3: Filtro de erro de predição largamente linear (FEPLL).....	94
Figura 4.4. Equivalência entre o filtro de predição LL e um sistema polifásico: (a) sistema SISO mais operação de conjugação, (b) sistema SIMO equivalente, (c) filtro de erro de predição LL na forma polifásica equivalente.....	99
Figura 4.5. Esquema de equalização cega usando o filtro de predição LL.....	106
Figura 4.6: Diagrama de blocos dos dois primeiros estágios do esquema de equalização robusto proposto usando FEPLL adaptativo.....	112
Figura 4.7: Diagrama de blocos completo do esquema de equalização proposto.....	114
Figura 4.8. Implementação em FPGA: detalhe do ajuste do ganho.....	115
Figura 4.9. Implementação do FEPLL em FPGA: (a) detalhe do cálculo do erro, (b) visão geral do FEPLL, (c) detalhamento de um dos filtros transversais.....	116
Figura 4.10. Equalização com filtro de predição largamente linear (critério ZF), sem compensação do ganho complexo do canal. a) Sinal transmitido, b) constelação após o canal, c) resposta em frequência do canal, d) sinal equalizado.....	118
Figura 4.11. Equalização usando o critério MSE, com compensação do ganho complexo introduzido pelo canal: a) zeros do canal, b) constelação após o canal, c) resposta em frequência do canal, d) sinal equalizado. Relação sinal-ruído de 30 dB.....	119
Figura 4.12. Equalização cega – (a) diagrama de olho após a equalização, (b) evolução da constelação, (c) compensações de fase e (d) de ganho. Relação sinal-ruído de 30 dB..	120
Figura 4.13. Equalização cega: (a) diagrama de olho após a equalização para o canal definido por Porat e Friedlander, (b) evolução da constelação. Relação sinal-ruído de 40 dB....	120
Figura 4.13. Diagrama de olho de uma realização para as diversas técnicas cegas para o canal $\mathbf{h}_m$ e uma relação sinal-ruído de 30 dB: (a) FEPLL, (b) CMA, (c) FEPL, (d) Sato, (e) Benveniste-Goursat.....	122
Figura 4.14: Comparação dos algoritmos cegos: MSE do erro de decisão direta. Relação sinal-ruído de 30 dB.....	123
Figura 4.15. Equalização do canal $\mathbf{h}_1$ com FEPLL de passo variável: (a) constelação após o canal, (b) sinal após o equalizador (últimas 500 amostras), (c) evolução da constelação na saída do equalizador, (d) passos de adaptação em função no tempo.....	124

Figura 4.16. Equalização do canal $\mathbf{h}_2$ : (a) constelação após o canal, (b) evolução do sinal na saída do FEPLL, (c) ajuste de fase em função do tempo, (d) ajuste de ganho, (e) últimos 500 símbolos equalizados, (f) evolução dos passos de adaptação. ....	125
Figura 4.17. Diagramas de olho para o caso de variação brusca do canal, (b) evolução da fase, (c) evolução da correção do ganho em função do número de símbolos recebidos. ....	126
Figura 4.18. IES residual para os canais $\mathbf{h}_1$ e $\mathbf{h}_2$ , mudança brusca de $\mathbf{h}_1$ para $\mathbf{h}_2$ na amostra de número 5000. Relação sinal-ruído de 30 dB. ....	126
Figura 4.19. Diagramas de olho: (a) FEPLL de passo variável, (b) FEPLL de passo fixo, (c) equalizador de Godard, (d) valor médio do erro de decisão quadrático direto. ....	127
Figura 4.20. Equalização usando o algoritmo RLS, canal de Porat: (a) constelação após o canal, (b) sinal equalizado, (c) ganho e fase, (d) evolução da constelação após o equalizador. ....	128
Figura 4.21. Equalização usando o algoritmo RLS, canal com nulo espectral. ....	129
Figura 4.22. Variância do erro (em dB) na saída do equalizador LL treinado em função do atraso de equalização para o Canal 1. ....	130
Figura 4.23. Taxa de erro de símbolo (TES) para os canais simulados. ....	131
Figura A.1: Ganho do processamento LL sobre o linear para o caso de estimação. ....	144
Figura A.2: Ganho do processamento LL sobre o linear para o caso de detecção. ....	144
Figura B.1: Atraso ótimo para equalizadores LL (ZF e MSE) com 10 coeficientes em função da relação sinal-ruído para o canal $\mathbf{h}$ . ....	149
Figura B.2: Atraso ótimo para equalizadores LL (ZF e MSE) com 12 coeficientes em função da relação sinal-ruído para o canal $\mathbf{h}$ . ....	149
Figura B.3. Comparação entre as equações (3.53) e (3.58), com $s_i[k]$ inicializado com zero e $\gamma = 0,99$ . Sinal aleatório de média nula e variância unitária. ....	150
Figura B.4. Comparação entre as equações (3.53) e (3.58), com $s_i[k]$ inicializado com um e $\gamma = 0,99$ . Sinal aleatório de média nula e variância unitária. ....	150
Figura B.5: Evolução do erro de estimação $\mathbf{\epsilon}_R$ da matriz de correlação. ....	151
Figura B.6: Características dos canais. (a) Zeros do canal de fase mista, (b) zeros do canal de fase mínima, (c) zeros do canal de fase máxima, (d) resposta em frequência do canal de fase mista, (d) resposta em frequência do canal de fase mínima, (e) resposta em frequência do canal de fase máxima. ....	154
Figura B.7: Atraso ótimo e MMSE para o canal de fase mínima, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE função do número de coeficientes para o equalizador LL. ....	154
Figura B.8: Atraso ótimo e MMSE para o canal de fase mista, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE função do número de coeficientes para o equalizador LL. ....	155
Figura B.9: Atraso ótimo e MMSE para o canal de fase máxima, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE função do número de coeficientes para o equalizador LL. ....	155
Figura B.10: Equalizador FPGA de apenas um coeficiente, todos os multiplicadores com latência $D = 3$ . ....	157
Figura B.11: Efeito da latência e do passo no comportamento do algoritmo DLMS, (a) latência de apenas uma amostra, (b) latência de duas amostras, (c) latência de três amostras e (d) latência de quatro amostras. ....	158
Figura C.1 Equalizador cego e seus modos de operação. ....	159
Figura C.2. Diagrama de olho de uma realização para as diversas técnicas cegas + DD para o canal $\mathbf{h}_m$ e uma relação sinal-ruído de 30 dB: (a) FEPLL, (b) CMA, (c) FEPL, (d) Sato, (e) Benveniste-Goursat. ....	164

Figura C.3. Comparação dos algoritmos cegos: MSE do erro de decisão direta. ....	164
Figura C.4: Evolução dos coeficientes do filtro $\mathbf{f}_{\text{MSE}}$ em função da relação sinal-ruído (de 0 a 80 dB), (a) coeficiente $f_0$ , (b) coeficiente $f_1$ e (c) coeficiente $f_2$ .....	166
Figura C.4: Evolução dos coeficientes do filtro $\mathbf{g}_{\text{MSE}}$ em função da relação sinal-ruído (de 0 a 80 dB), (a) coeficiente $g_0$ , (b) coeficiente $g_1$ e (c) coeficiente $g_2$ .....	166
Figura C.6: Detalhamento do cálculo de um coeficiente em FPGA do FEPLL (“tap”).....	167
Figura C.7: Conjunto de “taps” e somadores para cálculo parcial da saída do FEPLL.....	168
Figura C.8: (a) correção de fase e (b) cálculo do símbolo decidido em FPGA.....	168
Figura C.9: Estimação de área do FEPLL.....	169
Figura C.10: Equalização com o FEPLL implementado em FPGA. (a) Sinal após o canal, (b) ajuste de ganho e de fase, (c) evolução da constelação, (d) diagrama de olho após a convergência. Relação sinal-ruído de 30 dB. ....	169

# Lista de Tabelas

Tabela 2.1: Esquema de codificação Alamouti .....	36
Tabela 3.1. Modelo LMS Largamente Linear .....	60
Tabela 3.2: Algoritmo LMS-LL-MS .....	65
Tabela 3.3: Taxa de amostragem para as estruturas FIR .....	71
Tabela 3.4: Estimação de recursos das estruturas implementadas em FPGA.....	71
Tabela 3.5: Comparação dos equalizadores .....	86
Tabela 4.1: Algoritmo RLS .....	109
Tabela 4.2: Algoritmo RLS para o filtro de erro de predição LL.....	109
Tabela 4.3. Parâmetros dos algoritmos cegos e resultados de simulação .....	122
Tabela 4.4: Parâmetros do FEPLL adaptativo para o canal $\mathbf{h}_1$ .....	123
Tabela 4.5: Parâmetros do FEPLL adaptativo para o canal $\mathbf{h}_2$ .....	124
Tabela 4.6. Parâmetros dos equalizadores para o canal $\mathbf{h}_3$ .....	127
Tabela 4.7: Coeficientes dos canais 1, 2 e 3 para FEPLL com atraso de equalização .....	130
Tabela B.1: Fator de diagonalização do equalizador LL-MS em função de $N$ .....	152



# Lista de Símbolos e Abreviaturas

## Símbolos

$\star$	– operação de convolução.
$a[\cdot]$	– símbolo de uma constelação real ou complexa.
$\tilde{a}[\cdot]$	– símbolo estimado.
$\hat{a}[\cdot]$	– símbolo decidido.
$E\{\cdot\}$	– operador esperança matemática, valor esperado.
$\det(\cdot)$	– determinante de uma matriz.
$\eta$	– ruído branco gaussiano.
$\boldsymbol{\eta}$	– vetor de ruído.
$\sigma_a^2$	– variância do símbolo (amostra) discreto.
$\sigma_\eta^2$	– variância do ruído.
$j$	– número complexo $j = \sqrt{-1}$ .
$\mathbf{h}$	– canal discreto (vetor coluna)
$\mathbf{I}$	– matriz identidade
$k_0$ ou $\Delta$	– atraso de $k_0$ ou $\Delta$ amostras.
$r[\cdot]$	– sinal recebido.
$\Re\{x\}$	– parte real de um valor complexo.
$\Im\{x\}$	– parte imaginária de um valor complexo.
$\mathbb{R}$	– conjunto dos números reais.
$\mathbb{C}$	– conjunto dos números complexos.
$\text{tr}(\mathbf{A})$	– traço (soma dos elementos da diagonal principal) da matriz $\mathbf{A}$ .
$\Psi(\cdot)$	– quantizador, dispositivo de decisão.
$x^*$	– conjugado complexo de $x$ .
$ x $	– módulo de um valor complexo ou de uma variável aleatória complexa.
$\ \mathbf{x}\ $	– norma euclidiana de um vetor real ou complexo.
$\ \mathbf{X}\ _F$	– norma de Frobenius de uma matriz real ou complexa.
$\langle \mathbf{x}, \mathbf{y} \rangle$	– produto interno entre dois vetores.
$\left\lfloor \frac{x}{2} \right\rfloor = \begin{cases} \frac{x}{2}, & x \text{ par} \\ \frac{x+1}{2}, & x \text{ ímpar} \end{cases}$	– menor inteiro igual ou maior que $x/2$ .
$\mathbf{X}^+$	– pseudo-inversa de uma matriz (não-quadrada).
$\mathbf{X}^*$	– conjugado complexo de uma matriz.
$\mathbf{X}^H$	– Hermitiana de uma matriz (operações de transposição e conjugação).
$\mathbf{X}^T$	– transposição de uma matriz.

## Abreviaturas

AR	– Autoregressivo (modelo)
AWGN	– Additive White Gaussian Noise.
BER	– Bit Error Rate.
BPSK	– Binary Phase Shift Keying.
CGA	– Controle Automático de Ganho.
CDMA	– Code Division Multiple Access.
CMA	– Constant Modulus Algorithm.
CRU	– Círculo de Raio Unitário.
CSI	– Channel State Information.
CT	– Continuous Time.
DFE	– Decision Feedback Equalization.
DS-CDMA	– Direct Sequence Code Division Multiple Access.
DT	– Discrete Time.
EOS	– Estatísticas de Ordem Superior.
FFT	– Fast Fourier Transform.
FIR	– Finite Impulse Response.
FSE	– Fractionally Spaced Equalizer.
FPGA	– Field Programmable Gate Arrays
GMSK	– Gaussian Minimum-Shift Keying.
GSM	– Global System for Mobile Communication.
HOS	– High Order Statistics.
i.i.d.	– Independente e Identicamente Distribuída.
IES	– Interferência Entre Símbolos.
IF	– Intermediate Frequency
IFFT	– Inverse Fast Fourier Transform.
IIR	– Infinite Impulse Response.
ISI	– Intersymbol Interference.
LL	– Largamente Linear.
LMS	– Least Mean Square.
LOS	– Line of Sight.
MIMO	– Multiple Input, Multiple Output.
MISO	– Multiple Input, Single Output.
MMSE	– Minimum Mean Square Error.
MSE	– Mean Square Error.
MSK	– Minimum Shift Keying.
NLOS	– Non Line of Sight.
OQAM	– Offset Quadrature Amplitude Modulation.
PAM	– Pulse Amplitude Modulation.
PDF	– Probability Distribution Function.
QAM	– Quadrature Amplitude Modulation.
QPSK	– Quadrature Phase Shift Keying.
RDS	– Rádio Definido por Software.
RF	– Rádio Frequência.
RLS	– Recursive Least Square.
RS	– Reed-Solomon.
Rx	– Receptor.

SDR	– Software Defined Radio.
SER	– Symbol Error Rate.
SIMO	– Single Input, Multiple Output.
SISO	– Single Input, Single Output.
SNR	– Signal-to-Noise Ratio.
SOS	– Second-Order Stationary.
STBC	– Space Time Block Code.
$T_c$	– Taxa de Código.
TEB	– Taxa de Erro de Bit.
$T_s$	– Tempo de símbolo.
Tx	– Transmissor.
v.a.	– variável aleatória.
WL	– Widely Linear (Largamente Linear).
WLAN	– Wireless LAN (Local Area Network).
WLP	– Widely Linear Processing.
WSS	– Wide Sense Stationary.
ZF	– Zero Forcing.

# Capítulo 1

---

## Introdução

Neste capítulo introdutório, apresentamos alguns conceitos fundamentais, a estrutura e os objetivos da presente tese, com o intuito de familiarizar o leitor com os principais aspectos teóricos e práticos deste trabalho. Estes conceitos serão explorados e aprofundados ao longo da tese.

Este capítulo é organizado como segue. Na Seção 1.1, apresentamos os fundamentos de um sistema de comunicação digital e alguns conceitos relacionados diretamente a esse tópico. Na Seção 1.2, definimos o que é processamento largamente linear e apresentamos a motivação básica para esta tese. A modelagem para o canal de comunicação digital e a notação que utilizaremos neste documento são apresentados na Seção 1.3. Na Seção 1.4, detalhamos os objetivos gerais e específicos deste trabalho. As nossas principais contribuições, na forma de artigos, decorrentes deste trabalho de pesquisa são listadas na Seção 1.5. Finalmente, na Seção 1.6, apresentamos a organização geral desta tese, incluindo um rápido comentário sobre cada um dos capítulos que a compõem.

### 1.1 Sistema de comunicação digital

Durante todo o século XX, o enorme progresso tecnológico nas áreas de telecomunicações e eletrônica fez com que os sinais transmitidos migrassem, em grande parte, do mundo analógico para o digital. Curiosamente, o primeiro sistema de comunicação usando sinais elétricos, a telegrafia<sup>1</sup>, era digital (Proakis, 1995; Carlson, 1986).

Uma mensagem digital pode ser definida como sendo uma seqüência de símbolos selecionados a partir de um conjunto finito de elementos discretos (“alfabeto”). Como a informação de uma mensagem digital está concentrada em símbolos discretos, um sistema digital de comunicação deveria entregar esses símbolos, com um grau especificado de

---

<sup>1</sup> A primeira demonstração pública do telégrafo elétrico, desenvolvido pelo norte-americano Samuel F. B. Morse (1791-1872), ocorreu em 1837. Na Inglaterra, W. Cooke e C. Wheatstone iniciaram a operação comercial do telégrafo em 1839.

precisão, no receptor em um intervalo de tempo definido (Carlson, 1986). A Figura 1.1 ilustra um modelo de sistema de comunicação digital (Proakis, 1995), para um modelo de comunicação mais detalhado ver, por exemplo, Sklar (1983). A seguir, comentaremos, em linhas gerais, o modelo ilustrado pela Figura 1.1.

Em um sistema digital, as mensagens produzidas pela fonte de informação, sejam elas digitais ou analógicas, precisam ser convertidas em uma seqüência de dígitos binários. Essa seqüência binária deve conter o menor número possível de dígitos. Assim, a tarefa do codificador de fonte é representar eficientemente a informação original em uma seqüência binária com pouca ou nenhuma redundância. Esse processo é conhecido como codificação de fonte ou compressão de dados. Um exemplo bem conhecido de compressão de dados, sem perdas, é aquela realizada pelo algoritmo de Huffman (Cover e Thomas, 1991).

O codificador de fonte entrega o fluxo de bits gerado ao codificador de canal. Este último acrescenta de forma controlada alguma redundância aos dados recebidos. A aplicação dessa redundância possibilita ao receptor detectar e até mesmo corrigir erros decorrentes do processo de transmissão melhorando, assim, a fidelidade do sinal recebido. Entre os diversos tipos de códigos que permitem a correção de erros podemos citar: os códigos de bloco lineares, os códigos convolucionais, os códigos turbo e os códigos da família LDPC (*Low Density Parity Check*). Um dos primeiros códigos de bloco não trivial<sup>2</sup> que surgiu foi o código de Hamming (Rappaport, 1996).

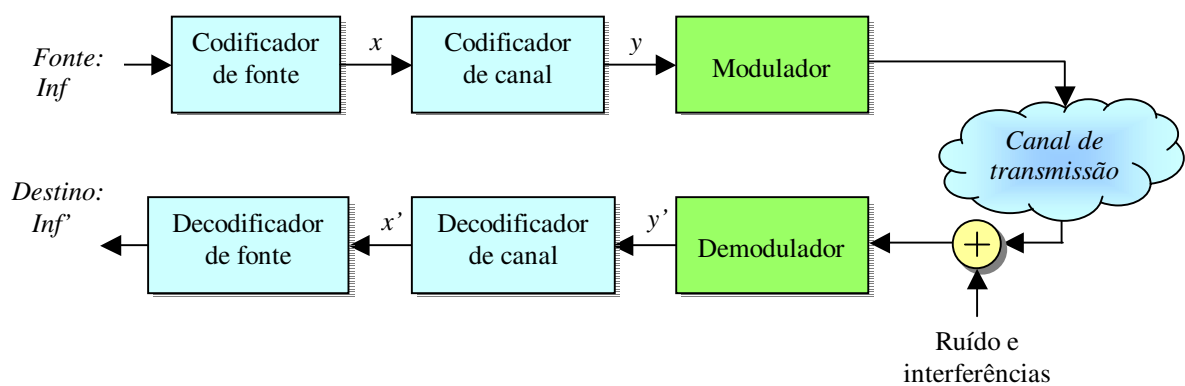


Figura 1.1: Modelo de um sistema comunicação digital.

<sup>2</sup> Um código de repetição é um exemplo de código trivial, por exemplo, o bit 0 é representado pela seqüência 000 para um código de repetição tripla.

A seqüência binária produzida pelo codificador de canal é passada ao modulador digital, que serve como uma interface para o canal de comunicação. Como quase todos os canais de comunicação são capazes de transmitir sinais elétricos, o propósito primário do modulador digital é mapear a seqüência de informação binária em sinais elétricos ou eletromagnéticos. O modulador pode simplesmente mapear os dígitos binários 0's em uma forma de onda analógica,  $s_0(t)$ , e os dígitos 1's em uma outra forma de onda analógica  $s_1(t)$ . Nesse caso, cada bit proveniente do codificador de canal é transmitido separadamente e temos o que é chamado de modulação binária. Alternativamente, o modulador digital pode transmitir  $b$  bits ao mesmo tempo, usando  $M = 2^b$  formas de onda distintas  $s_i(t)$ ,  $i = 0, 1, \dots, M-1$ , e a modulação é chamada de  $M$ -ária. Cada um desses blocos com  $b$  bits forma um *símbolo* que pode ser representado por um valor real ou complexo. Quando os sinais  $s_i(t)$  são mapeados em um conjunto de  $M$  sinais ortonormais em um espaço de dimensão  $N$  ( $N \leq M$ ), o conjunto desses símbolos forma uma *constelação*. A Figura 1.2 exemplifica alguns tipos de constelações comumente usadas. Finalmente, na última etapa do processo de modulação, os sinais  $s_i(t)$  são espectralmente trasladados por uma portadora de radiofrequência (RF). É este sinal de RF que é transmitido através do canal.

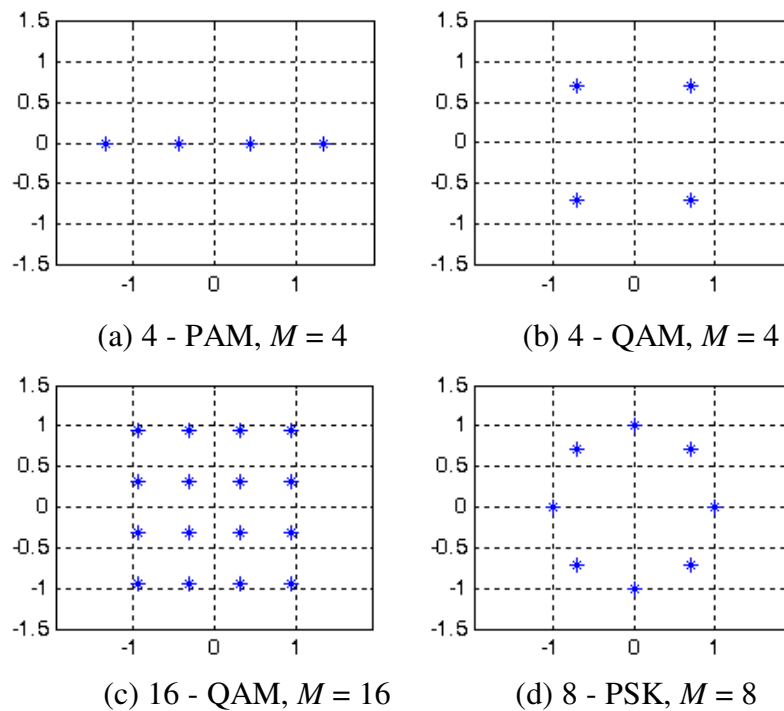


Figura 1.2: Exemplos de constelações, todas com energia média unitária.

O canal de comunicação é o meio físico usado para transmitir o sinal do transmissor ao receptor. Ao passar pelo canal, o sinal transmitido pode ser corrompido de forma aleatória por diversos mecanismos: adição de ruídos, atenuação, seletividade em frequência, deslocamento de fase, que são, em geral, dependentes do tempo. Por exemplo, o efeito de seletividade em frequência pode transformar um único pulso transmitido em um trem de pulsos atrasados no receptor. A duração desse trem de pulsos é chamada de espalhamento de atraso (*delay spread*) (Rappaport, 1996). Em canais de comunicação sem fio (*wireless*), a seletividade em frequência do canal é causada pelo efeito de múltiplos percursos, isto é, o sinal transmitido chega ao receptor por vários caminhos, como ilustra a Figura 1.3. Esse efeito é responsável pelo aparecimento da interferência entre símbolos<sup>3</sup> (IES), o que pode degradar fortemente o sinal recebido. Definiremos formalmente a IES mais adiante, na Seção 1.3.

Agora, analisaremos brevemente o receptor (ver Figura 1.1). No receptor, os passos são executados na ordem inversa à do transmissor, mas tendo que trabalhar com um sinal corrompido pelo canal.

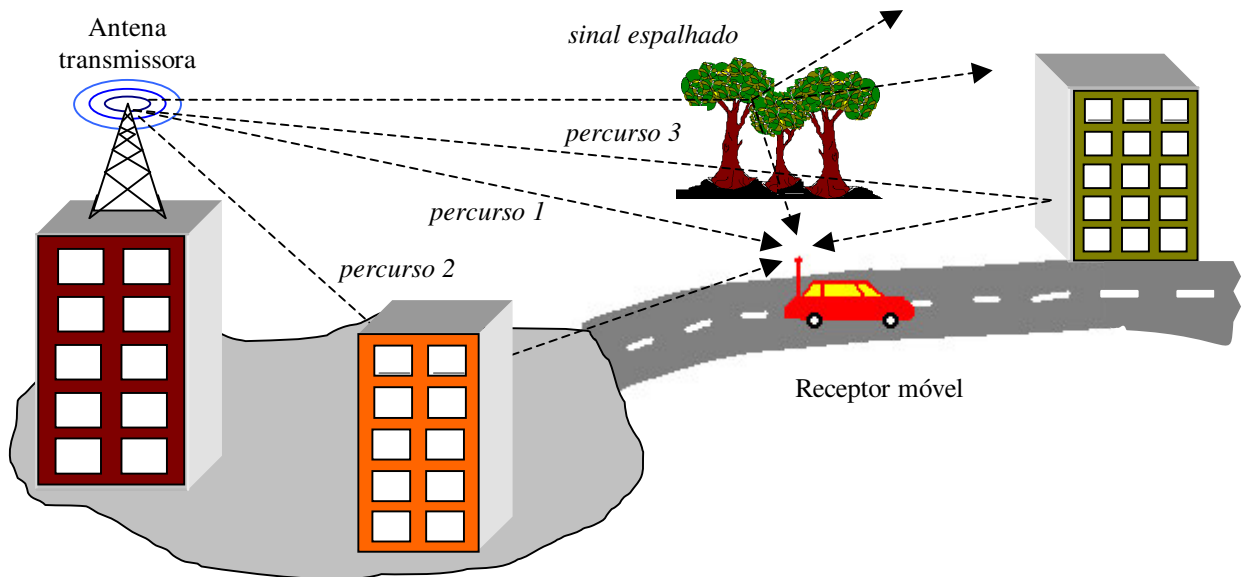


Figura 1.3: Meio de propagação com múltiplos percursos (canal seletivo em frequência).

<sup>3</sup> O termo “intersimbólico”, apesar de uso corrente na literatura técnica brasileira, não é encontrado entre os vocábulos do dicionário Aurélio.

O demodulador converte o sinal recebido em uma representação passa-baixa equivalente, podendo resultar em um sinal complexo discreto no tempo. O demodulador calcula uma estimativa do símbolo transmitido. Essa estimativa é passada ao decodificador de canal.

O decodificador de canal tenta achar a seqüência mais provável de bits de informação, levando em conta o símbolo calculado, a redundância introduzida pelo codificador de canal e o código utilizado. Naturalmente, podem ocorrer erros nesse processo. A probabilidade média de erros de bits na saída do decodificador é uma medida de desempenho da combinação demodulador-decodificador. Em geral, a probabilidade de erro é função das características do código utilizado, das formas de onda empregadas para transmitir a informação, da potência do sinal, da quantidade de ruído, dos métodos de demodulação e decodificação.

Normalmente, o desempenho de um sistema de comunicação digital é expresso em termos de taxa de erro de bit (TEB, em inglês BER – *Bit Error Rate*). Para isso, comparamos os conjuntos de bits transmitidos e recebidos que, idealmente, deveriam ser idênticos. O número de bits que são diferentes nos dois conjuntos é dividido pelo número total de bits para obter a taxa de erro de bit. Naturalmente, essa medida só pode ser realizada de forma controlada em simulações computacionais. Reportando-nos à Figura 1.1, a TEB pode, por exemplo, ser medida antes ou depois do codificador/decodificador de canal. Quando comparamos  $x$  e  $x'$  obtemos a TEB codificada. Comparando  $y$  e  $y'$  podemos calcular a TEB não codificada (“crua”). Nesta tese, usaremos a TEB não codificada, pois a codificação não é objetivo de estudo deste trabalho.

A Figura 1.4 mostra um exemplo da atuação do código Reed-Solomon (exemplo de código de bloco linear) em uma constelação 4-PAM de potência média unitária sobre um canal AWGN (*Additive White Gaussian Noise*). O código Reed-Solomon é designado pelo acrônimo  $RS(N,K)$ , onde  $N$  e  $K$  são os comprimentos do código e da informação, respectivamente. Nessa simulação usamos dois códigos RS, ambos com comprimento  $N = 15$ . O primeiro, com  $K = 5$  símbolos de informação tem uma taxa de código  $T_c = K/N = 1/3$ . Já o segundo, com  $K = 9$  símbolos de informação possui uma taxa de código  $T_c = K/N = 3/5$ . Podemos observar que houve um ganho de código de cerca de 0,90 dB para o RS (15,5) e de 0,97 dB para o RS (15,9), para uma TEB de  $10^{-4}$ . Podemos notar, ainda, que o desempenho do código só se mostra favorável quando a SNR é maior que 12 dB (definido na Equação 1.12).



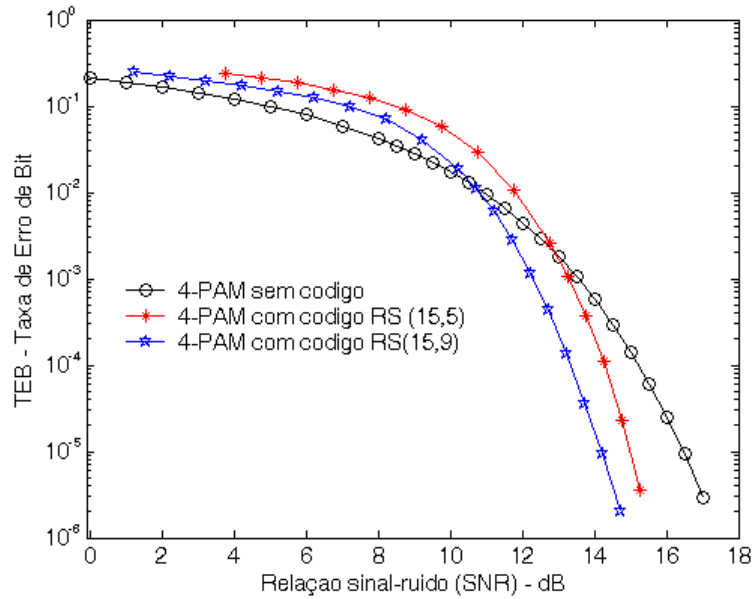


Figura 1.4: Comparação de taxas de erros de bits em uma transmissão com e sem código.

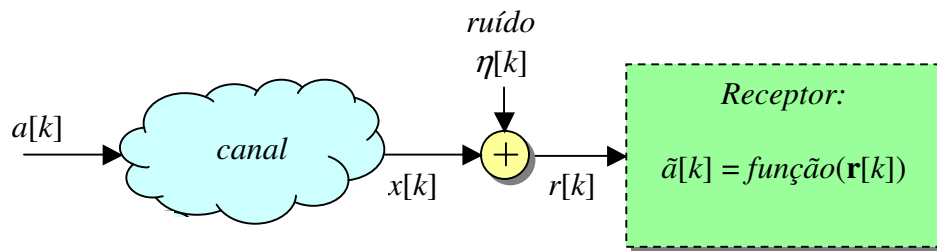


Figura 1.5: Modelo simplificado de um sistema de comunicação digital.

## 1.2 Motivação e processamento largamente linear

Como vimos na seção anterior, o canal de comunicação é um meio muito hostil à transmissão da informação digital. Por isso, é necessário algum processamento digital tanto no transmissor quanto no receptor para combater as distorções introduzidas pelo canal (possivelmente variante no tempo) de comunicação, principalmente a IES. Uma forma possível de realizar esse processamento é, a partir de uma combinação linear dos dados corrompidos  $r[k]$  que chegam ao receptor, encontrar uma estimativa  $\tilde{a}[k]$  do sinal transmitido  $a[k]$  (ver Figura 1.5), a qual é descrita por:

$$\tilde{a}[k] \triangleq \sum_{n=0}^M f_n^* [k] r[k-n] = \mathbf{f}^H [k] \mathbf{r}[k], \tag{1.1}$$

onde  $(\cdot)^*$  indica a operação de conjugação complexa,  $\mathbf{f}[k] = [f_0[k] \ f_1[k] \ \dots \ f_M[k]]^T$  é o vetor de coeficientes do estimador, o sobrescrito  $H$  denota a operação conjunta de transposição e conjugação (hermitiana) e  $\mathbf{r}[k] = [r[k] \ r[k-1] \ \dots \ r[k-M]]^T$  é o vetor dos dados recebidos. O erro de estimação é dado por  $e[k] = d[k] - \tilde{a}[k]$ , onde  $d[k]$  é o sinal desejado. A meta da estimação linear em (1.1) é encontrar um conjunto de coeficientes  $\mathbf{f}[k]$  que, usando algum critério de otimização, minimize o erro de estimação no instante  $k$ .

Um dos critérios de otimização mais utilizado é o MSE (*mean square error* - erro quadrático médio). Esse critério é derivado do método LSE (*least square error*) desenvolvido originalmente pelo matemático e físico Carl Friedrich Gauss (1777-1855), no final do século XVIII, e usado para determinar a órbita do asteroide Ceres, a partir de um número limitado de dados astronômicos, em 1801 (Sayed, 2003).

O objetivo de um estimador usando o critério MSE é minimizar o valor esperado do módulo do erro ao quadrado:  $J \equiv E\{|e[k]|^2\}$ . Este critério é simples e elegante, envolvendo somente estatísticas de segunda ordem. Outras razões para o critério MSE ser muito difundido são (Manolakis *et al.*, 2000):

- é matematicamente tratável;
- é líder no projeto de sistemas práticos;
- pode servir para comparação e avaliação de estimadores projetados a partir de outros critérios de otimização.

Estudos recentes têm mostrado que o *Processamento Largamente Linear*<sup>4</sup> (*Widely Linear Processing* – WLP ou simplesmente processamento LL) pode apresentar um ganho significativo quando comparado às técnicas de estimação linear convencionais. Contudo, esse ganho não ocorre em todas as situações, mas apenas naquelas onde a *pseudo-autocorrelação* da variável aleatória de interesse é não nula (Picinbono e Chevalier, 1995) e que, além disso, esta variável não seja “*duplamente branca*” (Picinbono e Bondon, 1997). Apresentaremos uma fundamentação matemática desses e de outros conceitos no Capítulo 2.

É interessante notar que o processamento LL, não com este nome, foi introduzido primeiramente por Brown e Crane (1969). Nesse artigo seminal, denominado “*Filtragem*

---

<sup>4</sup> Em virtude da sigla “PLL” já ser universalmente usada para “*Phase Locked Loop*”, iremos nos referir ao processamento largamente linear como “processamento LL”. *Optamos* por traduzir o termo em inglês “*widely*” por “*largamente*”, uma outra opção de tradução seria o termo “*amplamente*”.

*Linear Conjugada*”, o enfoque estava restrito à filtragem de um sinal complexo em banda base. Os autores mostraram que a filtragem do sinal combinado com o seu conjugado complexo podia resultar em uma melhora na relação sinal-ruído por um fator de até dois em receptores com filtros casados. Além disso, eles mostraram que esse ganho iria depender da correlação entre as partes real e imaginária do sinal processado.

Picinbono (1995) mostrou formalmente que o processamento LL é benéfico para os problemas de estimação e de predição linear usando o critério MMSE (*Minimum Mean Square Error*). Outros autores, posteriormente, demonstraram que o processamento LL também pode ser usado em formatação de feixe (McWhorte e Schreier, 2003; Chevalier e Blin, 2007), detecção multiusuário e supressão de interferência em sistemas DS-CDMA (*Direct Sequence Code Division Multiple Access*, Gelli *et al.*, 2000), equalização de canais SISO (*single input/single output*) e MIMO (*multiple input/multiple output*), como mostram, por exemplo, os artigos de Gerstacker *et al.* (2003, 2004) e Mattera *et al.* (2005).

É fácil perceber que o interesse no processamento LL não é somente teórico. Por exemplo, Chevalier e Pipon (2006), em um interessante artigo sobre o processamento LL aplicado à recepção de sinais usando um conjunto de antenas, nos informam que em relação à redução de interferência co-canal, o padrão GSM (*Global System for Mobile Communication*) foi o primeiro a adotar a filtragem LL.

Basicamente, no processamento LL, tanto o vetor de dados complexos  $\mathbf{r}$  quanto o seu conjugado  $\mathbf{r}^*$  são utilizados para a estimação ou predição da variável aleatória escalar  $a[k]$ . Dessa forma, a Equação 1.1 é reescrita como:

$$\tilde{a}_{LL}[k] \triangleq \sum_{n=0}^M f_n^*[k] r[k-n] + \sum_{n=0}^M g_n^*[k] r^*[k-n] = \mathbf{f}^H[k] \mathbf{r}[k] + \mathbf{g}^H[k] \mathbf{r}^*[k]. \quad (1.2)$$

Observando (1.2), podemos ter a impressão que o processamento LL tem um custo computacional duas vezes maior do que o processamento estritamente linear, mas isso não é necessariamente verdade. A solução de um sistema LL pode ter uma complexidade computacional praticamente igual ou até mesmo menor que a solução estritamente linear, como podemos ver, por exemplo, em (Schober *et al.*, 2004) para o caso de equalização adaptativa com o uso de um algoritmo do tipo LMS (*Least Mean-Square*).

### 1.3 Canais discretos equivalentes

Ao longo desta tese, modelaremos os canais de comunicação como sendo variantes e discretos no tempo. Consideraremos, ainda, que este modelo discreto é linear e que pode ser representado, no instante discreto  $k$ , por um filtro FIR (*finite impulse response*) de ordem  $L$ :

$$H_k(z) = \sum_{n=0}^L h_n[k] z^{-n}, \quad (1.3)$$

ou, usando notação vetorial, podemos representar o canal por sua resposta impulsiva:

$$\mathbf{h}[k] = [h_0[k] \ h_1[k] \ \dots \ h_L[k]]^T, \quad (1.4)$$

sempre que for conveniente.

Não é difícil perceber que esse modelo para o canal de comunicação é uma simplificação cômoda do caso real. É óbvio que o número total de coeficientes de  $\mathbf{h}[k]$  é uma função dos filtros de transmissão e recepção e do máximo espalhamento de atraso do canal físico. Os filtros de transmissão e recepção possuem, em teoria, resposta ao impulso infinita para manterem a largura de banda de frequência limitada. Portanto,  $\mathbf{h}[k]$  é, necessariamente, um filtro IIR (*infinite impulse response*). Entretanto, na prática, a resposta transitória dos filtros de transmissão e recepção caem rapidamente para algum valor próximo a zero e é razoável supor que somente em uma faixa finita os coeficientes de  $\mathbf{h}[k]$  possuem uma amplitude essencialmente não nula. Então, pela eliminação desses coeficientes que pouco afetam a resposta do canal, este pode ter a sua resposta ao impulso truncada para um filtro FIR, o que justifica as expressões (1.3) e (1.4).

A seguir, apresentamos os tipos de canais mais discutidos na literatura, considerando sempre um modelo de comunicação SISO, como ilustrado na Figura 1.5.

Para tornar mais claro o efeito do canal, examinaremos o sinal recebido  $r[k]$  após a transmissão dos símbolos  $a[k]$ 's pelo canal considerado. Consideramos que os símbolos transmitidos possuem variância  $\sigma_a^2$  e média nula.

*Canal gaussiano* – Este modelo de canal é caracterizado por

$$r[k] = a[k] + \eta[k], \quad (1.5)$$

onde  $\eta[k]$  é uma variável aleatória (possivelmente complexa) com distribuição gaussiana de média nula e variância  $\sigma_\eta^2$ . A suposição de gaussianidade do ruído é devido ao fato deste originar-se de diversas fontes independentes. Por exemplo, uma causa inevitável de ruído elétrico é o movimento randômico de elétrons em meios condutores, tais como resistores e fios condutores. Como este movimento aleatório é realizado por muitos elétrons, o ruído térmico pode ser considerado como um processo estocástico de média nula e distribuição gaussiana. A atividade humana, através do acionamento de equipamentos eletrônicos, e fenômenos naturais também são responsáveis pela geração de ruídos e interferências nos sistemas de comunicação. Um estudo muito interessante contendo aspectos históricos sobre o ruído pode ser encontrado em Okwit (1984).

Nesta tese, a menos que especificado o contrário, consideraremos o ruído como sendo aditivo, branco e com distribuição gaussiana (AWGN – *Additive White Gaussian Noise*). Convém notar que o ruído gaussiano é o que mais reduz a capacidade de canal. Podemos definir a capacidade de canal como sendo a máxima quantidade de bits/segundo que o canal suporta transmitir com uma probabilidade de erro arbitrariamente baixa (Cover e Thomas, 1991).

Além do ruído gaussiano, onipresente em sistemas de comunicações, um outro tipo de ruído também é importante: o ruído impulsivo. O ruído impulsivo é de difícil caracterização estatística, mas apresenta duas propriedades básicas: sua energia é concentrada em curtos intervalos de tempo e é bem maior do que a energia do ruído de fundo (ruído térmico). A duração, a amplitude e o instante de ocorrência do ruído impulsivo são aleatórios (Blackard, *et al.*, 1993). A detecção de sinais na presença de ruído impulsivo exige estratégias diversas daquelas onde a principal interferência é o ruído gaussiano ver, por exemplo, Kuruoglu *et al.* (1998) e Zhidkov (2003).

*Canal com desvanecimento plano* – Um canal com desvanecimento plano preserva as características espectrais do sinal transmitido que chega ao receptor. Historicamente, este tipo de desvanecimento é o mais comumente descrito na literatura (Rappaport, 1996). Tal canal também é conhecido como canal de faixa larga, já que a largura de banda do sinal aplicado é

estreita quando comparada à largura de banda do canal. Para não carregar a notação, omitiremos a dependência temporal do canal. Para esse canal podemos equacionar:

$$r[k] = h_0 a[k], \quad (1.6)$$

onde o coeficiente  $h_0$  é uma variável aleatória gaussiana complexa. Em geral, as distribuições marginais da magnitude e da fase de  $h_0$  são consideradas Rayleigh e uniforme, respectivamente. Para o caso de uma transmissão via rádio, quando existe uma linha de visada direta entre o transmissor e o receptor, a magnitude de  $h_0$  apresenta uma distribuição Riceana. Caso a linha de visada direta desapareça, o sinal recebido é resultado de uma ou mais reflexões e a distribuição de  $h_0$  degenera para uma distribuição do tipo Rayleigh.

*Canal seletivo em frequência* – Neste caso, o sinal recebido pode ser expresso por:

$$r[k] = h_0 a[k] + h_1 a[k-1] + \dots + h_L a[k-L] = \sum_{n=0}^L h_n a[k-n] = \mathbf{h}^T \mathbf{a}[k], \quad (1.7)$$

onde  $\mathbf{a}[k] = [a[k] \ a[k-1] \ \dots \ a[k-L]]^T$  e  $\mathbf{h} = [h_0 \ h_1 \ \dots \ h_L]^T$ . Fica claro que o sinal recebido é uma combinação linear das amostras transmitidas nos instantes anteriores mais a amostra do instante atual, o que corresponde a um canal seletivo em frequência. Usando este modelo, podemos definir a IES como (Rocha, 1996):

$$\text{IES} \triangleq \sum_{n=0, n \neq \delta}^L h_n a[k-n], \quad (1.8)$$

onde o termo  $h_\delta a[k-\delta]$ , não incluído no somatório em (1.8), representa o símbolo desejado a menos de uma atenuação  $h_\delta$ , e  $0 \leq \delta \leq L$  é o atraso causado pelo canal. Uma outra definição que encontramos na literatura para a IES pode ser expressa como (Tong *et al.*, 1993; Souza *et al.*, 2002):

$$\text{IES} \triangleq \frac{\sum_{n=0}^L |h_n|^2 - \max_n |h_n|^2}{\max_n |h_n|^2} = \frac{\sum_{n=0}^L |h_n|^2}{\max_n |h_n|^2} - 1. \quad (1.9)$$

Os coeficientes  $h_n$  são modelados como variáveis aleatórias gaussianas complexas com alguma distribuição de amplitude especificada em geral, Rayleigh ou Rice. Como os

coeficientes  $h_n$  variam com o tempo, a energia recebida por símbolo pode ser diferente de um instante de amostragem a outro. Entretanto, em muitas situações práticas, esse canal pode ser considerado aproximadamente invariante durante a transmissão de um bloco de símbolos. Esta hipótese é razoável dado que a duração  $T_s$  de um símbolo é pequena quando comparada com a taxa de mudança do canal (Tarokh *et al.*, 1998).

Cabe observar que ao longo desta tese estaremos considerando que existe um sincronismo perfeito entre o transmissor e receptor, a menos que indicado o contrário.

Um conceito também muito importante e habitualmente usado em medidas de desempenho de sistemas de comunicação é a relação sinal-ruído (em inglês, *signal to noise ratio* – SNR), que podemos equacionar como:

$$SNR_{dB} = 10 \log_{10} \left( \sigma_a^2 \sum_{n=0}^L |h_n|^2 / \sigma_\eta^2 \right). \quad (1.10)$$

Para ilustrar melhor os conceitos dessa seção, veremos a seguir um exemplo simples de um canal seletivo em frequência e variante no tempo.

A Figura 1.6 mostra um exemplo de resposta ao impulso de um canal discreto, seletivo em frequência e variante no tempo, com uma distribuição do tipo Rayleigh. Neste exemplo, o número de múltiplos percursos é igual a sete. Os gráficos mostram o módulo da resposta ao impulso do canal em quatro instantes sucessivos de tempo. Essa é uma representação simplificada, já que estamos considerando, por exemplo, que o atraso de cada percurso é constante. Já a Figura 1.7 apresenta as magnitudes dos espectros correspondentes às respostas ao impulso indicadas na Figura 1.6.

## 1.4 Objetivos

Esta tese tem como objetivos principais a análise e a proposição de técnicas de processamento largamente linear para combater os efeitos indesejáveis introduzidos por um canal de comunicações de dados. A partir desse duplo objetivo, podemos relacionar os seguintes tópicos ou objetivos específicos:

- estudo aprofundado do processamento largamente linear aplicado principalmente a sistemas SISO;
- mostrar quando a técnica de processamento LL pode ser aplicada vantajosamente em relação ao processamento estritamente linear;

- obtenção de estruturas computacionalmente eficientes, que façam uso do processamento LL para a equalização do canal de comunicação.

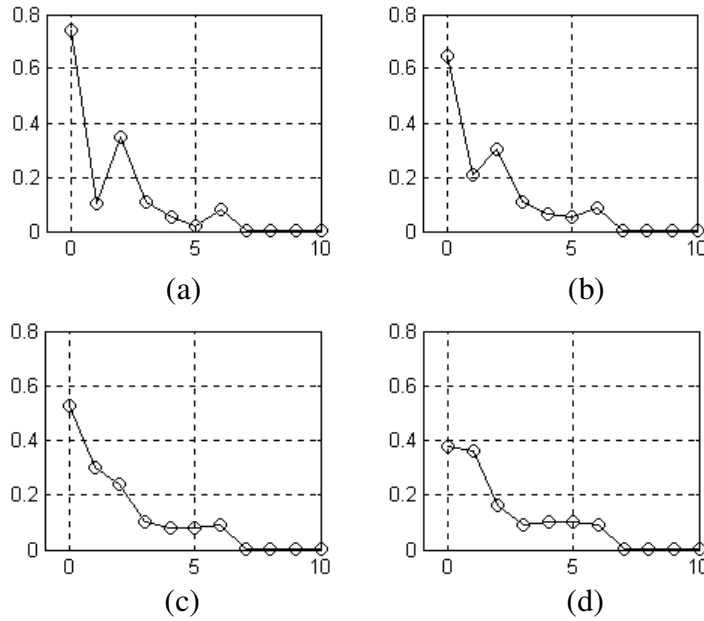


Figura 1.6: Exemplo de resposta ao impulso de um canal seletivo em frequência e variante no tempo devido ao efeito dos múltiplos percursos.

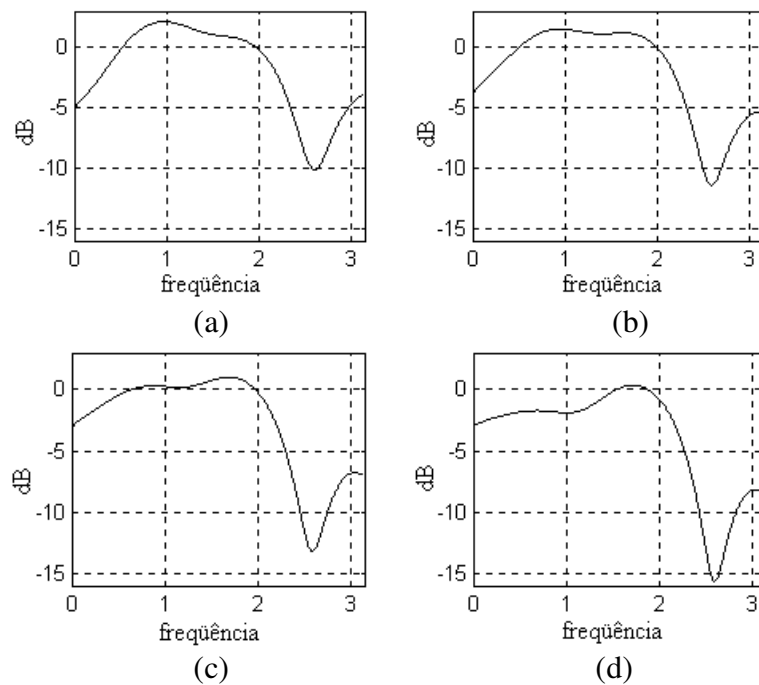


Figura 1.7: “Fotografia instantânea” das magnitudes dos espectros correspondentes às respostas ao impulso do canal mostrado na Figura 1.6.



## 1.5 Contribuições na forma de artigos publicados

Tivemos a oportunidade de publicar diversos artigos em congressos nacionais e internacionais a partir do segundo ano de doutorado. Nesses artigos estão disseminadas as nossas contribuições. Convém dizer que uma parte desses artigos não está ligada diretamente ao tema desta tese, por dois motivos principais: mudança do tema inicial de pesquisa e participação no projeto da TV Digital Brasileira (consórcio MI-SBTVD – *Modulação Inovadora para o Sistema Brasileiro de Televisão Digital*).

O foco inicial do nosso doutorado estava voltado para o tema “Rádio Definido por Software” (RDS). Mais especificamente, estudamos processamento multitaxa usando filtros CIC (*Cascaded Integrator Comb*) e propusemos uma modificação na sua estrutura básica para torná-lo aplicável em um sistema RDS. Seguindo esta linha de pesquisa, os artigos mais interessantes que publicamos foram:

- *Uma Estrutura Alternativa para Filtros CIC Aplicável em um Sistema de Rádio Definido por Software*. Apresentado no XXII Simpósio Brasileiro de Telecomunicações (SBrT'05), Campinas, São Paulo, setembro de 2005. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha, Leonardo S. Resende.
- *Rádio Definido por Software: Conceitos, Benefícios e Princípios de Projeto*. V Encontro de Pesquisa e Pós-Graduação (V ENPPG) do CEFET-Ce, Fortaleza, Ceará, novembro de 2005. Autores: Francisco J. A. de Aquino, Carlos E. B. Duarte e Carlos A. F. da Rocha.
- *Design of CIC Filters for Software Radio System*. IEEE International Conference on Acoustics, Speech, and Signal Processing – ICASSP 2006, Toulouse, França, maio de 2006. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha, Leonardo S. Resende.

Em paralelo à pesquisa de doutorado, também trabalhamos no Projeto da TV Digital Brasileira durante os anos de 2005 e 2006 tendo como foco a camada física e o uso de algumas das técnicas mais recentes de modulação. Os seguintes artigos são frutos da nossa participação nesse projeto:

- *A Noncoherent Space-Frequency Scheme for Frequency-Selective Fading Channels with Application to the Brazilian Digital TV Standard*. 4<sup>th</sup> International Information

and Telecommunication Technologies Symposium – I2TS, dezembro de 2005, Florianópolis, Santa Catarina. Autores: Francisco J. A. de Aquino, Gustavo Lima, Renato Machado, Bartolomeu Uchôa-Filho e Carlos A. F. da Rocha.

- *A Non-coherent Space-Frequency Coded OFDM Scheme for Frequency-Selective Fading Channels Applied to Mobile Digital TV*. Global Signal Processing Conference & Expos for the Industry, GSPx – 2006 – TV to Móbile Conference, março de 2006, Amsterdam, Holanda. Autores: Francisco J. A. de Aquino, Gustavo C. Lima, Renato Machado, B. F. Uchôa-Filho, Carlos A. F. da Rocha.
- *Design, Simulation and Hardware Implementation of a Digital Television System: STC-OFDM and Channel Estimation*. The 9<sup>th</sup> International Symposium on Spread Spectrum Techniques and Applications - ISSSTA - 2006, Manaus, Brasil, agosto de 2006. Autores: Gustavo C. Lima, Richard D. Souza, Luciano L. Mendes, Bartolomeu F. Uchôa Filho, Francisco J. A. de Aquino e Mário N. Neto (artigo convidado).
- *Projeto MI-SBTVD: STC-OFDM e Estimação de Canal*. Revista Inatel de Telecomunicações, edição especial, 3<sup>o</sup> trimestre de 2006. Autores: Gustavo C. Lima, Francisco J. A. de Aquino, Renato Machado, Bartolomeu F. Uchôa Filho, Mário Noronha Neto, Richard Demo Souza e Luciano L. Mendes.

Somente no segundo semestre de 2005 iniciamos o estudo do processamento LL. A idéia inicial era aliar o processamento LL ao RDS. Entretanto, o processamento LL se mostrou suficientemente rico e acabamos tirando totalmente o foco do tema RDS. Seguindo essa nova linha de pesquisa, trabalhamos principalmente com o problema de equalização largamente linear, visando tanto aspectos práticos, por exemplo, a implementação em FPGA, quanto estudos mais teóricos, como análise estatística do algoritmo LMS largamente linear. As nossas contribuições na forma de artigos estão relacionadas a seguir:

- *Accelerating the Convergence of the Widely Linear LMS Algorithm for Channel Equalization*, VI International Telecommunications Symposium (ITS2006), setembro 2006, vol. 1, pp. 57-61, Fortaleza-CE, Brazil. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
- *Processamento WL – Parte - I: Fundamentos do processamento WL aplicado à equalização de canais de comunicação*, VI Encontro de Pesquisa e Pós-Graduação,

- (VI ENPPG) do CEFET-Ce, Fortaleza, Ceará, agosto de 2006. Autores: Francisco J. A. de Aquino, Carlos Eugênio B. Duarte, Carlos Aurélio F. da Rocha.
- *Processamento WL - Parte- II: Acelerando a Convergência do Algoritmo LMS Largamente Linear Aplicado a Equalização de Canais Seletivos em Frequência*, VI Encontro de Pesquisa e Pós-Graduação, (VI ENPPG) do CEFET-Ce, Fortaleza, Ceará, agosto de 2006. Autores: Francisco J. A. de Aquino, Carlos Eugênio B. Duarte, Carlos Aurélio F. da Rocha.
  - *A FPGA Implementation of the Widely Linear LMS Adaptive Filter for Channel Equalization*, Proceedings of the 5<sup>th</sup> International Information and Telecommunication Technologies Symposium (I2TS2006), dezembro de 2006, Cuiabá-MT, Brazil. Autores: Francisco J. A. de Aquino, Cesar H. V. Vargas e Carlos Aurélio F. da Rocha.
  - *Widely Linear Prediction Applied to Blind Equalization*, International Workshop on Telecommunications (IWT2007), fevereiro de 2007, Santa Rita do Sapucaí-MG, Brazil. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
  - *Widely Linear Prediction for Blind Equalization*. In: IEEE International Conference on Communications - ICC - Glasgow, Scotland, junho de 2007. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
  - *Equalização de Canal Usando um Algoritmo LMS Largamente Linear Multi Split*. In: XXV Simpósio Brasileiro de Telecomunicações (SBrT'07), setembro de 2007, Recife. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
  - *Equalização Cega via um Banco de Filtros de Erro de Predição Largamente Lineares com Atraso de Equalização*. In: XXV Simpósio Brasileiro de Telecomunicações (SBrT'07), setembro de 2007, Recife. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
  - *Implementação e Comparação das Estruturas Paralela, Semi-Paralela e Sequencial de um Equalizador Largamente Linear em FPGA*. In: XXV Simpósio Brasileiro de Telecomunicações (SBrT'07), setembro de 2007, Recife. Autores: Cesar H. V. Vargas, Francisco J. A. de Aquino e Carlos Aurélio F. da Rocha.
  - *Multi-Split Widely Linear LMS Algorithm for Channel Equalization*. In: XXI Colloque GRETSI, setembro de 2007, Troyes – France. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.

- *Equalização Cega Usando Filtro de Erro de Predição Largamente Linear de Passo Variável*, Proceedings of the 6<sup>th</sup> International Information and Telecommunication Technologies Symposium (I2TS2007), dezembro de 2007, Brasília – Distrito Federal. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
- *Um Novo Equalizador Cego Misto Baseado em Filtro de Erro de Predição e Algoritmo CMA Largamente Lineares*, II Jornada Nacional de Produção Científica em Educação Profissional e Tecnológica, dezembro de 2007, São Luis – Maranhão. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.
- *Análise Estatística do Algoritmo LMS Largamente Linear e Cálculo do Atraso Ótimo para Equalização Largamente Linear*. XXV Simpósio Brasileiro de Telecomunicações (SBrT'08), setembro de 2008, Rio de Janeiro – RJ. Autores: Francisco J. A. de Aquino, Carlos A. F. da Rocha e Leonardo S. Resende.

Temos ainda um outro artigo publicado no SBrT 2008, mas sem relação direta com esta tese e um artigo para revista: *Equalização Largamente Linear Adaptativa de Canal de Comunicação Digital*, submetido à Revista IEEE América Latina em agosto de 2008.

## 1.6 Organização da tese

Organizamos a estrutura e as contribuições desta tese como segue.

No Capítulo 2, apresentamos uma revisão bibliográfica e a fundamentação teórica necessária ao bom entendimento do processamento LL e de sua aplicação a sistemas SISO e MIMO. Nesse capítulo, também incluímos alguns exemplos que ilustram os conceitos introduzidos. No final do capítulo, comentamos alguns trabalhos recentes que usam o processamento largamente linear.

No Capítulo 3, apresentamos a equalização de canais complexos usando filtros largamente lineares quando o sinal é impróprio. Mostramos as condições para a perfeita inversão de um canal FIR usando um equalizador largamente linear e o uso de algoritmos LMS de passo fixo ou variável aplicado ao processamento LL. Apresentamos uma análise estatística do algoritmo LMS largamente linear e a aplicação da transformada *multi-split* ao equalizador LL. Ainda no Capítulo 3, falamos brevemente sobre a tecnologia FPGA e sobre a implementação de equalizadores (lineares e largamente lineares) nessa tecnologia. Ao final do capítulo, incluímos algumas simulações ilustrando cada uma das seções apresentadas.

Convém observar que esse capítulo é uma mescla de algumas contribuições nossas e de resultados já estabelecidos na literatura.

No Capítulo 4 está uma parte importante das nossas contribuições: o uso e justificativa do filtro de erro de predição largamente linear para a equalização adaptativa de canais de fase não-mínima; o estabelecimento de um algoritmo LMS de passo variável para esse equalizador e a sua implementação em FPGA. Aqui, a equalização do canal é realizada de forma autodata ou cega<sup>5</sup>. Nesse capítulo, também apresentamos o algoritmo RLS aplicado ao filtro de erro de predição LL e um equalizador robusto baseado em um conjunto de filtros de erro de predição largamente lineares com atraso de equalização.

Finalmente, no Capítulo 5, apresentamos as conclusões e perspectivas para a continuação da pesquisa.

Incluimos ainda alguns apêndices nesta tese onde encontramos detalhes matemáticos, informações adicionais ou exemplos extras que complementam os capítulos apresentados.

---

<sup>5</sup> Usaremos os termos “cego” (*blind*) e “autodata” (*self-learning*) como sinônimos. Na literatura, o termo “cego” é de uso mais corrente que o termo “autodata”. Além desses dois termos, também podemos encontrar a expressão “não-supervisionado” na literatura.

# Capítulo 2

---

## Fundamentação Teórica

Neste capítulo revisaremos os fundamentos do processamento largamente linear em sistemas SISO e em sistemas MIMO. Antes, porém, apresentaremos o conceito de circularidade e algumas propriedades estatísticas de segunda ordem de sinais complexos, visando introduzir o processamento largamente linear.

Em seguida, relacionamos o conceito de circularidade com a estimativa largamente linear de um sinal real ou complexo, observando quando o processamento LL pode ser aplicado com vantagens em relação ao processamento estritamente linear.

Concluimos este capítulo apresentando o esquema de diversidade proposto por Alamouti (1998) e mostramos as vantagens da utilização do processamento LL nesse esquema.

### 2.1 Circularidade

A idéia básica que dá suporte ao processamento LL é o conceito de circularidade. O conceito de circularidade é detalhadamente estudado por Picinbono (1994). Apresentamos, aqui, apenas um resumo. Deixamos para o Apêndice A alguns conceitos básicos e detalhes matemáticos que foram omitidos neste capítulo.

#### 2.1.1 Conceito de circularidade

Seja  $\mathbf{z}$  um vetor cujos elementos são amostras de um processo aleatório complexo pertencente a  $\mathbb{C}^m$  e de média nula, isto é,  $E\{\mathbf{z}\} = \mathbf{0}$ . Este vetor pode ser escrito como  $\mathbf{z} = \mathbf{x} + j\mathbf{y}$ , onde  $\mathbf{x}$  é parte real,  $\mathbf{y}$  é a parte imaginária e  $j \triangleq \sqrt{-1}$ . O vetor  $\mathbf{z}$  é considerado normal (ou gaussiano) se  $\mathbf{x}$  e  $\mathbf{y}$  são um par de vetores conjuntamente normais. Neste caso, a distribuição de probabilidade deste par é definida inteiramente pelas matrizes de covariância de  $\mathbf{x}$  e  $\mathbf{y}$  e pela matriz de covariância cruzada entre  $\mathbf{x}$  e  $\mathbf{y}$ . Notamos que  $E\{\mathbf{z}\} = \mathbf{0}$  implica em  $E\{\mathbf{x}\} = \mathbf{0}$  e  $E\{\mathbf{y}\} = \mathbf{0}$ .

O vetor  $\mathbf{z}$  é chamado de circular normal se ele é normal e, além disso, se

$$\mathbf{C} \triangleq E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{0}, \quad (2.1)$$

onde  $\mathbf{C}$  é a matriz de *pseudo-covariância* (Nesser e Massey, 1993) ou *matriz de covariância complementar* (Schreier e Scharf, 2003), ou ainda, *função de relação* (Picinbono e Bondon, 1997). Quando (2.1) se verifica, o vetor  $\mathbf{z}$  é dito ser *próprio*. Do contrário, quando  $\mathbf{C} \neq \mathbf{0}$ ,  $\mathbf{z}$  é dito ser *impróprio*. Daí decorre um conceito mais amplo para dois vetores descorrelacionados: eles só são descorrelacionados quando a matriz de covariância e a matriz de pseudo-covariância são ambas nulas.

Podemos verificar que, para uma variável aleatória  $z = x + jy$ , com  $x$  e  $y$  independentes, de mesma variância e com média nula, o valor esperado  $E\{zz\}$  é nulo, como detalhado no Apêndice A, Seção A.3. Tendo isto em vista, é fácil concluir que a Equação (2.1) é satisfeita para vetores complexos se as partes reais e imaginárias são independentes, apresentam médias nulas e mesma variância, ver Therrien (1992).

A Figura 2.1 apresenta um exemplo numérico de dois sinais complexos, sendo o primeiro próprio e o segundo impróprio. O “sinal 1” foi gerado a partir da filtragem por um sistema linear invariante com parâmetros complexos de um sinal formado pela soma de dois sinais independentes com distribuição gaussiana e mesma variância, sendo o primeiro puramente real e o outro puramente imaginário. O “sinal 2” foi gerado a partir de um sinal com distribuição gaussiana, variância unitária, puramente real, filtrado pelo mesmo sistema utilizado para “o sinal 1”. Enquanto o primeiro sinal é próprio, o segundo não é, como podemos ver claramente na figura ao compararmos as suas funções de pseudo-covariância.

Uma consequência imediata do conceito de circularidade em (2.1) é a “invariância à rotação”, isto é, se  $\mathbf{z}$  é circular normal, então  $\mathbf{z}_\alpha = e^{j\alpha}\mathbf{z}$  e  $\mathbf{z}$  têm a mesma distribuição de probabilidade, sendo  $\alpha$  um ângulo qualquer em radianos.

- *Prova:* seja  $\mathbf{z}_\alpha$  o vetor  $e^{j\alpha}\mathbf{z}$ . Se  $\mathbf{z}$  é normal,  $\mathbf{z}_\alpha$  também é normal. Também é evidente que  $\mathbf{z}_\alpha$  e  $\mathbf{z}$  têm a mesma matriz de covariância, e se (2.1) se verifica, a matriz  $\mathbf{C}_\alpha$  correspondente a  $\mathbf{z}_\alpha$  também é nula, completando, assim, a prova.

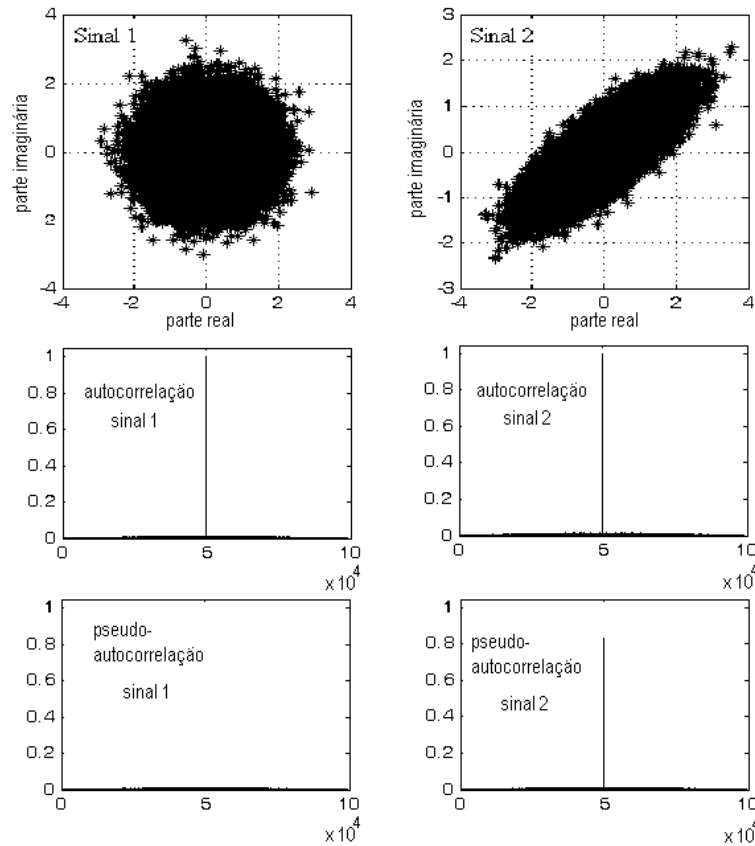


Figura 2.1. Exemplo de um sinal próprio e de um sinal impróprio.

### 2.1.2 Definições de circularidade

Relaxando a suposição de normalidade, podemos ter várias definições não equivalentes de circularidade. Inicialmente, considere o caso de uma variável aleatória (v.a.) escalar  $z$ . Esta v.a. é circular se, para qualquer  $\alpha$ ,  $z$  e  $e^{j\alpha}z$  possuem a mesma distribuição de probabilidade. Sejam  $A$  e  $\Phi$  duas variáveis aleatórias que denotam o módulo e a fase de  $z$ , respectivamente. A circularidade de  $z$  é caracterizada por:

$$p(a; \phi) = \frac{1}{2\pi} p(a), \text{ com } 0 \leq \phi < 2\pi, \quad (2.2)$$

onde  $p(a; \phi)$  e  $p(a)$  são as funções de densidade de probabilidade (PDF) de  $z$  e  $A$ , especificamente. Isto implica que  $A$  e  $\Phi$  são independentes e que  $\Phi$  é uniformemente distribuída no intervalo  $[0, 2\pi)$ . Naturalmente, a PDF de  $A$  é arbitrária. As definições sobre circularidade são apresentadas a seguir.



*Circularidade Marginal:* considere que os elementos  $z_k$  de um vetor aleatório complexo  $\mathbf{z}$  são v.a. complexas. Na distribuição marginal, consideramos separadamente cada componente independentemente uma da outra. Isto nos leva à seguinte definição: um vetor aleatório complexo  $\mathbf{z}$  é marginalmente circular se seus elementos  $z_k$  são v.a. complexas e circulares. A PDF marginal de cada componente  $z_k$  é então dada por (2.2), onde  $p(a)$  agora pode depender de  $k$ .

*Circularidade Fraca:* um vetor aleatório  $\mathbf{z}$  é fracamente circular se  $\mathbf{z}$  e  $e^{j\alpha}\mathbf{z}$  têm a mesma distribuição de probabilidade. Seja  $p(\mathbf{a}; \phi_1, \phi_2, \dots, \phi_m)$  a PDF conjunta de  $A_k$  e  $\Phi_k$  de  $Z_k$ ,  $1 \leq k \leq m$ . A circularidade fraca é caracterizada por

$$\begin{aligned} p(\mathbf{a}; \phi) &= p(\mathbf{a}; \phi_1, \phi_2, \dots, \phi_m) \\ &= p(\mathbf{a}; \phi_1 + \alpha, \phi_2 + \alpha, \dots, \phi_m + \alpha), \quad \forall \alpha. \end{aligned} \quad (2.3)$$

Como uma consequência,  $p(\mathbf{a}; \phi)$  pode ser descrita em função de  $m - 1$  variáveis  $\phi_{k'}s$ :

$$p(\mathbf{a}; \phi) = p(\mathbf{a}; \phi_2 - \phi_1, \dots, \phi_m - \phi_1). \quad (2.4)$$

Usando o fato de que as fases são definidas com módulo  $2\pi$ , a partir de (2.2) e (2.4) podemos verificar que circularidade fraca implica em circularidade marginal.

*Circularidade Forte:* um vetor aleatório  $\mathbf{z}$  é fortemente circular se

$$p(\mathbf{a}; \phi) = p(\mathbf{a}; \phi_1 + \alpha_1, \phi_2 + \alpha_2, \dots, \phi_m + \alpha_m) \quad (2.5)$$

para valores arbitrários de  $\alpha_k$ . Isto implica em

$$p(\mathbf{a}; \phi) = \left( \frac{1}{2\pi} \right)^m p_A(\mathbf{a}), \quad \text{com } 0 \leq \phi < 2\pi, \quad (2.6)$$

o que indica que as fases são variáveis aleatórias i.i.d. com distribuição uniforme e são independentes do vetor  $\mathbf{A}$  de módulos  $A_k$ ,  $1 \leq k \leq m$ . É claro que (2.5) implica em (2.3), o que indica que circularidade forte implica em circularidade fraca.

*Circularidade Total:* a circularidade total ocorre quando as variáveis aleatórias  $Z_{k's}$  são independentes e circulares. Isto é caracterizado pelo fato que  $p_A(\mathbf{a})$  em (2.6) pode ser escrito como o produto de fatores de funções  $p_k(a_k)$ . Notamos que circularidade total implica em circularidade forte, e que circularidade marginal e independência levam à circularidade total. A circularidade total ocorre principalmente no caso normal, com a matriz de autocorrelação  $\mathbf{R}_z = E\{\mathbf{z}\mathbf{z}^H\}$  diagonal. Neste caso, os módulos  $A_{k's}$  são distribuídos de acordo com uma PDF tipo Rayleigh. Na verdade, é fácil mostrar que se um vetor aleatório complexo normal é fortemente circular, sua matriz de autocorrelação é diagonal, sendo então totalmente circular (Picinbono, 1994).

## 2.2 Função de relação ou pseudo-covariância

Seja  $z[k]$  um sinal complexo discreto no tempo, onde  $k$  é um inteiro que descreve a dependência temporal do sinal. Assumindo que  $z[k]$  é aleatório e de média nula, sua estatística de segunda ordem é descrita pela função de autocovariância, definida por:

$$\gamma_z[k_1, k_2] \triangleq E\{z[k_1]z^*[k_2]\}, \quad (2.7)$$

onde  $*$  indica a conjugação complexa. Esta função, entretanto, não é suficiente para definir inteiramente a estatística de segunda ordem de  $z[k]$ , especialmente quando  $z[k]$  e seu conjugado  $z^*[k]$  são correlacionados. Neste caso, é interessante usarmos também a função de relação, ou pseudo-covariância, definida por:

$$r_z[k_1, k_2] \triangleq E\{z[k_1]z[k_2]\}. \quad (2.8)$$

Em geral, se  $z[k]$  corresponde à amostragem de um sinal aleatório passa-faixa de média nula, (2.8) é igual a zero (ver, por exemplo, Therrien (1992)) e pode, portanto, ser omitida. Este é o caso, por exemplo, do sinal analítico de qualquer sinal estacionário e, de forma mais geral, de qualquer sinal circular. Como uma consequência óbvia, observamos que os sinais reais não podem ser circulares. Contudo, não existe qualquer razão específica para que a pseudo-covariância de sinais complexos seja sempre nula, sendo, por isso, necessária para definir completamente as estatísticas de segunda ordem.

Da mesma forma que a função de covariância não pode ser arbitrária (é, por exemplo, uma função definida positiva), a função de relação também apresenta algumas características próprias. O conceito de função de relação também nos leva a rever o conceito de ruído branco complexo.

Tendo em mente todos esses conceitos, vamos investigar agora as relações entre circularidade e estacionaridade.

### 2.2.1 Conceito de estacionaridade

O sinal complexo  $z[k]$  é dito ser estacionário no sentido amplo (WSS – *wide sense stationary*) se seu valor médio é constante e se sua covariância (2.7) é uma função somente da diferença entre os instantes de tempo  $k_1$  e  $k_2$ . Esta definição não implica em qualquer condição para a função de relação em (2.8).

Podemos dizer que um sinal é estacionário de segunda ordem (SOS – *second-order stationary*) se ele é WSS e se sua função de relação depende também somente de  $k_1 - k_2$ . É claro que para sinais reais os dois conceitos coincidem. Por outro lado, sinais complexos estacionários no sentido amplo não são necessariamente SOS.

Assumindo que  $z[k]$  é estacionário, então a covariância e a função de relação dependem somente da diferença  $\kappa = k_1 - k_2$ :  $\gamma_z[\kappa] = \gamma_z[k_1, k_2]$  e  $r_z[\kappa] = r_z[k_1, k_2]$ . Sejam  $\Gamma_z(\nu)$  e  $R_z(\nu)$  as transformadas de Fourier de  $\gamma_z[\kappa]$  e  $r_z[\kappa]$ , respectivamente. Veremos, a seguir, as propriedades que as funções  $\Gamma_z(\nu)$  e  $R_z(\nu)$  precisam satisfazer.

Se  $z[k]$  é real, então essas duas funções são iguais e  $\Gamma_z(\nu)$  é o espectro de potência do sinal. Neste caso,  $\Gamma_z(\nu)$  precisa ser uma função simétrica, não negativa e a sua integral no intervalo  $[-1/2, 1/2]$  é um valor finito. Vale dizer que qualquer função satisfazendo essas condições pode ser considerada como sendo a transformada de Fourier da função de covariância de um sinal real.

No caso de  $z[k]$  ser complexo,  $\Gamma_z(\nu)$  mantém as características anteriores, exceto a propriedade de simetria. Como  $z[k]$  é estacionário,  $r_z[\kappa]$  e  $R_z(\nu)$  são simétricos, isto é,  $R_z(\nu) = R_z(-\nu)$ . Na Figura 2.2, mostramos exemplos de  $\Gamma_s(\nu)$  e  $R_s(\nu)$  para um sinal estacionário complexo  $s[k]$  e  $\Gamma_p(\nu)$  para sinal real  $p[k]$ , também estacionário.

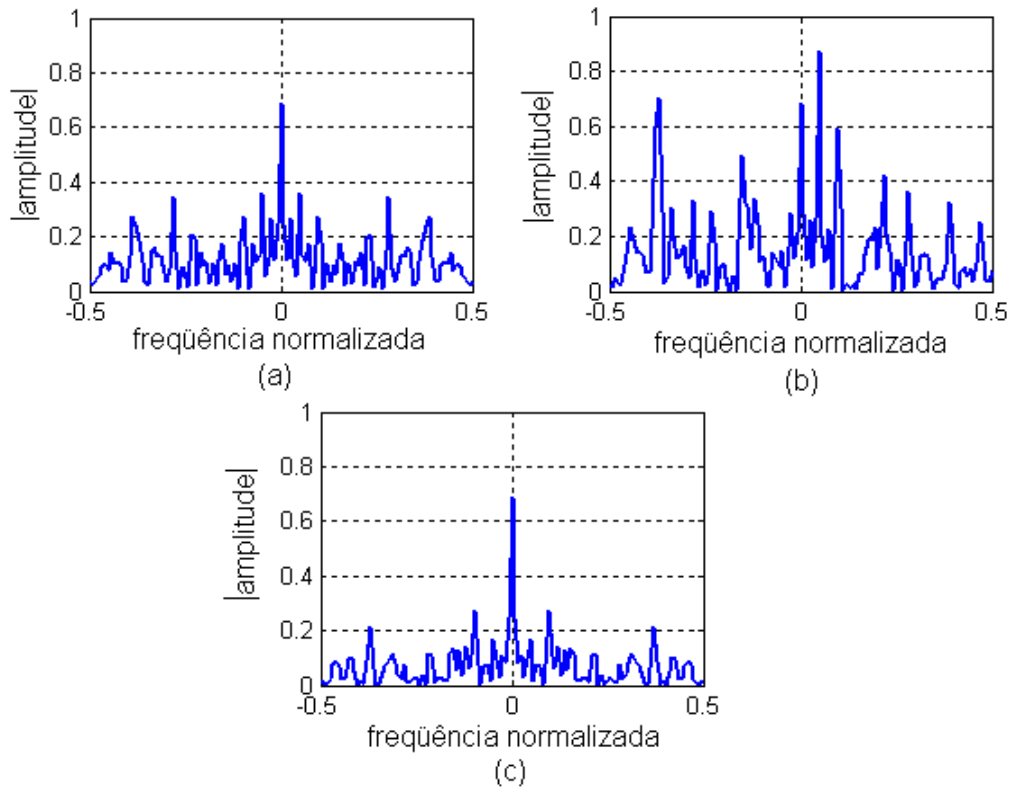


Figura 2.2. Transformada de Fourier (a) da função de relação do sinal complexo  $s[k]$ , (b) da covariância do sinal complexo  $s[k]$  e (c) da covariância do sinal real  $p[k]$ .

Considere o vetor complexo  $\mathbf{z}[k]$  definido por:

$$\mathbf{z}[k] = \begin{bmatrix} z[k], z^*[k] \end{bmatrix}_{(2 \times 1)}^T, \quad (2.9)$$

sendo  $\mathbf{z}[k]$  de média nula. A matriz de covariância deste vetor é dada por  $E\{\mathbf{z}[k]\mathbf{z}^H[k-\kappa]\}$  e possui transformada de Fourier  $\Gamma_z(\nu)$ , que pode ser expressa por

$$\Gamma_z(\nu) = \begin{bmatrix} \Gamma_z(\nu) & R_z(\nu) \\ R_z^*(-\nu) & \Gamma_z(-\nu) \end{bmatrix}, \quad (2.10)$$

onde  $\Gamma_z(\nu)$  é denominada de matriz espectral. Como  $R_z(\nu)$  é uma função simétrica e  $\Gamma_z(\nu)$  é uma matriz que precisa ser não negativa definida, então prontamente estabelecemos que

$$|R_z(\nu)|^2 \leq \Gamma_z(\nu)\Gamma_z(-\nu). \quad (2.11)$$

Esta é a condição necessária e também suficiente que  $R_z(\nu)$  precisa satisfazer para ser uma transformada de Fourier de um sinal com espectro de potência  $\Gamma_z(\nu)$  (Picinbono e Bondon, 1997). Podemos verificar que a relação em (2.11) ocorre na Figura 2.2.

### 2.2.2 Ruído branco e duplamente branco

Um sinal WSS  $z[k]$  é dito ser branco se seu espectro de potência é constante. Em termos de sua função de covariância, esta condição é equivalente a

$$\gamma_z[k, k-m] = \gamma_z[m] = \sigma_z^2 \delta[m], \quad (2.12)$$

onde  $\sigma_z^2$  é a variância do sinal e  $\delta[\cdot]$  é o impulso unitário (delta de Dirac). Entretanto, o fato do sinal ser branco não estabelece qualquer condição para a função de relação, que pode ser inclusive não-estacionária. Por este ponto de vista, podemos ter vários tipos de ruídos brancos. Se o conceito de SOS é adicionado, a função de relação precisa satisfazer a restrição indicada em (2.11), que toma a forma de

$$|R_z(\nu)|^2 \leq \gamma_z^2 = (\sigma_z^2)^2. \quad (2.13)$$

Em geral, é assumido que a propriedade de o sinal ser branco implica em  $R_z(\nu)$  igual a zero. Neste caso, podemos dizer que o ruído branco é circular. É fácil verificarmos que, para este caso, as partes real e imaginária de  $z[k]$  são descorrelacionadas e brancas. Quando isso não ocorre, a função de relação pode assumir uma forma semelhante a (2.12). Considerando a restrição imposta por (2.11) e levando em conta que a função de relação é simétrica, podemos escrever

$$r_z[m] = \mu_z (\delta[m-d] + \delta[m+d]), \quad (2.14)$$

onde  $\mu_z$  é um número complexo,  $d$  um valor inteiro positivo e  $|r_z[m]| \leq \sigma_z^2$ . A partir de (2.14), podemos expressar  $R_z(\nu)$  por

$$R_z(\nu) = 2\mu_z \cos(2\pi\nu d), \quad (2.15)$$

onde, de (2.13), concluímos que

$$|\mu_z|^2 \leq \frac{1}{4}(\sigma_z^2)^2. \quad (2.16)$$

Um sinal com essa característica é chamado de *duplamente branco* (Picinbono e Bondon, 1997). Não é difícil construir um sinal desse tipo. Considere, por exemplo,  $z[k]$  dado por:

$$z[k] = (1 + |a|^2)^{-1/2} (w[k] + aw^*[k-d]), \quad (2.17)$$

onde  $w[k]$  é um ruído branco circular ( $\sigma_z^2 = 1$ ),  $a$  é um número complexo (ou real) e  $d$  é um valor positivo inteiro. Vemos que a função de relação deste sinal corresponde a (2.14) e que  $z[k]$  permanece um ruído branco unitário. A partir de (2.14), temos que

$$\mu_z = a(1 + |a|^2)^{-1}. \quad (2.18)$$

Logo  $|\mu_z| \leq 1/2$ , cujo valor máximo ocorre quando  $|a| = 1$ . A Figura 2.3 mostra os gráficos da função de covariância e da função de relação de um sinal duplamente branco,  $z_{db}[k]$ , conforme (2.17), com  $a = 0,8$  (real),  $d = 2$  e  $w[k]$  um ruído branco de média nula e variância unitária. Notamos, ainda, que esse sinal é impróprio.

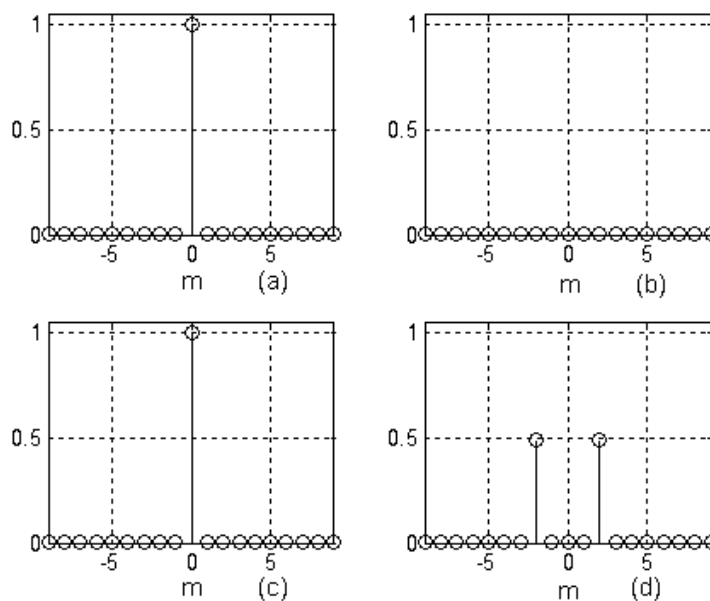


Figura 2.3: Amplitudes (a) covariância do sinal  $w[k]$ , (b) função de relação de  $w[k]$ , (c) covariância de  $z_{db}[k]$ , (d) função de relação de  $z_{db}[k]$ .

Aparentemente, o sinal construído com base em (2.17) seria artificial e não deveria ter aplicação prática. Contudo, devemos lembrar que, por exemplo, o código proposto por Alamouti (1998) como uma técnica de diversidade no transmissor emprega o sinal e o seu conjugado na transmissão da informação.

## 2.3 Processamento largamente linear

A idéia básica do processamento largamente linear (Picinbono e Chevalier, 1995) é manusear tanto o vetor de dados complexos  $\mathbf{x}[k]$  quanto o seu conjugado  $\mathbf{x}^*[k]$  para fazer uma estimativa da variável escalar  $y[k]$ :

$$\tilde{y}[k] = \sum_{n=0}^L c_n^*[k] x[k-n] + \sum_{n=0}^L d_n^*[k] x^*[k-n] = \mathbf{c}^H[k] \mathbf{x}[k] + \mathbf{d}^H[k] \mathbf{x}^*[k], \quad (2.19)$$

onde  $c_n[k]$  e  $d_n[k]$  são valores complexos.

A equação (2.19) indica uma operação de filtragem linear de  $x[k]$  e  $x^*[k]$ . É evidente que (2.19) não é uma operação de filtragem linear no sentido clássico. De fato, se associarmos as saídas  $\tilde{y}_1[k]$  e  $\tilde{y}_2[k]$  com as entradas  $x[k]$  e  $x^*[k]$ , respectivamente, a saída associada com  $x[k] + x^*[k]$  será  $\tilde{y}_1[k] + \tilde{y}_2[k]$ . Essa é uma condição necessária, mas não suficiente, para um sistema ser linear. Entretanto, a saída associada com  $\lambda x[k]$  não é, em geral,  $\lambda \tilde{y}_1[k]$  quando  $\lambda$  é um número complexo, que é uma propriedade necessária aos filtros lineares clássicos. Por outro lado, as características de segunda ordem da saída  $\tilde{y}[k]$  podem ser obtidas a partir da entrada  $x[k]$ . Portanto, o uso das expressões “*sistemas largamente lineares*” e “*processamento largamente linear*” são justificáveis.

A partir de (2.19), observamos que o problema da estimação largamente linear é encontrar os vetores  $\mathbf{c}$  e  $\mathbf{d}$  tais que  $E\{|y - \tilde{y}|^2\}$  seja minimizado. Com isto em mente, notamos primeiro que o conjunto escalar das variáveis aleatórias complexas  $z$  na forma  $z[k] = \mathbf{a}^H \mathbf{x}[k] + \mathbf{b}^H \mathbf{x}^*[k]$ , onde  $\mathbf{a}$  e  $\mathbf{b}$  pertencem a  $\mathbb{C}^N$ , constitui um espaço linear. Este espaço se torna um subespaço de Hilbert com o produto escalar definido por  $\langle z_1, z_2 \rangle = E\{z_1 z_2^*\}$ . Como conseqüência,  $\tilde{y}[k]$  é a projeção ortogonal de  $y[k]$  neste subespaço

e é caracterizado pelo princípio da ortogonalidade, isto é,  $(y[k] - \tilde{y}[k])$  é ortogonal a todos os componentes de  $\mathbf{x}[k]$  e de  $\mathbf{x}^*[k]$ . Este fato nos permite escrever

$$E\{\tilde{y}^*[k]\mathbf{x}[k]\} = E\{y^*[k]\mathbf{x}[k]\} \text{ e } E\{\tilde{y}^*[k]\mathbf{x}^*[k]\} = E\{y^*[k]\mathbf{x}^*[k]\}. \quad (2.20)$$

A partir de (2.19) e (2.20) e omitindo a dependência temporal “[k]” por questão de simplicidade, podemos obter o sistema de equações

$$\mathbf{R}\mathbf{c} + \mathbf{C}\mathbf{d} = \mathbf{r}, \quad (2.21)$$

$$\mathbf{C}^*\mathbf{c} + \mathbf{R}^*\mathbf{d} = \mathbf{s}^*, \quad (2.22)$$

onde

$$\mathbf{R} = E\{\mathbf{xx}^H\}; \quad \mathbf{C} = E\{\mathbf{xx}^T\}; \quad \mathbf{r} = E\{y^*\mathbf{x}\}; \quad \mathbf{s} = E\{y\mathbf{x}\}. \quad (2.23)$$

As soluções ótimas, segundo o critério MSE (LL) para  $\mathbf{c}$  e  $\mathbf{d}$  podem ser expressas por:

$$\mathbf{c} = [\mathbf{R} - \mathbf{C}\mathbf{R}^{-1*}\mathbf{C}^*]^{-1}[\mathbf{r} - \mathbf{C}\mathbf{R}^{-1*}\mathbf{s}^*] \quad (2.24)$$

$$\mathbf{d} = [\mathbf{R}^* - \mathbf{C}^*\mathbf{R}^{-1}\mathbf{C}]^{-1}[\mathbf{s}^* - \mathbf{C}^*\mathbf{R}^{-1}\mathbf{r}]. \quad (2.25)$$

O MSE correspondente vale:

$$\varepsilon^2 = P_y - \mathbf{c}^H\mathbf{r} - \mathbf{d}^H\mathbf{s}^*, \quad (2.26)$$

onde  $P_y = E\{|y|^2\}$ . Este erro é menor que o obtido pelo processamento estritamente linear, o qual é igual a  $\varepsilon_L^2 = P_y - \mathbf{r}^H\mathbf{R}^{-1}\mathbf{r}$ . A vantagem do procedimento LL sobre o estritamente linear é caracterizada pela quantidade  $\varepsilon_\delta^2 = \varepsilon_L^2 - \varepsilon^2$ , expressa também por

$$\varepsilon_\delta^2 = [\mathbf{s}^* - \mathbf{C}^*\mathbf{R}^{-1}\mathbf{r}]^H [\mathbf{R}^* - \mathbf{C}^*\mathbf{R}^{-1}\mathbf{C}]^{-1} [\mathbf{s}^* - \mathbf{C}^*\mathbf{R}^{-1}\mathbf{r}] \geq 0. \quad (2.27)$$



A igualdade em (2.27) ocorre somente quando  $\mathbf{s}^* - \mathbf{C}^* \mathbf{R}^{-1} \mathbf{r} = 0$ , pois a matriz  $\mathbf{R}^* - \mathbf{C}^* \mathbf{R}^{-1} \mathbf{C}$  é definida positiva. Note que, neste caso,  $\mathbf{d} = \mathbf{0}$ , o que corresponde a um processamento estritamente linear.

A seguir, analisaremos a relação entre circularidade e processamento largamente linear. Na subseção 2.3.4, apresentamos rapidamente um comentário entre o processamento LL e o estritamente linear aplicados ao problema de detecção e estimação.

### 2.3.1 Caso conjuntamente circular

Ocorre quando o sinal for próprio, isto é,  $\mathbf{C} = \mathbf{0}$ , e, além disso,  $\mathbf{s} = E\{y\mathbf{x}\} = 0$ . Nesse caso, os sinais  $y$  e  $\mathbf{x}$  são conjuntamente circulares. Como exemplo de um sinal circular temos o sinal analítico de um sinal real estacionário. Para esta situação, o sistema de equações (2.21) e (2.22) se reduz ao caso estritamente linear,  $\mathbf{R}\mathbf{c} = \mathbf{r}$ , e nenhum ganho pode ser obtido com o processamento LL sobre o processamento estritamente linear.

### 2.3.2 O vetor de observação é circular

Nessa situação, a matriz  $\mathbf{C} = \mathbf{0}$ , mas  $\mathbf{s}$  não é nulo. Esta situação pode ocorrer, por exemplo, em um sistema DS-CDMA (*Direct Sequence Code Division Multiple Access*) com uma constelação QPSK e sinal interferente não-circular (Gelli *et al.*, 2000). Para este caso, as equações (2.21) e (2.22) são bastante simplificadas e os coeficientes ótimos podem ser calculados por:

$$\mathbf{c} = \mathbf{R}^{-1} \mathbf{r} \text{ e } \mathbf{d}^* = \mathbf{R}^{-1} \mathbf{s}. \quad (2.28)$$

Nesse caso, ocorre um ganho do procedimento LL sobre o processamento estritamente linear, que pode ser expresso por  $\varepsilon_{\delta}^2 = \mathbf{s}^H \mathbf{R}^{-1} \mathbf{s} \geq 0$ .

### 2.3.3 O sinal transmitido é real

Consideremos, agora, o caso de um sinal real  $y$  (proveniente de uma constelação PAM, por exemplo) sendo transmitido através de um canal de comunicação modelado como um sistema linear com parâmetros complexos. Neste caso, o sinal recebido  $\mathbf{x}$  é também complexo

e impróprio. Para este caso, podemos chegar facilmente à conclusão de que  $\mathbf{r} = \mathbf{s}$ . Isto acaba por resultar em  $\mathbf{c} = \mathbf{d}^*$  e, em consequência,

$$\tilde{y} = 2\Re\{\mathbf{c}^H \mathbf{x}\}. \quad (2.29)$$

Em outras palavras, a estimativa  $\tilde{y}$  de um equalizador/estimador largamente linear de um sinal real é também um sinal real, o que não ocorre com um equalizador/estimador estritamente linear. O ganho nesse caso é dado por  $\varepsilon_\delta^2 = \mathbf{c}^H \mathbf{r}$ .

### 2.3.4 Detecção e estimação: comparação entre processamento LL e estritamente linear

Schreier *et al.* (2003) realizaram uma análise comparativa entre os processamentos estritamente linear e o largamente linear aplicados aos problemas de estimação e detecção. Nessa análise, eles mostram que o MSE de estimação do processamento LL é sempre menor e pode chegar a ser a metade do MSE do processamento estritamente linear quando a energia do ruído de estimação tende a zero.

Já para o problema de detecção, Schreier *et al.* mostram que o ganho máximo de desempenho também pode chegar a dois quando a energia do ruído tende a zero ou, ao contrário, quando a energia do ruído é muito maior que a energia do sinal.

No Apêndice A, mostramos um resumo do artigo desses pesquisadores com a sua formulação matemática. É interessante notar que, para o caso de equalização, seja ela treinada ou cega, o ganho do processamento LL sobre o linear pode ser muito maior que um fator de dois. Mostraremos isto nos capítulos 3 e 4 desta tese.

## 2.4 Exemplos de sinais complexos impróprios

Vimos que a convolução de uma seqüência de dados pertencente a uma constelação real por um canal complexo gera um sinal impróprio. Além dessa situação óbvia em que o processamento largamente linear é superior ao processamento linear, existem constelações complexas que também são impróprias e, conseqüentemente, podemos esperar um ganho adicional ao empregarmos o processamento largamente linear.

As técnicas de modulação complexas que apresentam um deslocamento (“*offset*”) das componentes em quadratura em relação às componentes em fase são conhecidas como “*staggered modulation*”. Exemplos dessa classe de sinais são: modulação OQAM (“*offset quadrature amplitude modulation*”), MSK (“*minimum-shift keying*”) e suas variantes como a modulação GMSK (“*Gaussian minimum-shift keying*”), a qual é empregada no sistema GSM (“*Global System for Mobile Communication*”). Todas estas constelações complexas geram sinais impróprios após serem filtradas por um canal complexo.

A constelação de um sinal OQAM é igual a uma constelação QAM convencional, mas sem as transições de 180° graus. As transições de 180° podem causar grandes variações na amplitude do sinal e a sua passagem por zero. Se forem usados amplificadores não-lineares, os quais são mais eficientes em relação ao consumo de energia, o sinal será distorcido e o espectro do sinal de saída será degradado (Goldsmith, 2005).

A Figura 2.4 apresenta o diagrama de blocos de um transmissor OQAM (as outras modulações “*staggered*” seguem esse mesmo esquema). Neste transmissor, os dados de entrada têm uma taxa de bit  $R_b$ . Após o conversor série/paralelo a taxa é reduzida para  $R_b/2$ . O atraso  $T_b = 1/R_b$  é introduzido somente na seqüência de bits  $a_Q(t)$ . Esta defasagem entre as seqüências  $a_I(t)$  e  $a_Q(t)$  garante que a transição de fase de um símbolo OQAM para outro seja de apenas 90°. A seguir, mostraremos que um sinal OQAM após passar por um canal complexo é impróprio.

Para a transmissão do sinal OQAM sobre um canal seletivo em frequência, podemos escrever o sinal recebido, sem ruído, como (Gerstacker *et al.*, 2003):

$$r(t) = \sum_{n=-\infty}^{+\infty} (-1)^n a[2n] g(t - 2nT_b) + j \sum_{n=-\infty}^{+\infty} (-1)^n a[2n+1] g(t - (2n+1)T_b), \quad (2.30)$$

onde  $g(t)$ ,  $2T_b$ ,  $a[2n]$  e  $a[2n+1]$  denotam a resposta ao impulso incluindo todos os efeitos do canal contínuo, a duração do bit e as componentes em fase e quadratura do sinal OQAM, respectivamente. Os símbolos  $a[\cdot]$  pertencem a uma constelação  $\sqrt{M}$ -ária ASK.

Após a amostragem à taxa de símbolo  $1/T_b$ , podemos escrever (2.30) como

$$r[k] = r(t) \Big|_{t=kT_b} = \sum_{m=-\infty}^{+\infty} \hat{a}_Q[m] g(kT_b - mT_b) + j \sum_{m=-\infty}^{+\infty} \hat{a}_I[m] g(kT_b - mT_b), \quad (2.31)$$

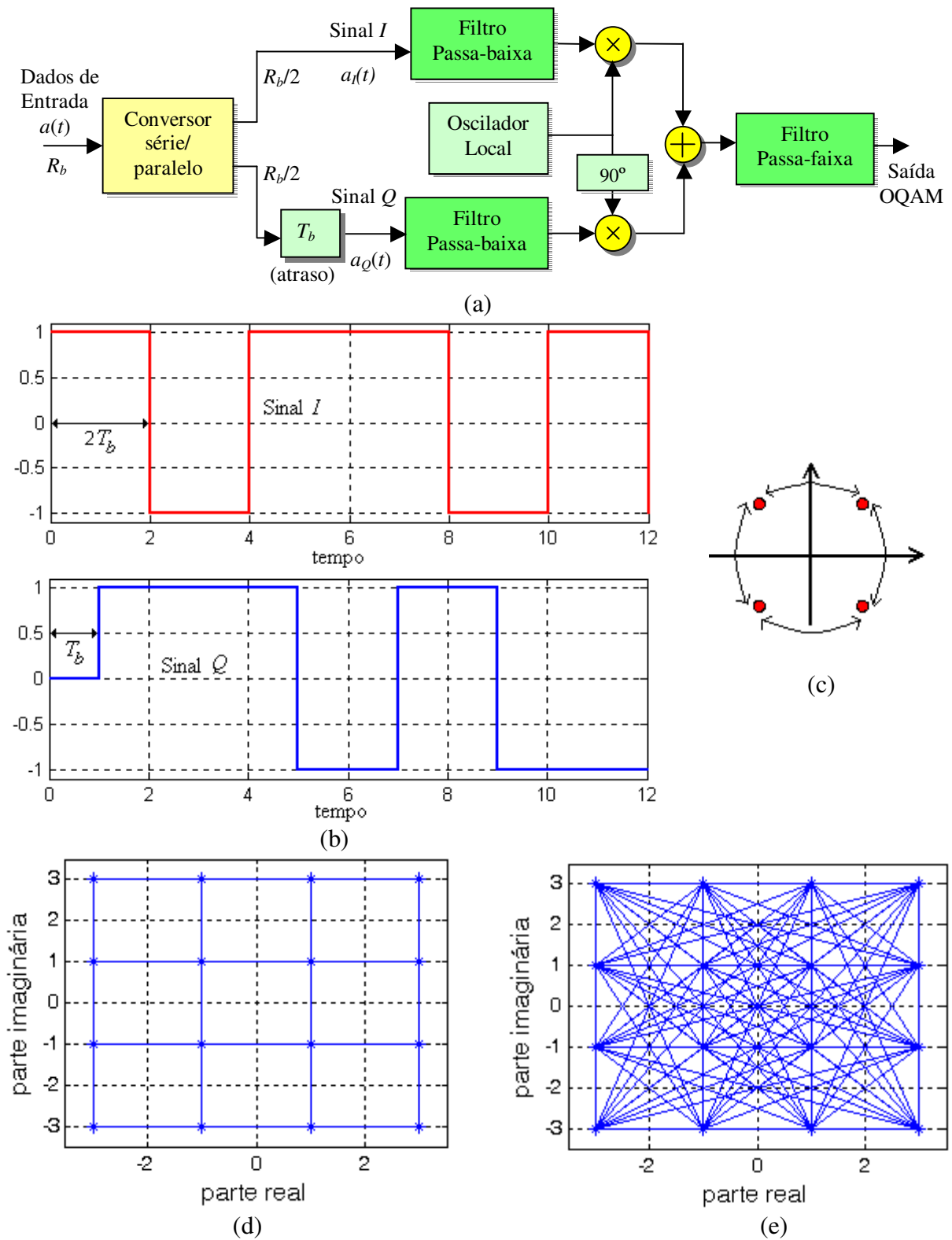


Figura 2.4: (a) Diagrama de blocos de um transmissor OQAM, (b) sinais  $I$  e  $Q$  defasados de  $T_b$ , (c) transição entre os símbolos de uma constelação OQAM de quatro pontos, (d) constelação 16-OQAM mostrando as transições, (e) constelação 16-QAM.

com

$$\hat{a}_o[m] = \begin{cases} (-1)^{m/2} a[m], & m \text{ par} \\ 0, & m \text{ ímpar} \end{cases} \quad (2.32)$$

$$\hat{a}_l[m] = \begin{cases} 0, & m \text{ par} \\ (-1)^{(m-1)/2} a[m], & m \text{ ímpar.} \end{cases} \quad (2.33)$$

Sendo que

$$\hat{a}_o[m] + j\hat{a}_l[m] = j^m a[m]. \quad (2.34)$$

Portanto, a seqüência recebida  $\bar{r}[k]$  depois da compensação de fase<sup>6</sup>, pode ser expressa por

$$\bar{r}[k] = \sum_{n=-\infty}^{+\infty} h[n] a[k-n], \text{ com } h[n] = j^{-n} g(nT). \quad (2.35)$$

Assim, podemos representar o sinal recebido pertencente a uma modulação OQAM como sendo o resultado da convolução de um sinal real por um canal complexo e, portanto, o processamento largamente linear pode ser aplicado vantajosamente. De forma semelhante, também pode ser mostrado que para a modulação MSK a Expressão (2.35) é uma boa aproximação do sinal recebido depois da amostragem e da compensação de fase.

Vale à pena ressaltar que Jerry C. Tu (1993) mostrou que os equalizadores convencionais projetados para esquemas de modulação QAM/QPSK são sub-ótimos para as modulações com “*offset*”. No mesmo artigo, o autor apresentou um esquema de equalização DFE (“*Decision Feedback Equalization*”) alternativo que aproveita o fato da modulação com “*offset*” ser “alternada” entre os eixos real e imaginário. No esquema proposto é usado um conversor série/paralelo e um atraso  $T_b$ , após o filtro “*feedforward*”, gerando uma seqüência de dados “reais” e outra seqüência de dados “imaginários”, ambas pertencentes a uma constelação  $\sqrt{M}$ -ária ASK.

---

<sup>6</sup> De acordo com (2.34),  $\hat{a}_r[m] + j\hat{a}_l[m]$  é gerada pela rotação de  $a[m]$  por um ângulo de  $m\pi/2$ . No receptor, a rotação pelo ângulo de  $-k\pi/2$  (compensação de fase), ou seja, a multiplicação da amostra por  $j^{-k}$  é realizada no instante  $k$ .

## 2.5 Processamento LL e códigos espaço-temporais de bloco (STBC)

A codificação espaço-temporal foi introduzida com o objetivo de combater o desvanecimento em situações em que podem ser empregadas múltiplas antenas transmissoras e uma ou mais antenas receptoras. O processamento LL apresenta um desempenho melhor que o processamento estritamente linear no caso de uma transmissão usando o código espaço-temporal de bloco proposto por Alamouti<sup>7</sup> (1998), independentemente da constelação usada ser real ou complexa (Gerstacker *et al.*, 2004).

### 2.5.1 Esquema de Alamouti

Faremos uma breve revisão do esquema de diversidade na transmissão proposto por Alamouti (1998), no qual são usadas duas antenas transmissoras e uma antena receptora. O canal é considerado plano e aproximadamente constante durante a transmissão de dois símbolos consecutivos. A Figura 2.5 mostra a representação em banda base desse esquema. Em um dado período  $T_s$  de símbolo, dois sinais são transmitidos simultaneamente a partir das duas antenas transmissoras. A antena transmissora 1 envia o sinal  $a_1[k]$  e a antena 2 envia o sinal  $a_2[k]$ . No próximo período de símbolo, a antena 1 envia o sinal  $-a_2^*[k]$  e a antena 2 envia o sinal  $a_1^*[k]$ . Esta operação está mostrada na Tabela 2.1. As seqüências  $a_1[k]$  e  $a_2[k]$  são consideradas identicamente distribuídas e pertencentes a uma constelação  $M$ -ária complexa ou real com variância  $\sigma_a^2$ .

O canal entre a antena transmissora 1 e a antena receptora é indicado por  $h_1 = \alpha_1 e^{j\theta_1}$  e entre a antena transmissora 2 e antena receptora é denotado por  $h_2 = \alpha_2 e^{j\theta_2}$ . Os sinais que chegam ao receptor podem ser expressos como

$$r[k] = r(kT_s) = h_1 a_1[k] + h_2 a_2[k] + n[k] \quad (2.36)$$

$$r[k+1] = r(kT_s + T_s) = -h_1 a_2^*[k] + h_2 a_1^*[k] + n[k+1], \quad (2.37)$$

<sup>7</sup> O esquema proposto por Alamouti, devido a sua simplicidade e elegância já faz parte dos padrões W-CDMA e CDMA-2000 (Gesbert, 2003).

onde  $n[k]$  e  $n[k+1]$  são sinais complexos que representam o ruído e a interferência captados pelo receptor. O combinador mostrado na Figura 2.5 realiza o seguinte cálculo usando os valores  $\hat{h}_1$  e  $\hat{h}_2$  estimados:

$$\tilde{a}_1[k] = \hat{h}_1^* r[k] + \hat{h}_2 r^*[k+1] = (\alpha_1^2 + \alpha_2^2) a_1[k] + \hat{h}_1^* n[k] + \hat{h}_2 n^*[k+1] \quad (2.38)$$

$$\tilde{a}_2[k] = \hat{h}_2^* r[k] - \hat{h}_1 r^*[k+1] = (\alpha_1^2 + \alpha_2^2) a_2[k] - \hat{h}_1 n^*[k+1] + \hat{h}_2^* n[k]. \quad (2.39)$$

Esses sinais obtidos são enviados ao detector de máxima verossimilhança para a decisão dos símbolos recebidos.

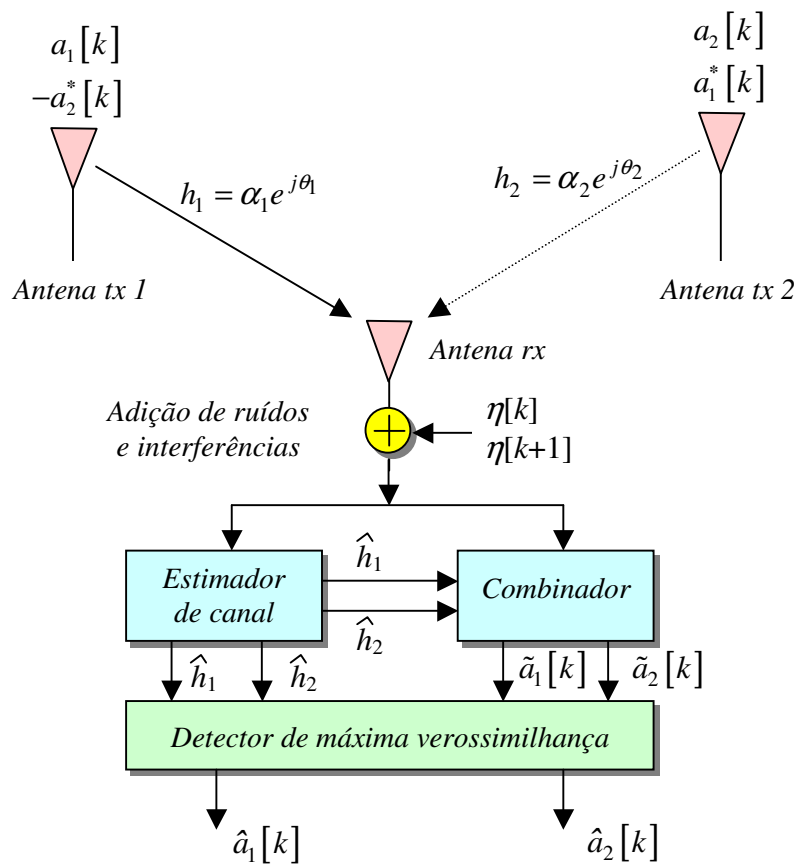


Figura 2.5. Esquema Alamouti: diversidade na transmissão.

TABELA 2.1: ESQUEMA DE CODIFICAÇÃO ALAMOUTI

	Antena 1	Antena 2
Instante $kT_s$	$a_1[k]$	$a_2[k]$
Instante $kT_s + T_s$	$-a_2^*[k]$	$a_1^*[k]$

## 2.5.2 Processamento LL e o esquema de Alamouti

Considerando o esquema de Alamouti, as seqüências de saída das antenas transmissoras 1 e 2 podem ser descritas por

$$\begin{aligned} s_1[k] &= \begin{cases} a_1[m], & k = 2m \\ -a_2^*[m], & k = 2m+1 \end{cases} \\ s_2[k] &= \begin{cases} a_2[m], & k = 2m \\ a_1^*[m], & k = 2m+1. \end{cases} \end{aligned} \quad (2.30)$$

Devido à repetição do sinal, os processos  $s_1[\cdot]$  e  $s_2[\cdot]$  são conjuntamente cicloestacionários com período 2. Para a nossa análise, entretanto, é mais conveniente organizar essas componentes polifásicas em um único vetor (Gerstacker *et al.*, 2004):

$$\mathbf{s}[m] \triangleq [s_1[2m] \ s_2[2m] \ s_1[2m+1] \ s_2[2m+1]]^T. \quad (2.31)$$

A função de relação desse vetor pode ser calculada por

$$\mathbf{C}[\kappa] = E\{\mathbf{s}[m]\mathbf{s}^T[m-\kappa]\}, \quad (2.32)$$

$$\mathbf{C}[\kappa] = \delta[\kappa] \left( \sigma_a^2 \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} \xi_a^2 & 0 & 0 & 0 \\ 0 & \xi_a^2 & 0 & 0 \\ 0 & 0 & (\xi_a^2)^* & 0 \\ 0 & 0 & 0 & (\xi_a^2)^* \end{bmatrix} \right), \quad (2.33)$$

onde  $\xi_a^2 \triangleq E\{(a_1[k])^2\} = E\{(a_2[k])^2\}$ . Por exemplo, para uma constelação  $M$ -PSK, com  $M > 2$ ,  $\xi_a^2 = 0$ . Dessa forma, como  $\mathbf{C}[\kappa] \neq \mathbf{0}_{4 \times 4}$ , o processo  $\mathbf{s}[\cdot]$  é impróprio (rotacionalmente variante) e o uso de processamento LL pode melhorar o desempenho do receptor.



## 2.6 Revisão bibliográfica

Além dos trabalhos sobre os fundamentos do processamento largamente linear realizados principalmente por Picinbono, nos últimos anos têm surgido diversas aplicações que utilizam o processamento LL. Iremos, agora, fazer uma rápida revisão de alguns desses trabalhos que empregam o processamento LL e de algumas teses que abordam principalmente a equalização de canais de comunicação.

### 2.6.1 Sistemas SISO

Talvez o trabalho mais interessante sobre equalização LL seja o de Gerstacker *et al.* (2003). Neste artigo os autores descrevem vários esquemas de equalização usando equalizadores lineares e com realimentação (DFE), além de revisarem os conceitos mais importantes para o processamento LL. Os autores também derivam um algoritmo LMS LL e o aplicam na equalização de canais fixos e canais aleatórios.

Dielt *et al.* (2005) mostram que o processamento LL combinado com detecção iterativa (receptor *turbo*) apresenta um desempenho superior aos esquemas lineares e uma complexidade computacional de mesma ordem.

Schober *et al.* (2004) introduzem e analisam três algoritmos estocásticos (LMS) cegos ou semi-cegos (“*data-aided*”) para filtros LL aplicados à supressão de interferências no sistema DS-CDMA. Os autores mostram que esses algoritmos apresentam uma complexidade computacional levemente menor que os seus correspondentes lineares, para o caso de todos os usuários usarem constelações reais.

Gelli *et al.* (2000) apresentam estruturas LL para detecção multiusuário de sinais em um sistema CDMA. Eles mostram, através de simulações, que essas estruturas apresentam um desempenho superior aos receptores lineares convencionais, em termos de capacidade de supressão de interferências, tanto em banda larga quando em banda estreita.

Jeon *et al.* (2006) propõem um algoritmo MOE (“*minimum output energy*”) LL para receptores CDMA. Esse algoritmo é aplicável tanto para constelações complexas quanto reais, sendo esta a sua maior vantagem. Quando o sinal empregado é real (modulação com valores reais) ou a interferência é imprópria, o algoritmo apresenta um desempenho muito superior aos algoritmos MOE lineares. Em contrapartida, perde em desempenho quando comparado ao algoritmo LL proposto por (Schober *et al.*, 2004), citado acima.

## 2.6.2 Sistemas MIMO

O número de trabalhos enfocando sistemas MIMO e processamento LL disponíveis na literatura ainda é relativamente pequeno. A seguir, destacaremos apenas dois desses trabalhos.

Um equalizador MMSE baseado em um DFE LL construído com estruturas FIR para a compensação de canais MIMO dispersivos no tempo é proposto por Mattera *et al.* (2005). Os autores mostram que esta combinação apresenta um melhor desempenho em relação aos equalizadores LL e DFE ao custo de um incremento moderado no esforço computacional. Também é mostrado que a estrutura DFE LL é mais sensível a erros na estimação da ordem do canal ou da sua resposta ao impulso.

O segundo trabalho (Gerstacker *et al.*, 2004) foca o uso do código de bloco espaço-temporal (STBC - *space-time block code*) proposto por Alamouti (1998) no desenvolvimento de receptores para canais com desvanecimento seletivos na frequência. Nesse trabalho, os autores demonstram que só é possível o emprego de equalizadores convencionais quando o número de antenas receptoras é pelo menos igual ao número de antenas transmissoras. Também é salientado que essa restrição pode ser contornada pelo emprego de estruturas LL, permitindo o uso de diversidade somente na transmissão ( $N_T = 2$  e  $N_R = 1$ ). Os resultados de simulação apresentados mostram que o processamento LL pode ser empregado vantajosamente em receptores GSM/EDGE (“*Enhanced data rates for GSM evolution*”).

## 2.6.3 Teses consultadas

Comentaremos brevemente as teses de doutorado de Sterle (2005) e Kuchi (2006) que abordam o processamento largamente linear.

O foco principal da tese de Sterle é a exploração do processamento largamente linear em sistemas de comunicação MIMO. Essa tese consta de apenas três capítulos. No primeiro, é apresentado o modelamento do canal MIMO e o ganho de um sistema de comunicação MIMO sobre um sistema SISO, tendo como ponto de vista a teoria da informação. Também nesse primeiro capítulo, são comentadas estratégias de equalização MIMO (não largamente lineares). No segundo capítulo, o foco é a equalização de canais MIMO usando uma estrutura DFE largamente linear. No terceiro capítulo, é estudado o projeto de transceptores largamente lineares considerando que o canal é conhecido tanto no receptor quanto no transmissor. Os principais resultados da tese de Sterle estão publicados em Mattera *et al.* (2003 e 2005).

Já a tese de Kuchi traz uma generalização do processamento (filtragem) largamente linear aplicado à equalização e supressão de interferências em sistemas PAM/QAM, para os casos de duas ou mais antenas receptoras e uma ou mais antenas transmissoras. A generalização proposta por Kuchi reside no fato de que o processamento LL passa a ser aplicável também quando são usadas constelações complexas do tipo QAM. Os principais resultados apresentados na tese de Kuchi estão condensados em Kuchi *et al.* (2005).

Além dessas duas teses diretamente relacionadas com a nossa pesquisa, consultamos várias outras teses defendidas recentemente que abordam o problema de equalização cega. Entre esses trabalhos podemos destacar: Claudio Bordin Jr. (2006), Romis Attux (2005) e Magno Silva (2005).

Na tese de Bordin Jr. são propostos e analisados métodos numéricos para a equalização autodidata de canais de comunicação FIR baseados em filtros de partículas, uma técnica recursiva para solução Bayesiana de problemas de filtragem estocástica. Esses filtros apresentam robustez e velocidade de convergência melhores que os métodos baseados exclusivamente em momentos. O autor reconhece, entretanto, que devido o elevado custo computacional das técnicas propostas, elas ainda não são adequadas tendo em vista uma aplicação prática.

Já a tese de Romis Attux versa sobre a equalização e identificação de sistemas usando estruturas e algoritmos não-lineares, entre eles: algoritmos genéticos, redes neurais e lógica *Fuzzy* (nebulosa). Vale a pena ressaltar que o emprego de tais técnicas acarretam um grande esforço computacional quando comparadas às técnicas clássicas que utilizam um algoritmo do tipo gradiente.

A tese de Magno Silva é um pouco mais “convencional”. Depois de uma excelente introdução ao problema de equalização cega de canais de comunicação, são propostos algoritmos baseados em EOS (CMA acelerado) para os casos mono e multiusuário. Ainda nessa tese, são propostos um equalizador concorrente DFE-CMA e um equalizador DFE híbrido que usa uma rede neural para tentar evitar a realimentação de decisões erradas.

## 2.7 Conclusão do capítulo

Neste capítulo apresentamos os conceitos de circularidade, ruído duplamente branco, função de relação, sinal próprio e sinal impróprio, que fundamentam o processamento

largamente linear. Introduzimos também o processamento LL aplicado à estimação largamente linear e discutimos alguns casos em que ele pode ser aplicado vantajosamente.

Na Seção 2.5 mostramos exemplos de sinais complexos que apresentam a característica de impropriedade. Em seguida, revisamos o esquema de Alamouti de diversidade na transmissão. Com esse esquema de codificação espaço-temporal de bloco, não importa se a constelação usada é real ou complexa, o sinal recebido será sempre impróprio e o processamento LL sempre poderá ser aplicado.

Finalmente, concluímos este capítulo com alguns comentários sobre diversos trabalhos (artigos e teses) encontrados na literatura que aplicam o processamento LL ou abordam o problema de equalização de canais de comunicação.

No próximo capítulo, aplicaremos os conceitos aqui estabelecidos, tendo como foco principal a equalização de canais complexos em sistemas SISO usando equalizadores largamente lineares.

# Capítulo 3

---

## Equalização Largamente Linear em Sistemas SISO

Neste capítulo aplicaremos o processamento LL e os conceitos apresentados no capítulo anterior para a equalização de canais complexos usando filtros transversais. Usaremos o algoritmo LMS na sua versão LL (LMS-LL) para a adaptação dos coeficientes dos equalizadores LL. Estes equalizadores poderão, portanto, apresentar um desempenho superior aos equalizadores lineares ao compensar canais complexos quando forem transmitidos sinais “reais”. Estaremos, assim, usando sinais *impróprios* e que não são *duplamente brancos*. Neste capítulo, nos limitaremos à equalização treinada. O problema de equalização autoidata será o foco principal do próximo capítulo desta tese.

Inicialmente, justificamos a necessidade de equalização de um canal de comunicação e discutimos brevemente sobre os equalizadores lineares. Em seguida, projetaremos equalizadores largamente lineares ótimos para canais complexos. Mostraremos que esses equalizadores ótimos precisam ter apenas a mesma ordem do canal para sua perfeita inversão, o que é uma grande vantagem em termos de complexidade computacional. Essa inversão é possível mesmo nos casos em que o canal apresente nulos espectrais. Tal resultado não pode ser considerado novo, mas não tem sido destacado na literatura. Em seguida, apresentamos o algoritmo LMS-LL e realizamos sua análise estatística. Na seção seguinte, comentamos rapidamente sobre o uso de passo variável no algoritmo LMS-LL. Aplicamos a transformada Multi-Split ao equalizador LL na Seção 3.8. Na última seção deste capítulo, implementamos o equalizador LL de passo fixo em FPGA.

No final do capítulo, encontramos uma ampla seção dedicada a ilustrar a teoria apresentada através de vários exemplos numéricos de simulação.

### 3.1 Equalização linear

Os canais de comunicação, em geral, são variantes no tempo e podem causar interferência entre símbolos (IES). A redução da IES é requerida quando a duração  $T_s$  do símbolo é da mesma ordem de grandeza do valor rms  $T_\sigma$  do espalhamento de atraso do canal. Por exemplo, telefones sem fio operam tipicamente em lugares fechados, onde o

espalhamento de atraso é pequeno. Considerando, ainda, que a transmissão de um sinal de voz requer uma taxa de dados relativamente baixa, a equalização nesse caso não se faz, em geral, necessária. Por outro lado, o padrão celular digital IS-54 é projetado para uso em ambientes abertos, onde  $T_\sigma \approx T_s$ . Por isso a equalização faz parte desse padrão. Aplicações que requerem altas taxas são mais sensíveis ao espalhamento de atraso e geralmente requerem equalizadores de alto-desempenho no receptor ou outras técnicas de redução de IES. De fato, o tratamento dos problemas decorrentes do espalhamento de atraso é um dos pontos mais desafiadores para a transmissão de dados em altas taxas em sistemas de comunicação (Goldsmith, 2005).

Em geral, na saída do equalizador existe um dispositivo de decisão (quantizador) que determina qual símbolo da constelação transmitida é o mais próximo ao sinal obtido na saída do equalizador. A Figura 3.1 ilustra os efeitos de um canal complexo em uma constelação 4-QAM e a mesma constelação após ser compensada por um equalizador. Para este exemplo, os símbolos na saída do dispositivo de decisão são corretos, pois o diagrama de olho após o equalizador está bem aberto.

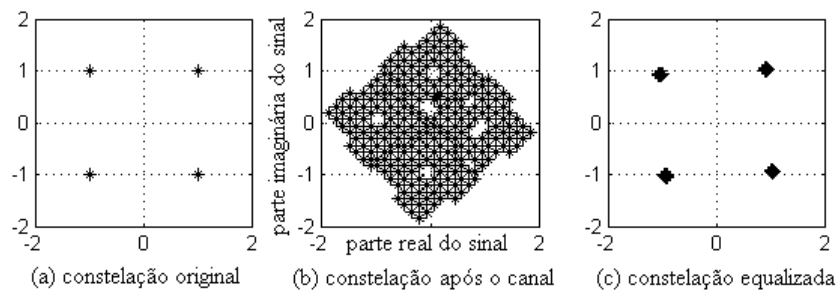


Figura 3.1. Exemplo dos efeitos de um canal de sobre uma constelação 4-QAM e sua equalização: (a) constelação transmitida, (b) constelação na saída do canal, (c) constelação equalizada.

No projeto de equalizadores, é necessário existir um equilíbrio entre a redução da IES e uma possível amplificação de ruído, já que tanto o ruído quanto o sinal são filtrados pelo equalizador. Além disso, como o canal é variante no tempo, o equalizador deve ser um filtro adaptativo. Para que ele tenha essa característica de adaptabilidade, deve haver um algoritmo de adaptação que ajuste os valores dos seus coeficientes. Contudo, quando o canal varia rapidamente, o ajuste dos coeficientes do equalizador pode não acompanhar a variação do canal e o desempenho do equalizador não será satisfatório.

Existem dois critérios para a redução da IES que são amplamente usados nos algoritmos adaptativos: os critérios ZF (*zero forcing*) e MSE (*mean square error*). O primeiro

a ser proposto na literatura foi o critério ZF, nos trabalhos pioneiros de R. Lucky de 1965 e 1966, publicados no *Bell System Technical Journal*.

Um equalizador ZF tenta minimizar a distorção<sup>8</sup>  $D$  provocada pelo canal utilizando o critério *minimax*. Como este critério não leva em conta o ruído presente na comunicação, a equalização ZF tende a amplificar mais o ruído que um equalizador usando o critério MSE. Além disso, não é garantida a convergência do algoritmo quando o diagrama de olho do sinal binário não equalizado está fechado, isto é, quando  $D > 1$ . Esses são os principais motivos para o uso mais difundido do critério MSE em relação ao critério ZF. Apesar disso, devido à simplicidade dos equalizadores ZF, esses foram os primeiros a serem incorporados aos modems comerciais (Benedetto *et al.*, 1987).

No seu artigo original, Lucky (1965) também apresenta uma implementação prática de um equalizador ZF com 13 coeficientes, sendo o coeficiente central fixo, e a proposta de uma estratégia mais elaborada para o algoritmo ZF quando  $D > 1$ . A expressão “equalização adaptativa” começou a ser usada a partir do artigo de Lucky de 1966.

Em geral, o algoritmo de adaptação utiliza um sinal de erro para realizar o ajuste nos coeficientes do equalizador. O sinal de erro pode ser calculado pela diferença entre um sinal de referência (sinal desejado) e a saída do equalizador. Dependendo de como esse sinal de referência tenha sido obtido, o equalizador pode estar operando em um dos seguintes modos: treinamento, decisão direta (*tracking* - rastreamento) e modo autodidata ou cego.

No modo de treinamento, uma seqüência de dados conhecidos é transmitida e a mesma seqüência é gerada no receptor. Pela comparação da saída do equalizador com a réplica sincronizada armazenada no receptor (sinal de referência), os parâmetros do filtro são ajustados. O modo de decisão direta ocorre ao final da fase de treinamento, quando o dispositivo de decisão já deve estar tomando decisões confiáveis, sendo a seqüência de treinamento substituída pela saída do dispositivo de decisão. Quando não existe uma seqüência de treinamento disponível para a inicialização do equalizador, mas somente o modelo estatístico do sinal transmitido, então o equalizador opera no modo autodidata.

A Figura 3.2 mostra dois diagramas de blocos representando um equalizador linear formado por um único filtro digital e o equalizador DFE (*decision feedback equalizer*). Nesse exemplo, os dois equalizadores usam apenas filtros do tipo FIR (*finite impulse response*). Esta

---

<sup>8</sup> A distorção  $D$  é definida como (Lucky, 1965):  $D = (1/h_0) \sum_{n=-\infty, n \neq 0}^{\infty} |h_n|$ , sendo  $h_n$  a resposta ao impulso do canal e  $h_0$  o impulso de maior energia.

restrição não é “obrigatória”, mas, na prática, a maioria dos equalizadores não utiliza estruturas IIR (*infinite impulse response*) (Monsen, 1971). Nos dois casos, os equalizadores tentam compensar os efeitos do canal e produzir uma estimativa correta do símbolo recebido. O erro de estimativa  $e[k]$  é utilizado para ajustar os valores dos coeficientes dos filtros no caso de um processamento adaptativo. A resposta desejada  $d[k]$  pode vir de uma seqüência de treinamento, quando isso for possível, ou dos símbolos decididos  $\hat{a}[k]$ .

Nesses equalizadores, a decisão pelo símbolo  $\hat{a}[k]$  depende somente do símbolo  $\tilde{a}[k]$  na entrada do quantizador, ou seja, ocorre uma decisão símbolo a símbolo. Essa é uma solução subótima que apresenta uma baixa complexidade computacional, podendo ser facilmente implementada por um circuito digital.

O desempenho do equalizador linear tende a ser pobre em canais que possuem zeros próximos ao círculo de raio unitário (CRU). O equalizador DFE é mais robusto nesse tipo de cenário e apresenta uma menor TEB. Entretanto, se um símbolo for decidido erroneamente, poderá ocorrer uma rajada de erros (“*erro burst*”) devido à realimentação.

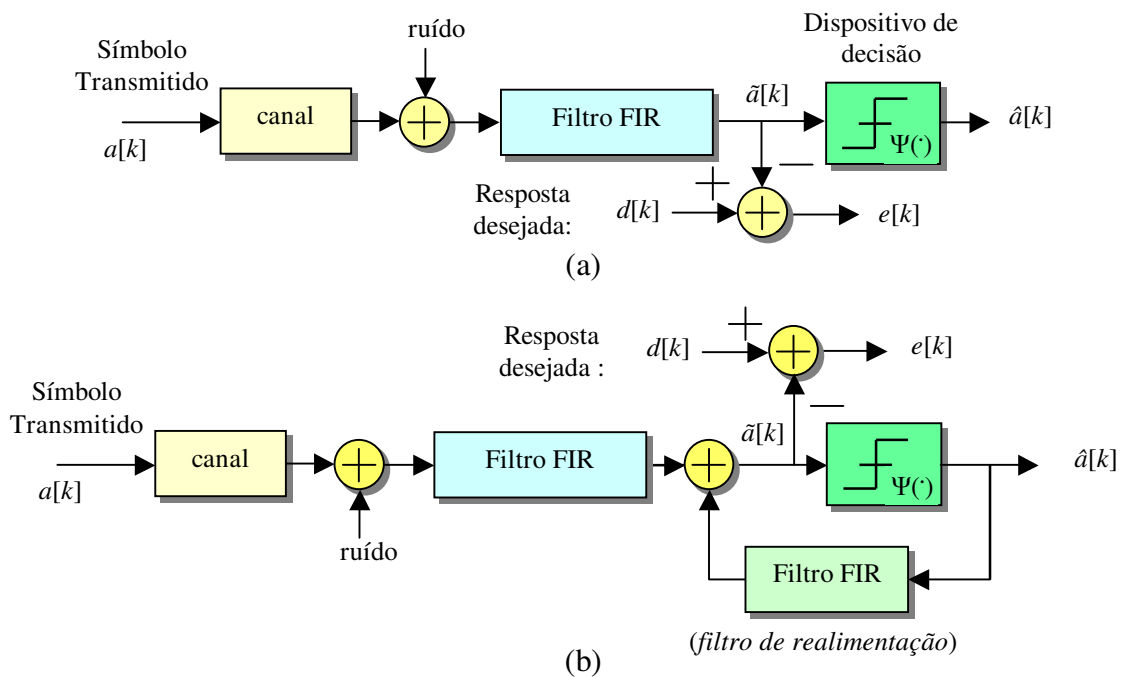


Figura 3.2: Diagrama de blocos (a) equalizador transversal (b) DFE.

Um outro ponto que deve ser levado em conta diz respeito à estabilidade do equalizador. Um filtro transversal (filtro FIR) é uma estrutura inerentemente estável, mesmo que tenha muitos coeficientes. Um filtro IIR pode ter um desempenho superior em relação a



um filtro FIR, no contexto de equalização de canal, com um número inferior de coeficientes, o que leva a uma economia de energia e *hardware*. Entretanto, especialmente no contexto adaptativo, sua estabilidade é um ponto crítico e não existe um compromisso possível entre baixa complexidade computacional e uma possível instabilidade, como destaca Barbosa *et al.* (2000). Esse é um dos principais motivos da predileção do uso de filtros lineares transversais para o problema de equalização. Isso, contudo, não impede que um equalizador cego utilize um filtro IIR, mas para isso o algoritmo adaptativo precisa ser especialmente robusto e, em consequência, complexo, como podemos ver na tese de doutorado de Attux (2005).

Um equalizador pode ser implementado no receptor em banda base, RF, ou IF. A maioria dos equalizadores é implementada digitalmente depois de conversão A/D em banda base (Goldsmith, 2005; Rappaport, 1996).

### 3.2 Equalizador largamente linear

A Figura 3.3 mostra o modelo de um equalizador largamente linear (equalizador LL). Os filtros  $f^*[k]$  e  $g^*[k]$  são do tipo FIR e de ordem  $P$ . Um ponto fundamental para este equalizador é a operação de conjugação efetuada antes do filtro  $g^*[k]$ . Assumiremos que a seqüência de símbolos transmitidos, representada por  $a[\cdot]$ , é constituída por símbolos de uma constelação real, independentes, identicamente distribuídos (i.i.d.) e com variância  $\sigma_a^2$ .

Considerando uma equalização adaptativa, o erro  $e[k]$  é usado para o ajuste dos coeficientes através de algum algoritmo adaptativo, como o algoritmo LMS. Quando o equalizador está na fase de treinamento, a *resposta desejada* é uma versão de  $a[k]$  atrasada de  $k_0$  amostras ( $k_0 \geq 0$ ). Após essa fase de treinamento, o equalizador entra na fase de decisão direta e a resposta desejada será o símbolo decidido  $\hat{a}[k]$ . A escolha do valor  $k_0$  deve ser criteriosa (e vai depender de cada canal), para que obtenhamos o melhor desempenho do equalizador (Voois *et al.*, 1996; Al-Dhahir e Cioffi, 1995; Endres *et al.*, 1996).

O ruído  $\eta[k]$  é considerado gaussiano complexo, branco, circular (próprio), com média nula, aditivo e com variância  $\sigma_\eta^2$ . O canal é admitido ser *verdadeiramente* complexo, isto é,  $\Re\{h_i[\cdot]\} \neq 0$  e  $\Im\{h_i[\cdot]\} \neq 0$ ,  $0 \leq i \leq L$ , de ordem  $L$  e representa os efeitos combinados da filtragem de transmissão, do canal contínuo no tempo, filtragem de recepção e da amostragem.

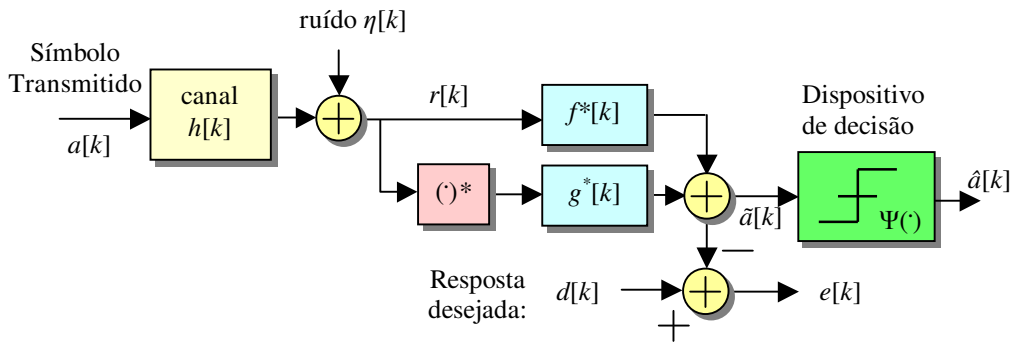


Figura 3.3: Modelo LL para filtragem inversa (equalização).

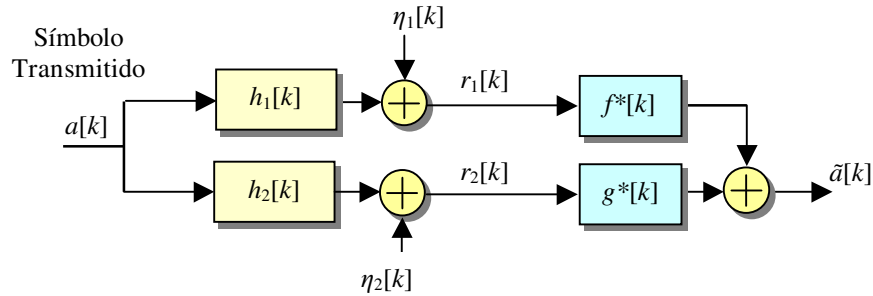


Figura 3.4: Modelo multicanal (polifásico) equivalente.

Ainda de acordo com a Figura 3.3, considerando uma transmissão com modulação linear sobre um canal com IES, podemos expressar o sinal complexo discreto equivalente recebido em banda base,  $r[k]$ , depois de uma amostragem à taxa de símbolo e sincronização perfeita da portadora, por

$$r[k] = \sum_{m=0}^L h_m[k] a[k-m] + \eta[k]. \quad (3.1)$$

O símbolo  $\tilde{a}[k]$ , calculado na saída do equalizador, pode ser expresso por

$$\begin{aligned} \tilde{a}[k] &= \sum_{n=0}^P f_n^*[k] r[k-n] + \sum_{n=0}^P g_n^*[k] r^*[k-n], \\ &= \mathbf{f}^H[k] \mathbf{r}[k] + \mathbf{g}^H[k] \mathbf{r}^*[k], \end{aligned} \quad (3.2)$$

onde definimos os vetores  $\mathbf{f}[k]$ ,  $\mathbf{g}[k]$  e  $\mathbf{r}[k]$  como:  $\mathbf{f}[k] \triangleq [f_0[k] \ f_1[k] \ \dots \ f_P[k]]^T$ ,  $\mathbf{g}[k] \triangleq [g_0[k] \ g_1[k] \ \dots \ g_P[k]]^T$  e  $\mathbf{r}[k] \triangleq [r[k] \ r[k-1] \ \dots \ r[k-P]]^T$ .

O equalizador linear da Figura 3.3 também pode ser representado utilizando uma estrutura multicanal (sistema polifásico), como indica a Figura 3.4 (Gerstacker *et al.*, 2003). Neste caso, o canal é modelado como dois subcanais complexos conjugados ( $h_1[k] = h_2^*[k] = h[k]$ ). Os ruídos  $\eta_1[k]$  e  $\eta_2[k]$  também são conjugados, mutuamente decorrelacionados (ver Seção 2.1) e apresentam a mesma variância,  $E\{|\eta_1[k]|^2\} = E\{|\eta_2[k]|^2\} = \sigma_\eta^2$ . Essa forma de visualizar o equalizador LL é interessante, pois nos permite usar parte da teoria já desenvolvida para equalizadores fracionalmente espaçados (FSE), como veremos nas próximas seções.

### 3.3 Equalizador LL ótimo

A seguir, determinaremos o equalizador LL ótimo segundo o critério MSE. De uma forma geral, os coeficientes ótimos devem ser calculados para cada instante  $k$ , mas, por uma questão de conveniência, iremos omitir essa dependência temporal dos filtros  $\mathbf{f}$  e  $\mathbf{g}$  e consideraremos o canal fixo. Observando (3.2) e usando a notação vetorial introduzida no Capítulo 1, podemos expressar o erro na resposta do equalizador LL por

$$\begin{aligned} e[k] &= d[k] - \tilde{a}[k] \\ &= a[k - k_0] - (\mathbf{f}^H \mathbf{r}[k] + \mathbf{g}^H \mathbf{r}^*[k]), \end{aligned} \quad (3.3)$$

onde  $\mathbf{r}[k]$  pode ser descrito como

$$\mathbf{r}[k] = \mathbf{H}\mathbf{a}[k] + \boldsymbol{\eta}[k], \quad (3.4)$$

onde  $\mathbf{H}$  é a matriz de convolução do canal com uma dimensão  $(P+1) \times (P+L+1)$ . A matriz  $\mathbf{H}$  pode ser descrita por:

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \dots & h_L & 0 & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_L & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_0 & h_1 & \dots & h_L \end{bmatrix}, \quad (3.5)$$

$\mathbf{a}[k] = [a[k] \ a[k-1] \ \dots \ a[k-P-L]]^T$  é um vetor de dimensão  $(P+L+1) \times 1$  contendo os símbolos transmitidos, e o vetor de ruído  $\boldsymbol{\eta}[k] = [\eta[k] \ \eta[k-1] \ \dots \ \eta[k-P-L]]^T$  tem dimensão igual a  $(P+1) \times 1$ .

Através da minimização de  $E\{|e[k]|^2\}$  podemos obter os valores ótimos dos filtros  $\mathbf{f}$  e  $\mathbf{g}$  pelo critério MMSE. Este cálculo é realizado usando as regras padrões de derivação complexa aplicadas a vetores, ver, por exemplo, (Haykin, 2002). Estes cálculos estão apresentados no Apêndice B. O sistema de equações resultante é descrito por:

$$\begin{bmatrix} \mathbf{H}\mathbf{H}^H + \xi\mathbf{I}_P & \mathbf{H}\mathbf{H}^T \\ \mathbf{H}^*\mathbf{H}^H & \mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I}_P \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} = \begin{bmatrix} \mathbf{h}_{k_0} \\ \mathbf{h}_{k_0}^* \end{bmatrix}, \quad (3.6)$$

onde  $\mathbf{I}_P$  é uma matriz identidade de dimensão  $(P+1) \times (P+1)$ ,  $\xi = \sigma_n^2 / \sigma_a^2$  é a razão entre a variância do ruído e do sinal, e  $\mathbf{h}_{k_0} = [h_{k_0} \ h_{k_0-1} \ \dots \ h_{k_0-P}]^T$ , sendo o atraso  $k_0 \geq 0$  e  $h_{k_0-P} = 0$  se  $k_0 - P < 0$ . A solução do sistema de equações em (3.6) resulta nos filtros ótimos:

$$\mathbf{f} = \left( \mathbf{H}\mathbf{H}^H + \xi\mathbf{I}_P - \mathbf{H}\mathbf{H}^T \left( \mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I}_P \right)^{-1} \mathbf{H}^*\mathbf{H}^H \right)^{-1} \left( \mathbf{h}_{k_0} - \mathbf{H}\mathbf{H}^T \left( \mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I}_P \right)^{-1} \mathbf{h}_{k_0}^* \right) \quad (3.7)$$

e

$$\mathbf{g} = \mathbf{f}^*. \quad (3.8)$$

Vale observar que é suficiente calcularmos o filtro  $\mathbf{f}$ , pois o filtro  $\mathbf{g}$  é simplesmente o seu conjugado. Na última seção deste capítulo, apresentaremos diversos exemplos numéricos para ilustrar o procedimento de equalização usando o equalizador LL.

### 3.4 Ordem mínima de um equalizador LL e atraso ótimo

Mostraremos nesta seção que o número mínimo de coeficientes nos filtros  $\mathbf{f}$  e  $\mathbf{g}$  de um equalizador LL necessários para compensar *perfeitamente* um canal FIR complexo, na ausência de ruído ( $\xi \rightarrow 0$ ), é igual a  $L+1$ , onde  $L$  é ordem do canal, quando a constelação utilizada é real. Este fato pode até nos surpreender à primeira vista, já que sabemos que um canal FIR só pode ser invertido perfeitamente por filtro também FIR de ordem infinita (no caso de sistemas SISO). Além disto, essa observação sobre a ordem mínima de um equalizador LL não é destacada na literatura, por isso achamos interessante reforçar este

ponto. Também calcularemos o atraso  $(k_0, \Delta)$  ótimo para uma equalização LL-ZF.

Aparentemente, Slock (1993) foi o primeiro pesquisador a notar que canais FIR superamostrados poderiam ser perfeitamente invertidos (equalização ZF) por um equalizador FSE de comprimento finito, apesar de os equalizadores FSE terem sido introduzidos já na segunda metade da década de 1970 (Guidoux e Macchi, 1975; Ungerboeck, 1976). Uma discussão detalhada sobre a ordem mínima de um equalizador FSE para canais FIR é apresentada por Harikumar e Bresler (1998). Estes autores provam formalmente que a ordem do equalizador FSE precisa ser somente maior ou igual a

$$N \geq \lceil (L-1)/(m-1) \rceil, \quad (3.9)$$

onde  $m > 1$  é um múltiplo da taxa de amostragem,  $L$  é o número de coeficientes de cada subcanal e  $N$  é o número de coeficientes do equalizador FSE, para que a convolução do canal com o equalizador resulte apenas em um impulso unitário (possivelmente atrasado).

Como já dissemos, o equalizador LL também pode ser visto como um equalizador polifásico de duas fases ( $m = 2$ ) e, sob algumas condições, a convolução do canal com o equalizador LL pode resultar em um único pulso atrasado (função delta de Dirac). Estas condições necessárias serão discutidas a seguir. Antes, porém, consideraremos o caso de equalização FIR convencional de um sistema com um único canal FIR.

Para o problema de equalização ZF de um canal FIR por um filtro também FIR podemos escrever, no domínio  $z$ :

$$F(z)H_1(z) = z^{-\Delta} \Rightarrow F(z) = \frac{1}{H_1^{\min}(z)} \frac{z^{-\Delta}}{H_1^{\max}(z)}, \quad (3.10)$$

onde  $H_1(z)$  foi fatorado em seus dois polinômios de fase mínima ( $H_1^{\min}(z)$ ) e de fase máxima ( $H_1^{\max}(z)$ ), respectivamente, assumindo que  $H_1(z)$  não tem zeros sobre o CRU. Como  $1/H_1^{\min}(z)$  é IIR e causal enquanto  $1/H_1^{\max}(z)$  é IIR e anticausal,  $F(z)$  é não causal e tem um comprimento *duplamente* infinito. Para um dado erro de aproximação,  $F(z)$  pode ser truncado para se tornar de comprimento finito e feito causal por uma escolha conveniente do atraso  $\Delta$ . O comprimento necessário de  $F(z)$  para obtermos o erro desejado vai depender da proximidade dos zeros de  $H_1(z)$  do CRU.

Para o equalizador LL podemos escrever (3.10), conforme a Figura 3.4, como:

$$F(z)H_1(z) + G(z)H_2(z) = z^{-\Delta}, \quad (3.11)$$

que corresponde à identidade ou teorema de Bezout<sup>9</sup> quando  $\Delta = 0$  (Papadias e Slock, 1999). Esta identidade estabelece que existem equalizadores FIR de comprimento finito para os “subcanais”  $H_1(z)$  e  $H_2(z)$ , desde que estes não possuam zeros em comum. Isto significa, no caso da equalização LL, que o canal  $H_1(z)$  não pode ter zeros reais e nem pares de zeros complexos conjugados para que seja possível a inversão perfeita do canal. A existência de zeros em comum em canais de comunicação complexos é um evento com probabilidade nula, segundo Harikumar e Bresler (1998). Entretanto, para canais com uma resposta ao impulso longa, certamente encontraremos muitos zeros próximos uns dos outros ou zeros “quase” reais. Estes zeros, “quase” comuns, contudo, nem sempre irão degradar significativamente o desempenho do equalizador correspondente (Papadias e Slock, 1999).

No domínio temporal, podemos escrever (3.11) como:

$$\mathbf{f} \star \mathbf{h}_1 + \mathbf{g} \star \mathbf{h}_2 = \mathbf{f} \star \mathbf{h} + \mathbf{g} \star \mathbf{h}^* = \delta[k - \Delta], \quad (3.12)$$

onde  $\star$  indica convolução. Expressando (3.12) como o produto da matriz de canal com o vetor de coeficientes do equalizador, obtemos:

$$\begin{bmatrix} h_0 & h_0^* & 0 & 0 & \dots & 0 & 0 \\ h_1 & h_1^* & h_0 & h_0^* & \ddots & 0 & 0 \\ \vdots & \vdots & h_1 & h_1^* & \ddots & \vdots & \vdots \\ h_L & h_L^* & \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & h_L & h_L^* & \ddots & h_0 & h_0^* \\ 0 & 0 & 0 & 0 & \ddots & h_1 & h_1^* \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & h_L & h_L^* \end{bmatrix} \begin{bmatrix} f_0^* \\ g_0^* \\ f_1^* \\ g_1^* \\ \vdots \\ f_P^* \\ g_P^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \Leftrightarrow \begin{bmatrix} h_0 & 0 & \dots & 0 & h_0^* & 0 & \dots & 0 \\ h_1 & h_0 & \ddots & 0 & h_1^* & h_0^* & \ddots & 0 \\ \vdots & h_1 & \ddots & \vdots & \vdots & h_1^* & \ddots & \vdots \\ h_L & \vdots & \ddots & 0 & h_L^* & \vdots & \ddots & 0 \\ 0 & h_L & \ddots & h_0 & 0 & h_L^* & \ddots & h_0^* \\ 0 & 0 & \ddots & h_1 & 0 & 0 & \ddots & h_1^* \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & h_L & 0 & \dots & 0 & h_L^* \end{bmatrix} \begin{bmatrix} f_0^* \\ f_1^* \\ \vdots \\ f_P^* \\ g_0^* \\ g_1^* \\ \vdots \\ g_P^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.13)$$

<sup>9</sup> Teorema de Bezout: sejam os polinômios  $H_i(z)$  com coeficientes complexos, com  $1 \leq i \leq p$ . Uma condição necessária e suficiente para a existência de polinômios  $F_i(z)$ , tais que  $\sum_{i=1}^p H_i(z)F_i(z) = 1$ , é que  $H_i(z)$  e  $F_i(z)$  não tenham zeros em comum em  $\mathbb{C}$  (Harikumar e Bresler, 1998).

ou, de forma mais compacta,

$$\mathbf{H}_z \mathbf{w}_z = \mathbf{h}_\Delta, \quad (3.14)$$

onde  $\mathbf{H}_z$  representa a matriz de canal com dimensão  $(L+P+1) \times (2P+2)$ , o vetor  $\mathbf{w}_z$  é formado pelo coeficientes  $f_i^*$  e  $g_i^*$ ,  $0 \leq i \leq P$ , do equalizador LL e possui dimensão  $(2P+2) \times 1$ , e  $\mathbf{h}_\Delta = [0, \dots, 0, 1, 0, \dots, 0]$  indica a resposta ao impulso combinada do canal e equalizador LL, com um certo atraso  $\Delta$ . O vetor  $\mathbf{h}_\Delta$  tem dimensão  $(2P+2) \times 1$ .

Temos, portanto, um sistema de equações lineares com  $2(P+1)$  variáveis e  $(L+P+1)$  equações. Para a existência de uma ou mais soluções em (3.13), a matriz de convolução do canal precisa ser de posto completo e, por conseqüência, obedecer à identidade de Bezout. Satisfeita a identidade de Bezout, precisamos ter  $2(P+1) \geq (L+P+1)$ , ou seja, o número mínimo de coeficientes do equalizador LL é  $P = L-1$  para uma perfeita compensação do canal (equalização ZF). Note que esta situação corresponde ao caso de uma equalização FSE com  $m = 2$  e é igual à solução dada por (3.9). Com esse número de coeficientes nos filtros  $\mathbf{f}$  e  $\mathbf{g}$ , o sistema de equações em (3.14) admite apenas uma solução. O número de coeficientes pode, entretanto, ser maior e o sistema (3.14) passa a permitir “infinitas” soluções.

Esse maior grau de liberdade pode ser usado para encontrar uma solução que amplifique menos o ruído. Nessa situação ( $P > L - 1$ ), a matriz do canal deixa de ser uma matriz quadrada e o sistema de equações em (3.14) admite infinitas soluções para os filtros  $\mathbf{f}$  e  $\mathbf{g}$  ( $\mathbf{f} = \mathbf{g}^*$ ) que podem compensar perfeitamente o canal, da mesma forma que uma equalização FSE-ZF (Papadias e Slock, 1999). Por exemplo, para um canal definido pela resposta ao impulso  $\mathbf{h}_c = [-0,3145+0,210j; 0,8386-0,3145j; 0,210+0,1048j]$ , os seguintes filtros conseguem equalizar o canal pelo critério ZF (atraso  $\Delta = 2$ ):

- $\mathbf{f}_1 = \mathbf{g}_1^* = [0,325-0,487j \quad 0,422-0,065j \quad -0,065-0,130j];$
- $\mathbf{f}_2 = \mathbf{g}_2^* = [-0,156+0,234j \quad 1,143-1,987j \quad -0,305-0,610j];$
- $\mathbf{f}_3 = \mathbf{g}_3^* = [0,223-0,335j \quad 0,574-0,469j \quad -0,116-0,231j].$

Dos equalizadores acima, somente o primeiro não amplifica significativamente um possível ruído que seja adicionado ao sinal. Entre essas infinitas soluções, a melhor é aquela que menos amplifica o ruído, logo tal condição restritiva pode ser usada para selecionar o melhor equalizador ZF. Se considerarmos um ruído branco na entrada do equalizador, a variância do ruído na saída do equalizador ZF pode ser expressa por (Papadias e Slock, 1999):

$$\begin{aligned}\sigma_{ZF}^2 &= \sigma_n^2 \sum_{i=0}^P |f_i^*|^2 + \sigma_n^2 \sum_{i=0}^P |g_i^*|^2 \\ &= 2\sigma_n^2 \sum_{i=0}^P |f_i^*|^2,\end{aligned}\tag{3.15}$$

desde que  $\mathbf{f} = \mathbf{g}^*$ .

Um procedimento para encontramos a solução ótima é fazer todos os coeficientes dos filtros  $\mathbf{f}$  e  $\mathbf{g}$  em (3.14) como função, por exemplo, de  $f_0^*$ . Ao minimizarmos (3.15) em relação a  $f_0^*$ , determinamos o valor ótimo de  $f_0^*$  e dos demais coeficientes. Todo este procedimento é equivalente a calcularmos a solução de norma mínima (Giannakis e Halford, 1997) quando o ruído é branco. Ou seja, o vetor de coeficientes  $\mathbf{w}_{z-\Delta}$  pode ser calculado por:

$$\mathbf{w}_{z-\Delta} = (\mathbf{H}_z^H \mathbf{H}_z)^{-1} \mathbf{H}_z^H \mathbf{h}_\Delta = \mathbf{H}^+ \mathbf{h}_\Delta,\tag{3.16}$$

onde  $\mathbf{H}^+ \triangleq (\mathbf{H}_z^H \mathbf{H}_z)^{-1} \mathbf{H}_z^H$  é uma matriz conhecida como a pseudo-inversa ou a inversa generalizada de Moore-Penrose de  $\mathbf{H}_z$ , tendo  $\mathbf{H}_z$  mais linhas que colunas. Se  $\mathbf{H}_z$  tem mais colunas que linhas, a pseudo-inversa é calculada por:  $\mathbf{H}^+ \triangleq \mathbf{H}_z^H (\mathbf{H}_z \mathbf{H}_z^H)^{-1}$ .

Sabemos que o atraso escolhido pode ter uma grande influência no desempenho do equalizador na presença de ruído. Diferentes atrasos podem resultar em diferentes ampliações do ruído. Para o equalizador LL-ZF, a variância do ruído na saída do equalizador é dada por (3.15), que pode ser escrita como  $\sigma_{ZF-\Delta}^2 = \sigma_n^2 \mathbf{w}_{z-\Delta}^H \mathbf{w}_{z-\Delta}$ . Logo, o atraso ótimo  $\Delta$  é aquele que resulta em um equalizador cujo  $\mathbf{w}_z$  possui a menor norma, dada por:

$$\Delta_{opt-ZF} = arg \min_{\Delta} \left\{ \left\| \mathbf{H}^+ \mathbf{h}_\Delta \right\| \right\}.\tag{3.17}$$

Portanto, considerando-se uma equalização usando o critério ZF, o atraso ótimo é aquele que seleciona a coluna da matriz pseudo-inversa do canal que possui a menor norma. Podemos fazer uma análise semelhante para o equalizador LL seguindo o critério MMSE. Dessa análise (ver apêndice B), encontramos que o atraso ótimo  $\Delta_{opt}$  é aquele que minimiza a seguinte expressão:

$$\Delta_{opt-MSE} = arg \min_{\Delta} \left\{ \mathbf{h}_\Delta^H \left( \mathbf{I} - \mathbf{H}_c (\mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I})^{-1} \mathbf{H}_c^H \right) \mathbf{h}_\Delta \right\}.\tag{3.18}$$



sendo  $\mathbf{H}_c = \begin{bmatrix} \mathbf{H}^T & \mathbf{H}^H \end{bmatrix}$  e  $\mathbf{h}_\Delta = [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0]^T$ . A expressão em (3.18) é similar à obtida para o atraso ótimo no caso de uma equalização FSE, ver Endres *et al.* (1996).

Observamos, através de diversas simulações, que se a relação sinal-ruído for alta (> 30 dB), o atraso ótimo  $\Delta$  é o mesmo que para um equalizador LL segundo o critério MMSE ou ZF, independentemente das características do canal.

Os equalizadores ótimos, segundo o critério MMSE ou ZF, podem ser obtidos iterativamente através, por exemplo, de um algoritmo do tipo LMS adaptado para o processamento largamente linear. Este será o tópico que abordaremos na próxima seção.

### 3.5 Algoritmo LMS-LL

Nesta seção, usaremos o algoritmo LMS para adaptar iterativamente os coeficientes do equalizador LL. Para isto, precisamos adequar o algoritmo LMS linear convencional para que este suporte o processamento LL. De forma geral, o algoritmo LMS para o cálculo dos coeficientes ótimos do filtro (equalizador, estimador) pode ser descrito como

$$\mathbf{w}[k+1] = \mathbf{w}[k] - \mu' \varphi(\mathbf{w}[k]), \quad (3.19)$$

onde  $\mathbf{w}[k]$  é o vetor de coeficientes do filtro largamente linear, definido como  $\mathbf{w}[k] \triangleq [\mathbf{f}^T[k] \ \mathbf{g}^T[k]]^T$ ,  $\varphi(\mathbf{w}[k])$  é uma estimativa não polarizada do gradiente do MSE no instante  $k$  e  $\mu' = \mu/2$  é uma constante escalar positiva “pequena” denotando o passo de adaptação.

O sinal de erro LL pode ser calculado por:  $e_{LL}[k] = a[k-k_0] - \mathbf{w}^H[k] \mathbf{r}_t[k]$ , onde  $\mathbf{r}_t[k] \triangleq [\mathbf{r}^T[k] \ \mathbf{r}^H[k]]^T$ . Para um dado  $\mathbf{w}$ , o MSE é calculado por:

$$\begin{aligned} J(\mathbf{w}[k]) &= E\{|e_{LL}[k]|^2\} \\ &= E\{(a[k-k_0] - \mathbf{w}^H[k] \mathbf{r}_t[k])(a[k-k_0] - \mathbf{w}^H[k] \mathbf{r}_t[k])^H\} \\ &= E\{a[k-k_0] a^*[k-k_0]\} - E\{a[k-k_0] \mathbf{r}_t^H[k] \mathbf{w}[k]\} + \\ &\quad - E\{\mathbf{w}^H[k] \mathbf{r}_t[k] a^*[k-k_0]\} + E\{\mathbf{w}^H[k] \mathbf{r}_t[k] \mathbf{r}_t^H[k] \mathbf{w}[k]\} \\ &= \sigma_a^2 - \mathbf{p}^H \mathbf{w}[k] - \mathbf{w}^H[k] \mathbf{p} + \mathbf{w}^H[k] \mathbf{R}_{LL} \mathbf{w}[k], \end{aligned} \quad (3.20)$$

onde  $\mathbf{p} = E\{\mathbf{r}_t[k]a^*[k-k_0]\}$  e  $\mathbf{R}_{LL} = E\{\mathbf{r}_t[k]\mathbf{r}_t^H[k]\}$ . O gradiente do MSE na  $k$ -ésima iteração pode ser calculado pela diferenciação de  $J(\mathbf{w}[k])$  com respeito a  $\mathbf{w}$ :

$$\nabla_{\mathbf{w}} J(\mathbf{w}) \Big|_{\mathbf{w}=\mathbf{w}[k]} = 2 \frac{\partial J}{\partial \mathbf{w}^*} = 2(-\mathbf{p} + \mathbf{R}_{LL} \mathbf{w}). \quad (3.21)$$

Vale observar que (3.21) é equivalente ao sistema descrito em (3.6). A partir de (3.6) é fácil calcular o valor ótimo para  $\mathbf{w}$ :

$$\mathbf{w}_{opt} = \mathbf{R}_{LL}^{-1} \mathbf{p}, \quad (3.22)$$

assumindo que  $\mathbf{R}_{LL}$  é inversível<sup>10</sup>. O valor de  $J_{min}$  é dado por:

$$J_{min} = \sigma_a^2 - \mathbf{p}^H \mathbf{w}_{opt} = \sigma_a^2 - \mathbf{p}^H \mathbf{R}_{LL}^{-1} \mathbf{p}. \quad (3.23)$$

O gradiente em (3.21) pode ser aproximado por uma estimativa instantânea:

$$\begin{aligned} \varphi(\mathbf{w}[k]) &= -2\mathbf{r}_t[k]a^*[k-k_0] + 2\mathbf{r}_t[k]\mathbf{r}_t^H[k]\mathbf{w}[k] \\ &= -2\mathbf{r}_t[k](a^*[k-k_0] - \mathbf{r}_t^H[k]\mathbf{w}[k]) \\ &= -2\mathbf{r}_t[k]e_{LL}^*[k]. \end{aligned} \quad (3.24)$$

Logo, o algoritmo LMS-LL pode ser expresso como (Schober *et al.*, 2004):

$$\tilde{a}_{LL}[k] = \mathbf{w}^H[k]\mathbf{r}_t[k], \quad (3.25)$$

$$e_{LL}[k] = a[k-k_0] - \tilde{a}_{LL}[k], \quad (3.26)$$

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \mu e_{LL}[k]\mathbf{r}_t[k]. \quad (3.27)$$

Como  $\mathbf{f}$  e  $\mathbf{g}$  são conjugados complexos, no algoritmo LMS-LL acima é suficiente calcularmos  $\tilde{a}_{LL}[k] = 2\Re(\mathbf{f}^H[k]\mathbf{r}[k])$ , atualizarmos os coeficientes de  $\mathbf{f}$  ( $\mathbf{f}[k+1] = \mathbf{f}[k] + \mu e_{LL}[k]\mathbf{r}[k]$ ) e, em seguida, fazer  $\mathbf{g} = \mathbf{f}^*$ , reduzindo a complexidade computacional do algoritmo e evitando erros numéricos, especialmente em implementações com precisão numérica finita. Cabe destacar que, nos casos em que o processamento largamente linear se aplica, a ordem do equalizador LL pode ser muito menor que a do equalizador linear convencional para um desempenho similar e, portanto, a complexidade computacional do algoritmo LMS-LL pode ser até menor quando comparamos com o caso estritamente linear.

<sup>10</sup> Sempre que o sinal for proveniente de um canal ruidoso,  $\mathbf{R}_{LL}$  será inversível. Note que (3.22) é equivalente as expressões (3.7) e (3.8).

### 3.6 Uma análise estatística do algoritmo LMS-LL

Nesta seção faremos uma análise estatística do algoritmo LMS-LL de passo fixo. Este tipo de estudo é importante para facilitar o projeto dos filtros e estimar o comportamento do algoritmo. Nessa análise, assumiremos que o sinal de entrada é gaussiano e de média nula, o que nos autoriza a usar o teorema da fatoração de momentos de variáveis conjuntamente gaussianas (Reed, 1962; Godara, 2004).

Os resultados desta seção combinados com parte do material das seções 3.4 e 3.5 formaram um artigo publicado no SBrT 2008 (Aquino *et al.*, 2008). Podemos encontrar uma análise estatística semelhante, mas sem o detalhamento matemático mostrado aqui, em Schober *et al.* (2004).

#### 3.6.1 Comportamento médio do vetor de coeficientes $\mathbf{w}$

O erro largamente linear no instante  $k$  é dado por:

$$e_{LL}[k] = d[k] - \mathbf{w}^H[k] \mathbf{r}_t[k], \quad (3.28)$$

sendo  $d[k]$  o sinal desejado (seqüência de treinamento). Usando a Expressão (3.28) e calculando o valor esperado em ambos os lados da Expressão (3.27), obtemos:

$$\begin{aligned} E\{\mathbf{w}[k+1]\} &= E\{\mathbf{w}[k]\} + \mu E\{\mathbf{r}_t[k] e_{LL}^*[k]\} \\ &= E\{\mathbf{w}[k]\} + \mu E\{\mathbf{r}_t[k] d^*[k]\} - \mu E\{\mathbf{r}_t[k] \mathbf{r}_t^H[k] \mathbf{w}[k]\}. \end{aligned} \quad (3.29)$$

Desprezando a dependência estatística entre os vetores de coeficientes e de sinal, podemos escrever (3.29) como

$$\begin{aligned} E\{\mathbf{w}[k+1]\} &= E\{\mathbf{w}[k]\} + \mu \mathbf{p} - \mu \mathbf{R}_{LL} E\{\mathbf{w}[k]\} \\ &= (\mathbf{I} - \mu \mathbf{R}_{LL}) E\{\mathbf{w}[k]\} + \mu \mathbf{p}, \end{aligned} \quad (3.30)$$

onde  $\mathbf{p} = E\{\mathbf{r}_t[k] d^*[k]\}$ . A Equação (3.30) é uma recursão determinística para o comportamento médio do vetor de coeficientes do algoritmo LMS-LL. Fazendo  $k \rightarrow \infty$  obtemos o valor de regime permanente desse vetor, que é dado por

$$\mathbf{w}_\infty = \lim_{k \rightarrow \infty} E\{\mathbf{w}[k]\} = \mathbf{R}_{LL}^{-1} \mathbf{p} = \mathbf{w}_{opt}. \quad (3.31)$$

Definindo o vetor de erro como

$$\boldsymbol{\varepsilon}[k] = \mathbf{w}[k] - \mathbf{w}_{opt}, \quad (3.32)$$

e substituindo para  $\mathbf{w}[k]$  em (3.27) e (3.30) nos leva a

$$\boldsymbol{\varepsilon}[k+1] = \boldsymbol{\varepsilon}[k] + \mu e_{LL}^*[k] \mathbf{r}_t[k] \quad (3.33)$$

e

$$E\{\boldsymbol{\varepsilon}[k+1]\} = (\mathbf{I} - \mu \mathbf{R}_{LL}) E\{\boldsymbol{\varepsilon}[k]\} = (\mathbf{I} - \mu \mathbf{R}_{LL})^{k+1} \boldsymbol{\varepsilon}[0], \quad (3.34)$$

que é uma recursão para o comportamento médio do vetor de erro. Usando uma transformação de similaridade para a matriz  $\mathbf{R}_{LL}$ , obtemos:

$$\begin{aligned} E\{\boldsymbol{\varepsilon}[k+1]\} &= (\mathbf{I} - \mu \mathbf{Q} \boldsymbol{\Lambda} \mathbf{Q}^H)^{k+1} \boldsymbol{\varepsilon}[0], \\ &= \mathbf{Q} (\mathbf{I} - \mu \boldsymbol{\Lambda})^{k+1} \mathbf{Q}^H \boldsymbol{\varepsilon}[0], \end{aligned} \quad (3.35a)$$

onde  $\boldsymbol{\Lambda}$  é uma matriz diagonal consistindo dos autovalores de  $\mathbf{R}_{LL}$ , associados aos autovetores da matriz unitária  $\mathbf{Q}$ . Progredindo nesse cálculo, temos também que

$$E\{\mathbf{Q}^H \boldsymbol{\varepsilon}[k+1]\} = (\mathbf{I} - \mu \boldsymbol{\Lambda})^{k+1} \mathbf{Q}^H \boldsymbol{\varepsilon}[0], \quad (3.35b)$$

e, equivalentemente,

$$E\{\mathbf{v}[k+1]\} = (\mathbf{I} - \mu \boldsymbol{\Lambda})^{k+1} \mathbf{v}[0], \quad (3.35c)$$

onde  $\mathbf{v}[k] = \mathbf{Q}^H \boldsymbol{\varepsilon}[k]$ . A partir deste equacionamento, podemos concluir que a condição de convergência do algoritmo LMS-LL é

$$0 < \mu < \frac{2}{\lambda_{max}}, \quad (3.36)$$

onde  $\lambda_{max}$  é o maior autovalor de  $\mathbf{R}_{LL}$ .

### 3.6.2 Comportamento do MSE

Usando (3.32), o erro de estimação pode ser descrito como

$$e_{LL}[k] = d[k] - (\boldsymbol{\varepsilon}[k] + \mathbf{w}_{opt})^H \mathbf{r}_t[k]. \quad (3.37)$$

A partir do equacionamento acima, podemos mostrar que:

$$\begin{aligned} |e_{LL}[k]|^2 &= d[k]d^*[k] - d[k]\mathbf{r}_t^H[k]\boldsymbol{\varepsilon}[k] - d[k]\mathbf{r}_t^H[k]\mathbf{w}_{opt} \\ &\quad - \boldsymbol{\varepsilon}^H[k]\mathbf{r}_t[k]d^*[k] + \boldsymbol{\varepsilon}^H[k]\mathbf{r}_t[k]\mathbf{r}_t^H[k]\boldsymbol{\varepsilon}[k] \\ &\quad + \boldsymbol{\varepsilon}^H[k]\mathbf{r}_t[k]\mathbf{r}_t^H[k]\mathbf{w}_{opt} - \mathbf{w}_{opt}^H\mathbf{r}_t[k]d^*[k] \\ &\quad + \mathbf{w}_{opt}^H\mathbf{r}_t[k]\mathbf{r}_t^H[k]\boldsymbol{\varepsilon}[k] + \mathbf{w}_{opt}^H\mathbf{r}_t[k]\mathbf{r}_t^H[k]\mathbf{w}_{opt}. \end{aligned} \quad (3.38)$$

Calculando o valor esperado em ambos os lados de (3.38) e desprezando as dependências estatísticas entre  $d[k]$  e  $\mathbf{r}_t[k]$  com  $\boldsymbol{\varepsilon}[k]$ , obtemos

$$\begin{aligned} E\{|e_{LL}[k]|^2\} &= \sigma_d^2 - \mathbf{p}^H E\{\boldsymbol{\varepsilon}[k]\} - \mathbf{p}^H \mathbf{w}_{opt} - E\{\boldsymbol{\varepsilon}^H[k]\}\mathbf{p} + \\ &\quad - E\{\boldsymbol{\varepsilon}^H[k]\}\mathbf{v} + \text{tr}(\mathbf{R}_{LL}\mathbf{K}[k]) + E\{\boldsymbol{\varepsilon}^H[k]\}\mathbf{R}_{LL}\mathbf{w}_{opt} + \\ &\quad - \mathbf{w}_{opt}^H\mathbf{p} + \mathbf{w}_{opt}^H\mathbf{R}_{LL}E\{\boldsymbol{\varepsilon}[k]\} + \mathbf{w}_{opt}^H\mathbf{R}_{LL}\mathbf{w}_{opt}, \end{aligned} \quad (3.39)$$

onde  $\mathbf{K}[k-1] = E\{\boldsymbol{\varepsilon}[k-1]\boldsymbol{\varepsilon}^H[k-1]\}$  é a matriz de correlação do vetor de erro  $\boldsymbol{\varepsilon}[k-1]$  e  $\text{tr}(\mathbf{M})$  é o traço da matriz  $\mathbf{M}$ . Pós-multiplicando (3.33) por sua transposta e calculando o valor esperado, encontramos

$$\begin{aligned} \mathbf{K}[k-1] &= \mathbf{K}[k-2] + \mu E\{\boldsymbol{\varepsilon}[k-2]\mathbf{r}_t^H[k-2]e_{LL}[k-2]\} + \\ &\quad + \mu E\{e_{LL}^*[k-2]\mathbf{r}_t[k-2]\boldsymbol{\varepsilon}^H[k-2]\} + \\ &\quad + \mu^2 E\{\mathbf{r}_t[k-2]e_{LL}^*[k-2]\mathbf{r}_t^H[k-2]e_{LL}[k-2]\}. \end{aligned} \quad (3.40)$$

Assumindo que  $\mathbf{r}_t[k-1]$  e  $e_{LL}[k-1]$  são conjuntamente gaussianos de média zero, o valor esperado do último termo em (3.40) pode ser calculado usando o teorema da fatoração de momentos de variáveis conjuntamente gaussianas (Reed, 1962; Godara, 2004). Novamente, desprezando a dependência estatística de  $\mathbf{r}_t[k-1]$  e  $e_{LL}[k-1]$  com respeito a  $\boldsymbol{\varepsilon}[k-2]$ , encontramos que

$$\begin{aligned} \mathbf{K}[k-1] = & \mathbf{K}[k-2] - \mu \mathbf{K}[k-2] \mathbf{R}_{LL} - \mu \mathbf{R}_{LL} \mathbf{K}[k-2] + \\ & + \mu^2 \left( \mathbf{R}_{LL} E\{\boldsymbol{\varepsilon}[k-2]\} E\{\boldsymbol{\varepsilon}^H[k-2]\} + E\{|e_{LL}[k-1]|^2\} \right) \mathbf{R}_{LL} \end{aligned} \quad (3.41)$$

onde

$$E\{|e_{LL}[k-1]|^2\} = J_{min} + tr(\mathbf{R}_{LL} \mathbf{K}[k-2]) = J_{min} + J_{ex}[k-1], \quad (3.42)$$

$J_{ex}[k-1]$  corresponde ao MSE em excesso no instante  $k-1$  e  $J_{min}$  foi definido em (3.23). Segue que (3.41) pode ser reescrita como

$$\begin{aligned} \mathbf{K}[k-1] = & \mathbf{K}[k-2] - \mu \mathbf{K}[k-2] \mathbf{R}_{LL} - \mu \mathbf{R}_{LL} \mathbf{K}[k-2] + \\ & + \mu^2 \mathbf{R}_{LL} E\{\boldsymbol{\varepsilon}[k-2]\} E\{\boldsymbol{\varepsilon}^H[k-2]\} \mathbf{R}_{LL} \\ & + \mu^2 \left( J_{min} + tr(\mathbf{R}_{LL} \mathbf{K}[k-2]) \right) \mathbf{R}_{LL} \end{aligned} \quad (3.43)$$

Logo, a Expressão (3.43) é uma forma recursiva para o cálculo de matriz de correlação do vetor de erro. Calculando o limite para  $k \rightarrow \infty$  em (3.42), encontramos o erro médio quadrático de regime permanente:

$$J_{\infty} = \lim_{k \rightarrow \infty} E\{|e_{LL}[k-1]|^2\} = J_{min} + tr(\mathbf{R}_{LL} \mathbf{K}_{\infty}) = J_{min} + J_{ex}[\infty] \quad (3.44)$$

onde  $\mathbf{K}_{\infty} = \lim_{k \rightarrow \infty} \mathbf{K}[k-1]$ . A partir de (3.43), obtemos

$$J_{ex}[\infty] = tr(\mathbf{R}_{LL} \mathbf{K}_{\infty}) = \frac{\mu J_{min} tr(\mathbf{R}_{LL})}{2 - \mu J_{min} tr(\mathbf{R}_{LL})} \quad (3.45)$$

e

$$J_{\infty} = J_{min} \left( 1 + \frac{\mu tr(\mathbf{R}_{LL})}{2 - \mu tr(\mathbf{R}_{LL})} \right). \quad (3.46)$$

Finalmente, o desajuste é dado por:

$$\xi = \frac{J_{ex}[\infty]}{J_{min}} = \frac{\mu tr(\mathbf{R}_{LL})}{2 - \mu tr(\mathbf{R}_{LL})}. \quad (3.47)$$

Como o desajuste  $\xi$  precisa ser finito, temos o seguinte limite superior de  $\mu$  para garantir a estabilidade do algoritmo LMS largamente linear:

$$\mu_{max} < \frac{2}{tr(\mathbf{R}_{LL})}. \quad (3.48)$$

A Tabela 3.1 resume o modelo desenvolvido.

TABELA 3.1. MODELO LMS LARGAMENTE LINEAR

---



---

*Inicialização:*

$$\mathbf{w}[0] = \mathbf{0}, \boldsymbol{\varepsilon}[0] = -\mathbf{w}_{opt}, 0 < \mu < \frac{2}{tr(\mathbf{R}_{LL})}, \mathbf{K}[0] = \mathbf{w}_{opt} \mathbf{w}_{opt}^H$$


---

*Regime permanente:*

$$\mathbf{w}_{\infty} = \mathbf{w}_{opt}, J_{\infty} = J_{min} + J_{ex}[\infty], \xi = \frac{\mu tr(\mathbf{R}_{LL})}{2 - \mu tr(\mathbf{R}_{LL})}, \boldsymbol{\varepsilon}[\infty] = \mathbf{0}$$


---



---

*Comportamento transiente:*

$$E\{\mathbf{w}[k+1]\} = (\mathbf{I} - \mu \mathbf{R}_{LL}) E\{\mathbf{w}[k]\} + \mu \mathbf{p}$$

$$E\{\boldsymbol{\varepsilon}[k+1]\} = (\mathbf{I} - \mu \mathbf{R}_{LL})^{k+1} \boldsymbol{\varepsilon}[0]$$

$$\begin{aligned} \mathbf{K}[k-1] &= \mathbf{K}[k-2] - \mu \mathbf{K}[k-2] \mathbf{R}_{LL} - \mu \mathbf{R}_{LL} \mathbf{K}[k-2] + \\ &+ \mu^2 \mathbf{R}_{LL} E\{\boldsymbol{\varepsilon}[k-2]\} E\{\boldsymbol{\varepsilon}^H[k-2]\} \mathbf{R}_{LL} \\ &+ \mu^2 (J_{min} + tr(\mathbf{R}_{LL} \mathbf{K}[k-2])) \mathbf{R}_{LL} \end{aligned}$$

$$J_{ex}[k] = tr(\mathbf{R}_{LL} \mathbf{K}[k-1])$$

$$E\{|e_{LL}[k]|^2\} = J_{min} + J_{ex}[k]$$


---



---

### 3.7 Algoritmo LMS-LL de passo variável

A escolha do passo de adaptação  $\mu$  reflete um compromisso entre velocidade de convergência e o desajuste entre a estimativa e o sinal desejado. O simples aumento do valor do passo de adaptação pode acelerar a convergência do algoritmo, mas leva a um desajuste final também maior, além de poder levar a uma instabilidade se o passo adotado for maior que um certo valor máximo, como mostrado na seção anterior. Da mesma forma que o algoritmo LMS convencional, o algoritmo LMS-LL também pode ter o seu desempenho otimizado em relação à taxa de convergência ou em relação ao erro de estado permanente com o uso de técnicas de passo variável (Aquino *et al.*, 2006a).

A normalização do passo pela energia do vetor de sinal, técnica proposta por Nagumo e Noda (1967), é uma forma eficiente de acelerar a convergência do algoritmo LMS, tendo um custo computacional relativamente baixo. É necessário modificar apenas a Equação (3.27) do algoritmo LMS-LL para termos a sua versão normalizada (LMS-LL-N):

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \mu_N e_{LL}^*[k] \mathbf{r}_t[k] / (\|\mathbf{r}_t[k]\|^2 + \varepsilon), \quad (3.49)$$

onde  $0 < \mu_N < 1$  é o passo normalizado e  $\varepsilon$  é uma constante positiva pequena. Naturalmente, as várias técnicas de passo variável já conhecidas na literatura (ver, por exemplo, o trabalho de Zipf *et al.*, 2007) poderiam ser empregadas. Entretanto, o ganho na velocidade de convergência ou no erro em regime permanente, em relação à simples normalização do passo, considerando o caso de um equalizador largamente linear com poucos coeficientes, não é muito significativo (Aquino *et al.*, 2006a).

Vale a pena ressaltar que nas aplicações estudadas nesta tese, o número de coeficientes é relativamente pequeno e podemos, de fato, esperar um ganho de desempenho do algoritmo LMS-LL normalizado em relação a sua versão de passo fixo. Para filtros com muitos coeficientes, o desempenho dos algoritmos LMS e LMS-N são similares para uma escolha apropriada do passo de adaptação (Costa e Bermudez, 2002).

### 3.8 Equalizador *Multi-Split* largamente linear

A partir do trabalho seminal de Delsarte e Genin (1986), que apresentava uma versão *split* do algoritmo de Levinson para matrizes Toeplitz reais, o processamento *split* ganhou um interesse crescente. Posteriormente, o processamento *split* foi expandido para outras áreas por diversos autores. O processamento *multi-split* (MS) foi introduzido por Resende *et al.* somente em 2001. Naquele trabalho foi criada uma nova estrutura para o filtro transversal *split*.

Nesta seção usamos a filtragem *multi-split* com parâmetros complexos aplicada ao algoritmo LMS largamente linear. O algoritmo LMS-LL-MS (LMS-LL *multi-split*) pode apresentar uma taxa de convergência maior que o LMS-LL normalizado com um custo computacional adicional pequeno. Esta seção é baseada nos nossos trabalhos apresentados nos congressos SBrT-2007 (Aquino *et al.*, 2007b) e GRETSI-2007 (Aquino *et al.*, 2007d).

#### 3.8.1 Transformada *Multi-Split* e algoritmo LMS

Inicialmente, consideramos o esquema clássico de filtragem transversal sendo o número de coeficientes do filtro FIR igual a  $N=2^L$ , com  $L \geq 1$ , como mostra a Figura 3.5-(a). Aplicando sucessivamente um processo de divisão da resposta ao impulso do filtro em suas partes simétrica e anti-simétrica, obtemos o esquema de filtragem mostrado na Figura 3.5-(b) (Resende *et al.*, 2004), onde



$$\mathbf{M}_N = \begin{bmatrix} \mathbf{M}_{N/2} & \mathbf{J}_{N/2}\mathbf{M}_{N/2} \\ \mathbf{J}_{N/2}\mathbf{M}_{N/2} & -\mathbf{M}_{N/2} \end{bmatrix}. \quad (3.50)$$

$\mathbf{J}_{N/2}$  é uma matriz de troca (ou matriz identidade anti-diagonal), isto é, é uma matriz cujos elementos da diagonal secundária são unitários e os demais são nulos,  $\mathbf{M}_1=[1]$  e  $w_{\perp i}$ , para  $i = 0, 1, \dots, N-1$ , são os coeficientes dos filtros de ordem zero conectados em paralelo, como mostra a Figura 3.5-(b). É fácil verificar que  $\mathbf{M}_N$  é uma matriz composta somente por +1's e -1's, e que o produto interno de duas colunas distintas quaisquer é zero. De fato,  $\mathbf{M}_N$  é uma matriz não singular e  $\mathbf{M}_N^T\mathbf{M}_N = \mathbf{I}_N$ , onde  $\mathbf{I}_N$  é uma matriz identidade  $N \times N$ .

Podemos visualizar o esquema *multi-split* como uma transformação linear dos dados de entrada dada por

$$\mathbf{r}_{\perp}[k] = \mathbf{M}_N \mathbf{r}[k], \quad (3.51)$$

onde  $\mathbf{r}[k] = [r[k] \ r[k-1] \ \dots \ r[k-N+1]]^T$  indica o vetor de entrada e  $\mathbf{r}_{\perp}[k] = [r_{\perp 0}[k] \ r_{\perp 1}[k] \ \dots \ r_{\perp N-1}[k]]^T$  o vetor transformado. A transformação em (3.51) pode ser eficientemente realizada por uma estrutura em borboleta que é facilmente implementável em *hardware*, pois só possui operações de somas e subtrações, como podemos visualizar na Figura 3.5-(c).

A transformação *multi-split* não reduz o espalhamento dos autovalores da matriz de correlação do sinal, mas ela pode aumentar o seu fator de diagonalização (Resende *et al.*, 2004). Dessa forma, podemos esperar um aumento na taxa de convergência do algoritmo LMS em relação à sua versão tradicional.

Wan e Ching (1997) identificaram que a matriz  $\mathbf{M}_N$  do procedimento MS é muito semelhante à matriz Walsh-Hadamard, exceto pela permutação de algumas linhas. Aqueles autores também mostraram que o desempenho de um filtro adaptativo usando o procedimento MS ou a transformada discreta de Walsh-Hadamard é exatamente o mesmo.

Dentro do contexto adaptativo, podemos usar o esquema MS em um algoritmo LMS de passo variável que explore a natureza da matriz transformada de correlação dos dados de entrada e atualize os coeficientes independentemente. Os coeficientes dos filtros de ordem zero podem ser complexos, mas isso não altera o procedimento *multi-split*, pois a resposta ao impulso do filtro pode ser decomposta em suas partes simétricas e anti-simétricas conjugadas (Souza *et al.*, 2005).

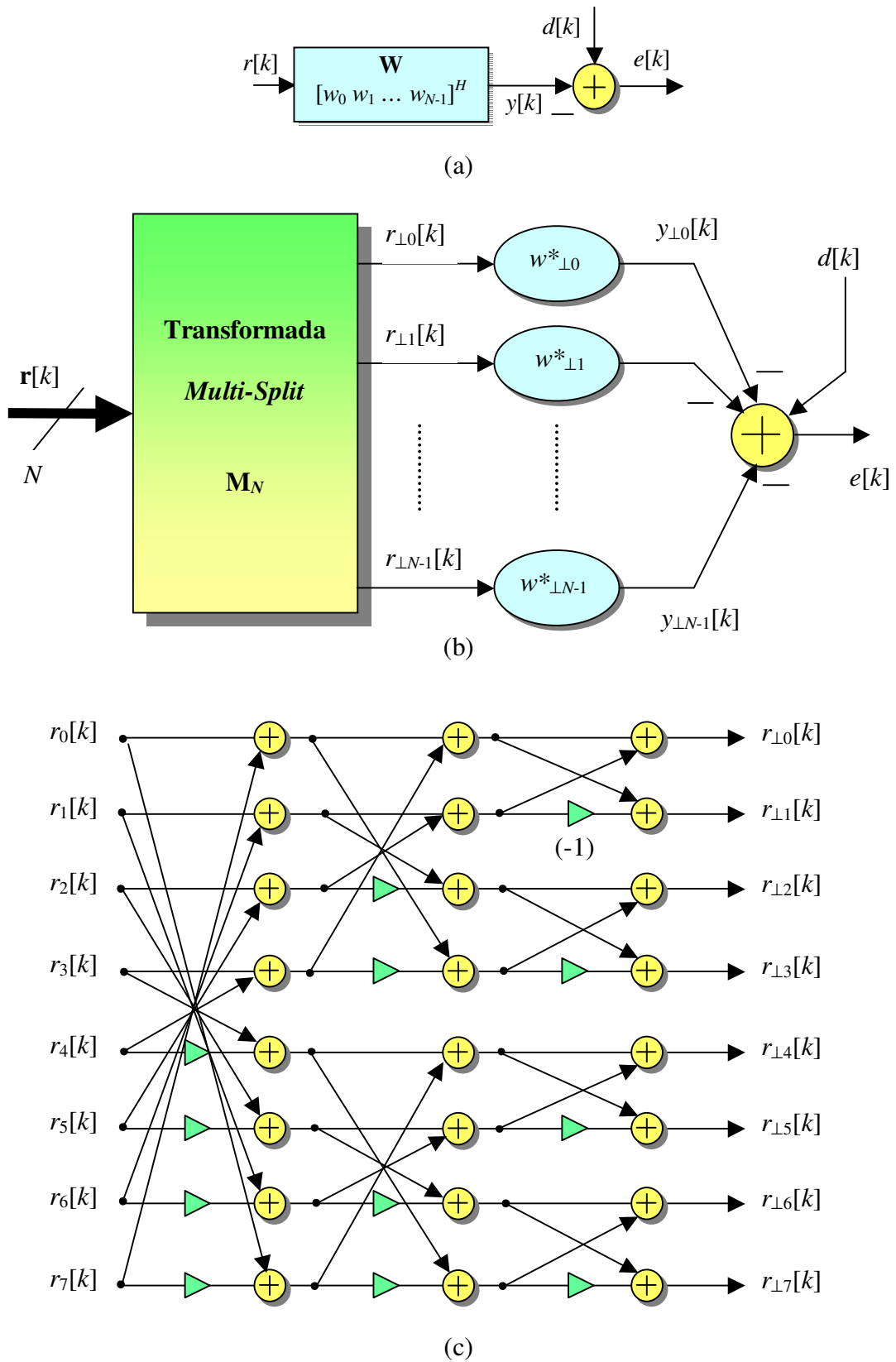


Figura 3.5: (a) Filtro transversal, (b) Filtragem *multi-split*, (c) Estrutura em borboleta para uma implementação *multi-split* eficiente

Em síntese, usando a transformação MS, temos um conjunto de  $N$  filtros de ordem zero em paralelo. Cada um desses filtros pode ser adaptado por um algoritmo de passo variável normalizado pela potência. Assim, podemos descrever o algoritmo *multi-split* LMS normalizado como

$$w_{\perp i}[k+1] = w_{\perp i}[k] + \frac{\mu_{MS}}{s_i[k]} r_{\perp i}[k] e^*[k], \quad (3.52)$$

onde

$$s_i[k] = \frac{\sum_{j=1}^k \gamma^{k-j} |r_{\perp i}[j]|^2}{\sum_{j=1}^k \gamma^{k-j}} = \frac{p_i[k]}{q_i[k]}, \quad (3.53)$$

para  $i=0, 1, \dots, N-1$ ,  $s_i[k]$  é uma estimativa da variância de  $r_{\perp i}$ , sendo  $p_i[k]$  e  $q_i[k]$  calculados recursivamente por:

$$p_i[k] = |r_{\perp i}[k-1]|^2 + \gamma p_i[k-1], \quad (3.54)$$

e

$$q_i[k] = 1 + \gamma q_i[k-1], \quad (3.55)$$

$$e[k] = d[k] - y[k], \quad (3.56)$$

$$y[k] = \sum_{i=0}^{N-1} r_{\perp i}[k] w_{\perp i}^*[k], \quad (3.57)$$

nas equações acima  $\mu_{MS}$  indica o passo de adaptação e  $\gamma$  ( $0 << \gamma \leq 1$ ) o fator de esquecimento. Para um ambiente estacionário no sentido amplo, o fator de esquecimento pode ser unitário. Uma forma mais “econômica” para o cálculo da variância de  $r_{\perp i}$  é dada simplesmente por (Goupil e Palicot, 2002):

$$s_i[k] = \gamma s_i[k-1] + (1-\gamma) |r_{\perp i}[k]|^2, \quad (3.58)$$

com o fator de esquecimento  $\gamma$  próximo à unidade. As estimativas dadas por (3.53) e (3.58) se confundem quando  $k \rightarrow \infty$ , como mostramos no Apêndice **B**.

A aplicação da transformada *multi-split* ao algoritmo LMS largamente linear é direta e muito semelhante a sua aplicação ao algoritmo LMS convencional. A Tabela 3.2 descreve o algoritmo LMS largamente linear *multi-split* (LMS-LL-MS).

TABELA 3.2: ALGORITMO LMS-LL-MS

1) <i>Transformação Linear</i> : $\mathbf{r}_\perp[k] = \mathbf{M}_N \mathbf{r}[k]$
2) <i>Atualização</i> :
$y[k] = \mathbf{f}_\perp^H[k] \mathbf{r}_\perp[k] + \mathbf{g}_\perp^H[k] \mathbf{r}^*_\perp[k]$
$e[k] = d[k] - y[k]$
$s_i[k] = p_i[k] / q_i[k]$ (ver equações (3.53) a (3.35))
$f_{\perp i}[k] = f_{\perp i}[k-1] + \mu_{MS} e^*[k] r_{\perp i}[k] / s_i[k]$
$\mathbf{g}_\perp[k] = \mathbf{f}^*_\perp[k]$

### 3.8.2 Análise da matriz de correlação largamente linear transformada

Agora, analisaremos o efeito da aplicação da transformada *multi-split* ao processamento largamente linear em relação ao fator de diagonalização e do espalhamento dos autovalores da matriz de correlação largamente linear.

Podemos escrever a matriz  $\mathbf{R}_{LL}$  como:

$$\mathbf{R}_{LL} = \begin{bmatrix} \mathbf{\Gamma} & \mathbf{C} \\ \mathbf{C}^* & \mathbf{\Gamma}^* \end{bmatrix} \quad (3.59)$$

onde  $\mathbf{\Gamma} = E\{\mathbf{x}(n)\mathbf{x}(n)^H\}$  e  $\mathbf{C} = E\{\mathbf{x}(n)\mathbf{x}(n)^T\}$ . A aplicação da transformada *multi-split* aos dados de entrada leva às seguintes expressões:

$$\mathbf{\Gamma}_\perp = E\{\mathbf{M}_N^T \mathbf{x}[k] \mathbf{x}^H[k] \mathbf{M}_N\} = 2^N \mathbf{M}_N^{-1} \mathbf{\Gamma} \mathbf{M}_N, \quad (3.60)$$

$$\mathbf{C}_\perp = E\{\mathbf{M}_N^T \mathbf{x}[k] \mathbf{x}^T[k] \mathbf{M}_N\} = 2^N \mathbf{M}_N^{-1} \mathbf{C} \mathbf{M}_N, \quad (3.61)$$

e a matriz de correlação transformada é dada por

$$\mathbf{R}_\perp = \begin{bmatrix} \mathbf{\Gamma}_\perp & \mathbf{C}_\perp \\ \mathbf{C}_\perp^* & \mathbf{\Gamma}_\perp^* \end{bmatrix}. \quad (3.62)$$

Considerando o caso em que cada filtro largamente linear tenha somente dois coeficientes, teremos

$$\mathbf{\Gamma}_{\perp} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} r_{00} & r_{01} \\ r_{10} & r_{11} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} r_{00} + r_{01} + r_{10} + r_{11} & r_{00} - r_{01} + r_{10} - r_{11} \\ r_{00} + r_{01} - r_{10} - r_{11} & r_{00} - r_{01} - r_{10} + r_{11} \end{bmatrix}. \quad (3.63)$$

Desde que  $r_{00} = r_{11}$  e  $r_{01} = r_{10}^*$ , podemos reescrever  $\mathbf{\Gamma}_{\perp}$  como

$$\mathbf{\Gamma}_{\perp} = 2 \begin{bmatrix} r_{00} + \text{Re}(r_{01}) & -j\text{Im}(r_{01}) \\ j\text{Im}(r_{01}) & r_{00} - \text{Re}(r_{01}) \end{bmatrix}. \quad (3.64)$$

De forma semelhante,  $\mathbf{C}_{\perp}$  é dado por

$$\mathbf{C}_{\perp} = 2 \begin{bmatrix} c_{00} + c_{01} & 0 \\ 0 & c_{00} - c_{01} \end{bmatrix}, \quad (3.65)$$

e, portanto,

$$\mathbf{R}_{\perp} = 2 \begin{bmatrix} r_{00} + \text{Re}(r_{01}) & -j\text{Im}(r_{01}) & c_{00} + c_{01} & 0 \\ j\text{Im}(r_{01}) & r_{00} - \text{Re}(r_{01}) & 0 & c_{00} - c_{01} \\ c_{00}^* + c_{01}^* & 0 & r_{00} + \text{Re}(r_{01}) & j\text{Im}(r_{01}) \\ 0 & c_{00}^* - c_{01}^* & -j\text{Im}(r_{01}) & r_{00} - \text{Re}(r_{01}) \end{bmatrix}. \quad (3.66)$$

Então, podemos verificar que o fator de diagonalização de  $\mathbf{R}_{\perp}$ , definido por (Wan e Ching, 1992):

$$\gamma(\mathbf{R}_{\perp}) = \frac{\text{tr}(\mathbf{R}_{\perp})}{\sum_i \sum_j |\mathbf{R}_{\perp}(i, j)| - \text{tr}(\mathbf{R}_{\perp})}, \quad (3.67)$$

é aumentado quando comparado com o fator de diagonalização de  $\mathbf{R}_{LL}$ , especialmente quando  $N \leq 16$ . No apêndice **B** apresentamos alguns exemplos numéricos desse aumento no fator de diagonalização. Ainda podemos verificar que  $\mathbf{R}_{\perp} = 2^N \mathbf{K}^{-1} \mathbf{R}_{LL} \mathbf{K}$ , onde

$$\mathbf{K} = \begin{bmatrix} \mathbf{M}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_N \end{bmatrix}, \quad (3.68)$$

o que indica que as matrizes  $\mathbf{R}_{\perp}$  e  $\mathbf{R}_{LL}$  são similares e, conseqüentemente,  $\mathbf{R}_{\perp}$  e  $\mathbf{R}_{LL}$  têm o mesmo espalhamento de autovalores. Logo, está justificado o uso da transformada *multi-split* no processamento largamente linear.

### 3.9 Implementação em FPGA do equalizador largamente linear

Nesta seção, apresentamos alguns conceitos e um breve histórico sobre a tecnologia FPGA (*Field Programmable Gate Arrays*) e uma implementação nessa tecnologia de um equalizador largamente linear de passo fixo. Comparamos o desempenho deste equalizador com um equalizador linear também de passo fixo. Esta seção é baseada nos nossos trabalhos apresentados no SBrT-2007 (Vargas *et al.*, 2007) e I2TS-2006 (Aquino *et al.*, 2006b). O leitor interessado poderá obter informações mais detalhadas, em português, sobre a tecnologia FPGA nas dissertações de mestrado de Cesar H. Vargas (Vargas, 2007) e Bruno Chang (Chang, 2008).

Convém observar que na implementação prática, vários fatores que, em geral, não são considerados nos projetos (puramente) teóricos começam a ter uma importância fundamental: a precisão numérica (número de bits), a latência das operações matemáticas, a área ocupada em silício, memória disponível, o consumo de energia, entre outros.

#### 3.9.1 A tecnologia FPGA

A tecnologia de dispositivos FPGA foi introduzida nos anos 1980 como um dispositivo para processamento de lógica digital. As FPGAs foram projetadas utilizando circuitos multi-nível, que poderiam incluir circuitos complexos em uma única pastilha. As FPGAs são pré-fabricadas e, portanto, são de uso fácil e de baixo custo. Além disso, dado que elas são facilmente reconfiguradas e permitem a correção de erros (*bugs*) de sistemas em desenvolvimento, as FPGAs são ideais para prototipagem (Reed, 2002; Hauck, 1998).

Vale a pena lembrar que o projetista também tem à sua disposição duas outras categorias de *hardware* além das FPGAs: ASICs (*application specific integrated circuits*) e DSP's (*digital signal processor*). Enquanto um DSP representa o tipo de *hardware* mais generalizado que pode ser programado para desempenhar várias funções, um ASIC é um circuito especializado e só pode ser usado para desempenhar a função para qual foi projetado.

Nos últimos anos, a tecnologia de dispositivos FPGAs tem evoluído significativamente, alcançando elevados níveis de densidade, altos índices de desempenho e menores custos de fabricação, como indica a Figura 3.6. Esta evolução tem tornado cada vez menor a distância entre FPGAs e CIs (Circuitos Integrados) para fins específicos (ASICs).

Além dos avanços em capacidade, desempenho e custos, os fabricantes de FPGAs têm introduzido, no decorrer dos anos, cada vez mais recursos de reconfigurabilidade.

Os recursos de reconfigurabilidade recentemente implantados pelos fabricantes de FPGAs têm possibilitado o projeto de sistemas dinamicamente reconfiguráveis. O termo “dinamicamente reconfigurável” indica a possibilidade de se alterar parcialmente a funcionalidade de um dispositivo sem prejudicar o funcionamento de sua lógica restante, que pode estar em operação (Lysaght e Dunlop, 1993).

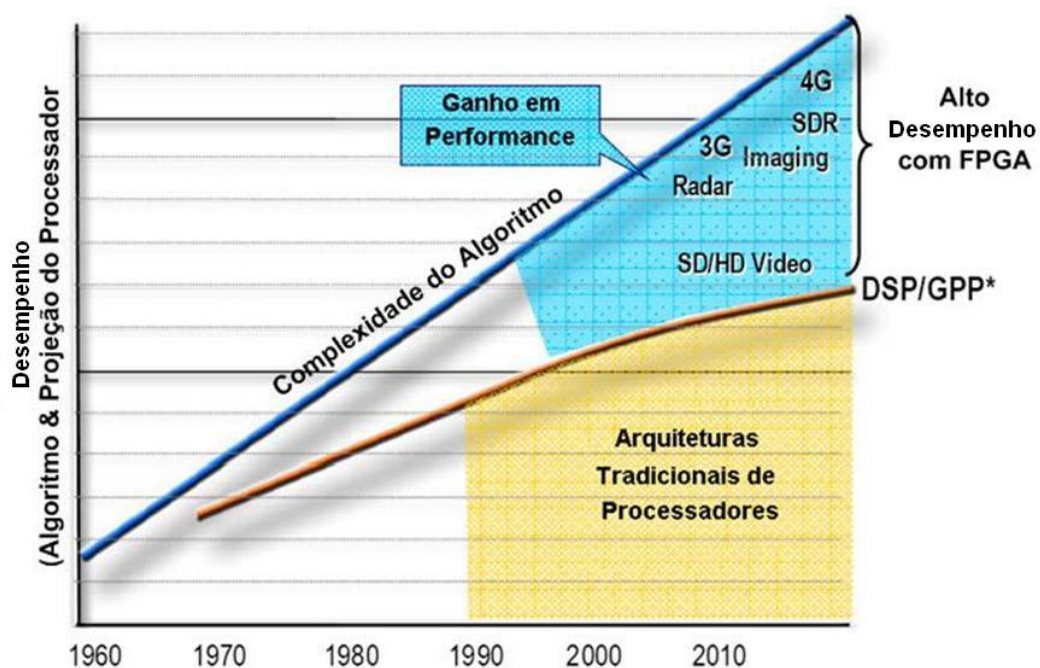


Figura 3.6: Desempenho exigido em função do tempo, adaptado de XILINX Inc. (2005b).

Existem vários fabricantes de FPGAs, entre eles ALTERA, Achronix Semiconductor e XILINX. Em nossas implementações usamos exclusivamente uma placa FPGA da família Virtex-4 produzida pela XILINX. Esta família compreende o seguinte conjunto de FPGAs:

- Virtex-4 LX: aplicações de alto desempenho.
- Virtex-4 SX: aplicações DSP (*Digital Signal Processing*) de alto desempenho.
- Virtex-4 FX: solução completa e de alto desempenho para aplicações de plataformas embarcadas.

Os componentes da Virtex-4 são uma evolução dos componentes já existentes em outras famílias (Virtex, Virtex-E, Virtex-2, Virtex-2 Pro e Virtex-2 Pro X) (Hwang *et al.*, 2001; XILINX Inc, 2006).

O *System Generator* é uma ferramenta de projeto integrado, em nível de sistema, para FPGAs, que utiliza o Simulink™ (Matlab®), como suporte de desenvolvimento e é apresentado em forma de uma biblioteca (*blockset*) (XILINX Inc., 2006). Podemos realizar todo o projeto de um circuito em FPGA no ambiente Simulink, como mostra a Figura 3.7.

Como ilustrado na Figura 3.8, o *System Generator*, através de co-simulação, gera um arquivo de configuração (\*.bit) necessário para a programação da FPGA (XILINX Inc., 2005). O projeto utilizando *System Generator* facilita a implementação dos equalizadores em FPGA, pois todos os detalhes de implementação são realizados nesta ferramenta e sem a necessidade de se estar conectada à placa FPGA.

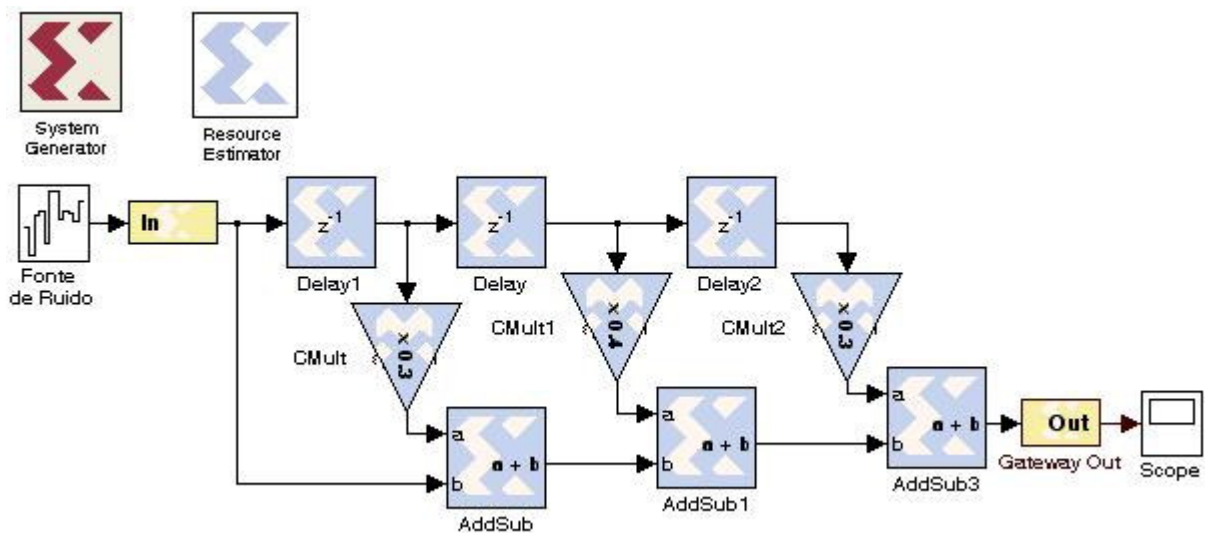


Figura 3.7: Exemplo de um projeto em FPGA usando *System Generator* em ambiente Simulink.

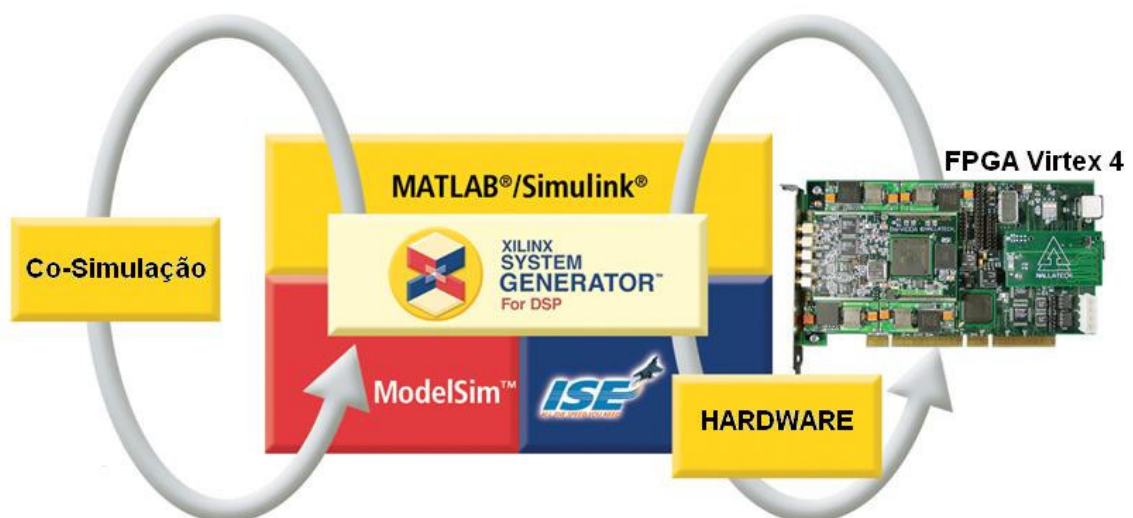


Figura 3.8: Fluxo de projeto em *Xilinx System Generator*, adaptado de XILINX Inc. (2005b).



Convém lembrar que o *System Generator* não tem por objetivo substituir o VHDL ou o Verilog, mas o seu uso leva a um ganho de produtividade ao reduzir o tempo gasto nos problemas de linguagem de programação e permitir que o foco fique sobre outras dificuldades (XILINX Inc., 2005).

### 3.9.2 Implementação de filtros FIR em FPGA

Uma ampla variedade de arquiteturas de filtros é disponível em FPGA, devido à flexibilidade desta tecnologia. O tipo de estrutura é determinado tipicamente pela quantidade de processamento requerida no número de ciclos de relógio disponível. Os fatores mais importantes são: a taxa de amostragem e o número de coeficientes. Além desses dois fatores, o projetista precisa estar atento à sincronização das operações, pois as operações matemáticas (multiplicações, somas) podem exigir um tempo (latência) para serem efetuadas.

A Figura 3.9 mostra como o incremento na taxa de amostragem e o incremento no número de coeficientes do filtro influenciam na escolha de uma determinada estrutura. A Tabela 3.3 mostra as relações que permitem calcular a taxa de amostragem do sinal de entrada para o filtro FIR (XILINX Inc., 2005).

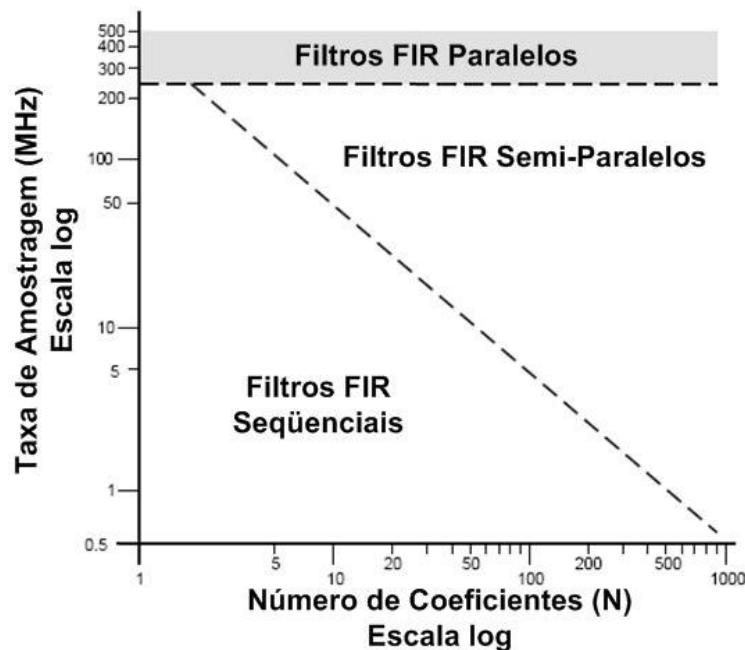


Figura 3.9: Comparação das estruturas dos filtros em relação à taxa de amostragem e número de coeficientes, adaptado de XILINX Inc. (2005b).

TABELA 3.3: TAXA DE AMOSTRAGEM PARA AS ESTRUTURAS FIR

Estrutura	Taxa de amostragem
<b>Paralela</b>	$Velc. do relógio$
<b>Seqüencial</b>	$\frac{Velc. do relógio}{Num. de coef.}$
<b>Semi - Paralela</b>	$\frac{Velc. do relógio}{Num. de coef.} \times Num. de Mult.$

Utilizando uma das ferramentas do *System Generator*, realizamos uma estimaco de rea em FPGA para um equalizador largamente linear com quatro coeficientes complexos. A Tabela 3.4 mostra os resultados obtidos. Como podemos observar, a estrutura seqüencial e semi-paralela consomem um menor nmero de recursos (*slices*, *flip-flops*, LUT's, etc), conseqüentemente, ocupam uma menor rea em FPGA que a estrutura paralela. Vale a pena ressaltar que os resultados mostrados na Tabela 3.4 dependem da preciso numrica escolhida para os elementos (somadores, multiplicadores) utilizados para o projeto dos equalizadores.

A rea ocupada pelas estruturas seqüencial e semi-paralela  uma funo no-linear do nmero de coeficientes, isto , a quantidade de recursos demandados cresce lentamente com o aumento do nmero de coeficientes do filtro, isso  especialmente verdade em relao ao nmero de flip-flops, como mostra a Figura 3.10. J na estrutura paralela, a demanda por recursos da FPGA  maior, pois para cada novo coeficiente do filtro  necessrio acrescentar novos elementos (multiplicadores, somadores, etc).

TABELA 3.4: ESTIMAO DE RECURSOS DAS ESTRUTURAS IMPLEMENTADAS EM FPGA

<b>FPGA Virtex 4 xc4vsx35 – 10ff668</b>			
	<b>Filtro FIR Seqüencial</b>	<b>Filtro FIR Semi-Paralelo</b>	<b>Filtro FIR Paralelo</b>
Slices	797	623	4288
Flip Flops	882	594	6918
BRAMs	2	4	0
LUTs	1235	875	6687
IOBs	110	141	98
Bem. Mults.	16	10	16
TBUFs	0	0	0

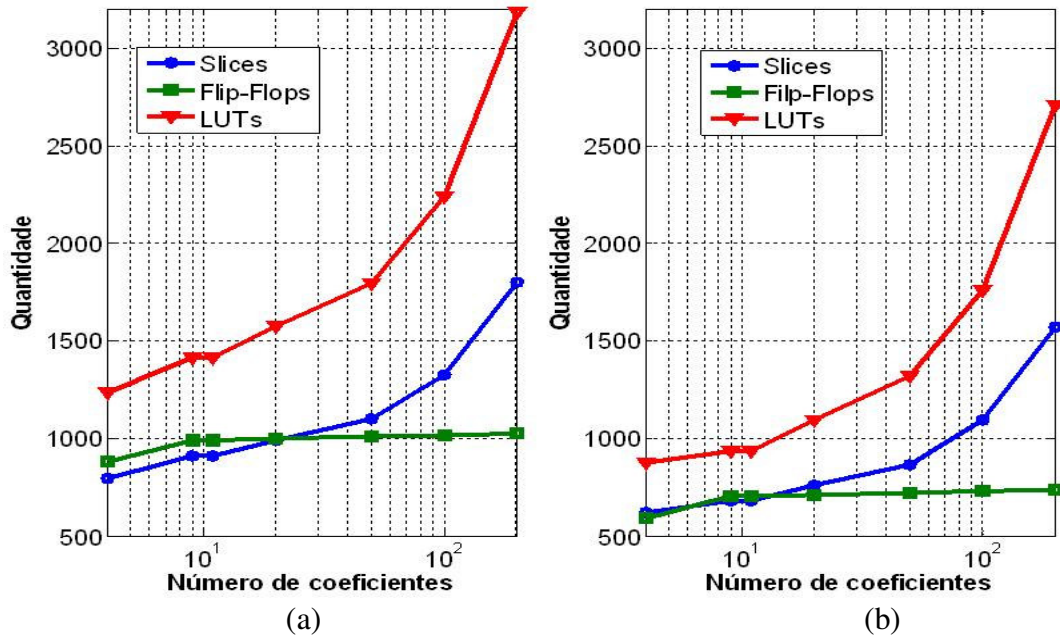


Figura 3.10 Demanda de recursos com número de coeficientes para as estruturas (a) sequencial e (b) semi-paralela.

Como uma última observação, destacamos que o algoritmo LMS implementado para a versão paralela é do tipo LMS atrasado (Delayed LMS - DLMS), devido à latência das multiplicações. Um algoritmo do tipo DLMS exige que o passo de adaptação  $\mu$  seja menor em relação a sua versão não atrasada para garantir a estabilidade (Long *et al.*, 1989). Tobias *et al.* (2000) apresentam o seguinte limitante superior para o passo de adaptação do algoritmo DLMS aplicado a um equalizador linear:

$$\mu < \frac{2}{2(D+1)\lambda_{\max} + \sum_{j=1}^N \lambda_j}, \quad (3.69)$$

onde  $D$  é a latência das operações de multiplicação,  $\mu$  o passo de adaptação do algoritmo e  $\lambda_{\max}$  o maior autovalor da matriz de autocorrelação dos dados de entrada. No Apêndice **B**, mostramos o efeito da latência nas multiplicações em um exemplo bem simples de um equalizador linear com apenas um coeficiente.

Apresentaremos os resultados da comparação destas estruturas na equalização de um canal FIR no final da seção de resultados de simulação.

### 3.10 Resultados de simulação

Em todos os resultados de simulação apresentados nesta seção a seqüência de símbolos transmitidos é proveniente de constelações reais, i.i.d. e de média nula, a menos que indicado de forma diferente. O ruído AWGN (complexo, próprio, de média nula), é adicionado ao sinal de saída do canal de modo a satisfazer uma certa razão sinal-ruído. Queremos com estes inúmeros exemplos ilustrar os principais aspectos teóricos e práticos discutidos ao longo desse capítulo.

#### 3.10.1 Equalização LL ótima.

*Exemplo 1. Canal com apenas um zero.* Considere um canal complexo com apenas um zero ( $L = 1$ ) definido por  $\mathbf{h}_c = [0,25-0,40j; -0,34+0,01j]$  e uma relação sinal-ruído de 30 dB,  $\xi = 10^{-3}$  e uma constelação binária (BPSK). Para a equalização deste canal, usaremos um atraso de equalização  $k_0 = 1$  e um equalizador de ordem  $P = 1$ , ou seja, os filtros  $\mathbf{f}$  e  $\mathbf{g}$  têm apenas dois coeficientes cada. A partir de (3.6) - (3.8), teremos os seguintes vetores e matrizes complexos:

$$\mathbf{h} = \begin{bmatrix} -0,34+0,01j \\ 0,25-0,40j \end{bmatrix}, \mathbf{H} = \begin{bmatrix} 0,25-0,40j & -0,34+0,01j & 0 \\ 0 & 0,25-0,40j & -0,34+0,01j \end{bmatrix}, \mathbf{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Efetuando os cálculos indicados em (3.8), obtemos:

$$\mathbf{f} = \begin{bmatrix} -0,5111+0,3184j \\ -0,0221+0,8348j \end{bmatrix} \text{ e } \mathbf{g} = \begin{bmatrix} -0,5111-0,3184j \\ -0,0221-0,8348j \end{bmatrix}.$$

Logo, a resposta ao impulso canal+equalizador LL será:

$$\mathbf{h}_{\text{MSE}} = \mathbf{h}_c \star \mathbf{f}^* + \mathbf{h}_c^* \star \mathbf{g}^* = [-0,0008 \quad 0,9979 \quad -0,0016],$$

onde  $\star$  indica a operação de convolução. Já pelo critério ZF podemos equacionar, como indica (3.14):

$$\begin{bmatrix} 0,25-0,40j & 0,25+0,40j & 0 & 0 \\ -0,34+0,01j & -0,34-0,01j & 0,25-0,40j & 0,25+0,40j \\ 0 & 0 & -0,34+0,01j & -0,34-0,01j \end{bmatrix} \begin{bmatrix} f_0^* \\ g_0^* \\ f_1^* \\ g_1^* \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix},$$

calculando a pseudo-inversa, obtemos os seguintes coeficientes:

$$\begin{bmatrix} f_0^* \\ g_0^* \\ f_1^* \\ g_1^* \end{bmatrix} = \begin{bmatrix} -0,5125 + 0,3203j \\ -0,5125 - 0,3203j \\ -0,0246 + 0,8378j \\ -0,0246 - 0,8378j \end{bmatrix}.$$

A resposta ao impulso é, como esperado, um impulso atrasado:  $\mathbf{h}_{ZF} = \mathbf{h}_c \star \mathbf{f}^* + \mathbf{h}_c^* \star \mathbf{g}^* = [0 \ 1,0000 \ 0]$ . Como a relação sinal-ruído é alta, os coeficientes do equalizador calculados pelo critério ZF e pelo critério MSE estão bem próximos.

*Exemplo 2. Canal com nulo spectral.* Faremos agora um exemplo mais interessante. Consideraremos o canal complexo  $\mathbf{h}_c = [0,6099+0,1830j; \ 0,1037+0,0976j; \ 0,0238-0,6776j; \ 0,2016+0,2719j]$ . Este canal possui um nulo spectral e um zero fora do CRU, como mostra a Figura 3.11(a). Para equalizarmos este canal, usaremos um equalizador LL com filtros  $\mathbf{f}$  e  $\mathbf{g}$  da mesma ordem do canal ( $P = 3$ , filtros  $\mathbf{f}$  e  $\mathbf{g}$  com 4 coeficientes cada), atraso  $k_0 = 2$  e o critério MSE. Novamente, estamos considerando uma relação sinal-ruído de 30 dB. A constelação usada é 4-PAM. Efetuando os cálculos indicados nas equações de (3.6) a (3.8) obtemos:

$$\mathbf{f} = \begin{bmatrix} 0,1248 + 0,4163j \\ 0,0931 + 0,1589j \\ 0,3699 + 0,0283j \\ -0,1227 - 0,0939j \end{bmatrix} \text{ e } \mathbf{g} = \begin{bmatrix} 0,1248 - 0,4163j \\ 0,0931 - 0,1589j \\ 0,3699 - 0,0283j \\ -0,1227 + 0,0939j \end{bmatrix}.$$

Na Figura 3.11(d) podemos observar o desempenho do equalizador para um atraso  $k_0 = 2$ . É interessante notar que o nulo spectral não foi obstáculo à equalização do canal. Na Figura 3.12 está indicada a variância do erro na saída do equalizador LL em função do atraso  $k_0$ . Por este gráfico, podemos verificar que o desempenho ótimo do equalizador ocorre, de fato, quando o atraso escolhido é igual a 2.

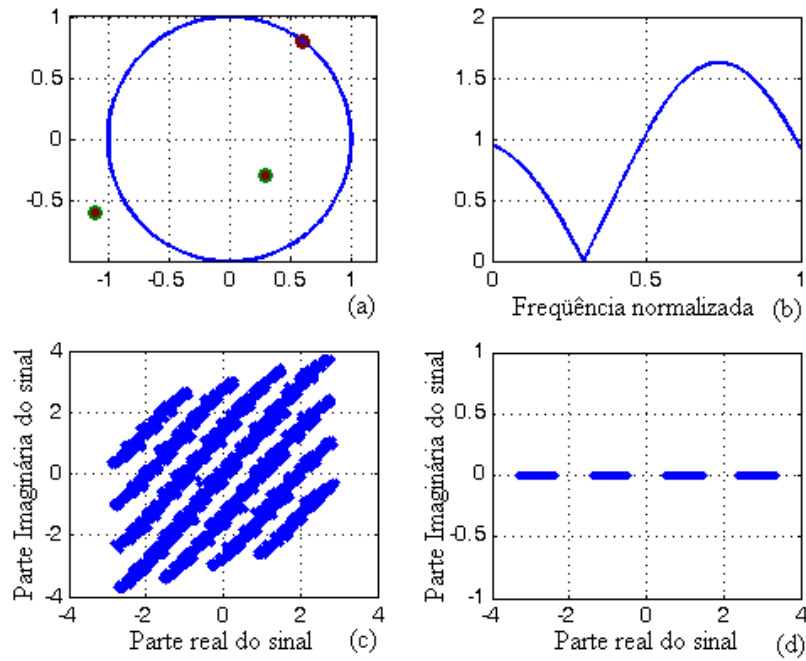


Figura 3.11. Canal simulado: (a) zeros, (b) resposta em frequência, (c) constelação após o canal, (d) constelação equalizada.

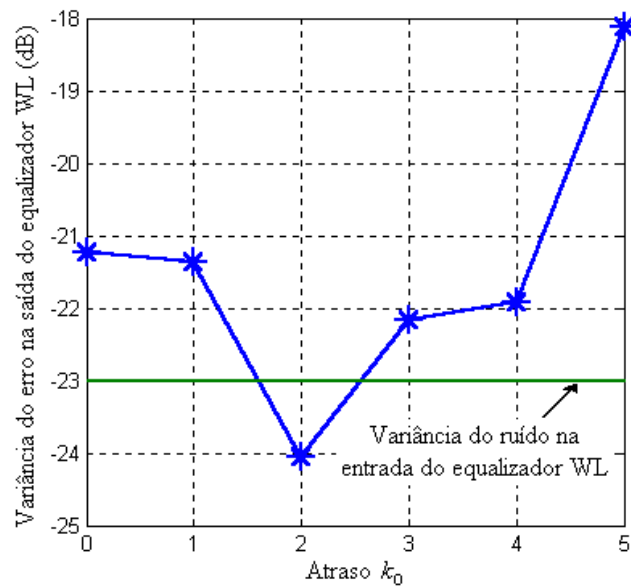


Figura 3.12: Variância do erro em função de  $k_0$ .

### 3.10.2 Equalização adaptativa (LMS-LL)

*Exemplo 1: equalizador adaptativo.* Novamente, iremos considerar um sinal 4-PAM sendo transmitido por um canal complexo sem nulos espectrais e de fase não-mínima cuja resposta ao impulso é dada por  $\mathbf{h}_c = [0,3054 + 0,0916j; -0,3726 - 0,6445j; 0,1225 + 0,3131j; 0,3659 - 0,3117j]$ . Usaremos o algoritmo LMS-LL com o passo de adaptação fixo  $\mu_{LL}$  igual a 0,0070

para ajustar os coeficientes do equalizador LL. O equalizador LL tem 16 coeficientes complexos, sendo oito no filtro  $\mathbf{f}$  e oito no filtro  $\mathbf{g}$ , com um passo de adaptação  $\mu$  igual a 0,0045 e um atraso  $k_0 = 6$  amostras. Para efeitos de comparação, também simulamos um equalizador linear (algoritmo LMS) com 32 coeficientes e um passo de adaptação  $\mu$  igual a 0,0060 e um atraso  $k_0 = 7$  amostras. Esse passo foi escolhido para que a convergência do equalizador linear fosse alcançada com até 3000 iterações. Os coeficientes dos dois equalizadores foram inicializados com zeros.

A Figura 3.13 mostra as curvas de aprendizado para estes dois equalizadores para várias SNR's (25, 30, 35 e 40 dB). Mediamos 500 realizações independentes para traçar cada curva do gráfico. No cálculo do erro (diferença entre o símbolo de treinamento e a saída do equalizador) levamos em conta apenas a parte real, já que a parte imaginária na saída do equalizador LL é nula e no equalizador linear ela não tem influência para a decisão do símbolo. Na Figura 3.13 também incluímos o valor teórico do  $J_{min}$  para os dois algoritmos.

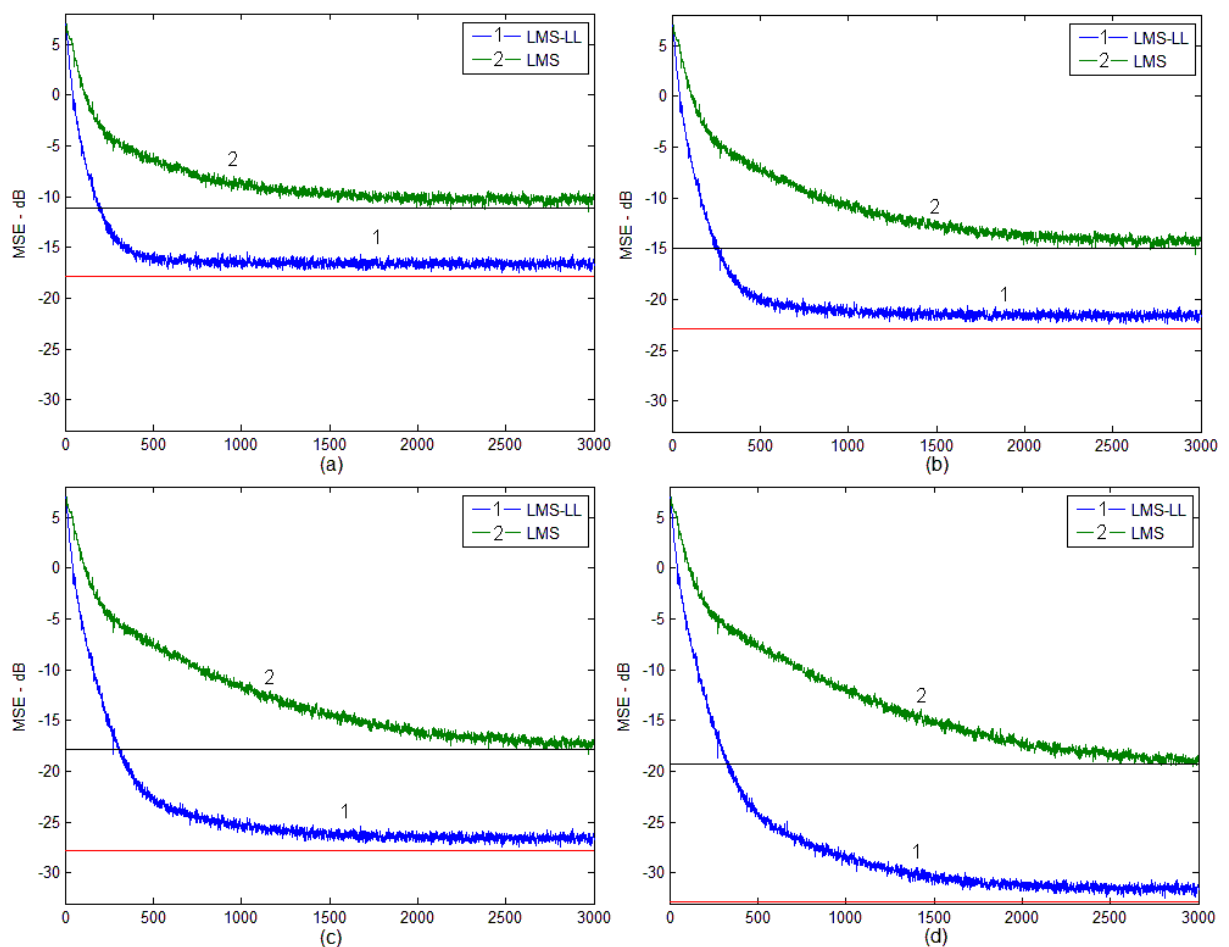


Figura 3.13: Comparação das curvas de aprendizado dos algoritmos LMS e LMS-LL. (a) SNR de 25 dB, (b) SNR de 30 dB, (c) SNR de 35 dB, (d) SNR de 40 dB.

Nesta aplicação, o desempenho do equalizador LL é claramente superior ao equalizador linear, especialmente quando a relação sinal-ruído é alta. De fato, se o canal apresentasse algum zero muito próximo ao CRU, o ganho em dB do equalizador LL em relação ao linear poderia ser muito maior que o ganho apresentado por este exemplo.

Finalmente, como o número de coeficientes do equalizador LL é menor que o do equalizador linear, a complexidade computacional também é menor, neste exemplo por aproximadamente um fator 2. Outro detalhe é que, para este canal, o aumento no número de coeficientes do equalizador LL pouco afetaria o seu desempenho, como mostramos no Apêndice B.

*Exemplo 2: equalizador adaptativo e constelação complexa.* Desta vez iremos considerar um sinal pertencente a uma constelação 16-QAM sendo transmitido por um canal cuja resposta ao impulso é dada por  $\mathbf{h} = [0,603; 0,3194-0,181j; 0,131+0,1368j; -0,3484-0,271j; 0,517+0,0615j]$ . Este canal é de fase mista e apresenta um “quase” nulo espectral, como podemos observar pela Figura 3.14. Consideramos uma relação sinal-ruído de 30 dB. Usaremos o algoritmo LMS-LL com o passo de adaptação fixo  $\mu$  igual a 0,02 para ajustar os coeficientes do equalizador LL. O equalizador LL tem sete coeficientes complexos no filtro  $\mathbf{f}$  e outros sete no filtro  $\mathbf{g}$ . Escolhemos um atraso  $k_0 = 3$ . Nesta simulação, estamos levando em conta os conceitos apresentados na Seção 2.4., incluindo a compensação de fase no receptor. Na Figura 3.15, observamos a constelação antes de passar pelo canal, após o canal e o sinal equalizado. A Figura 3.16 mostra a curva de aprendizado do algoritmo, após a mediação de 200 realizações independentes.

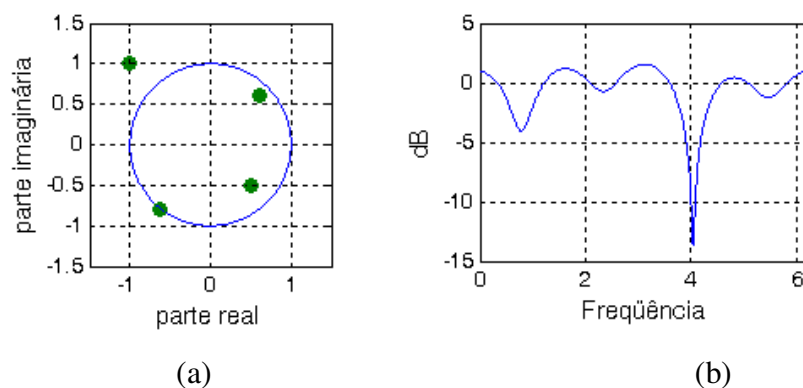


Figura 3.14. Canal do Exemplo 2: (a) zeros, (b) resposta em frequência.



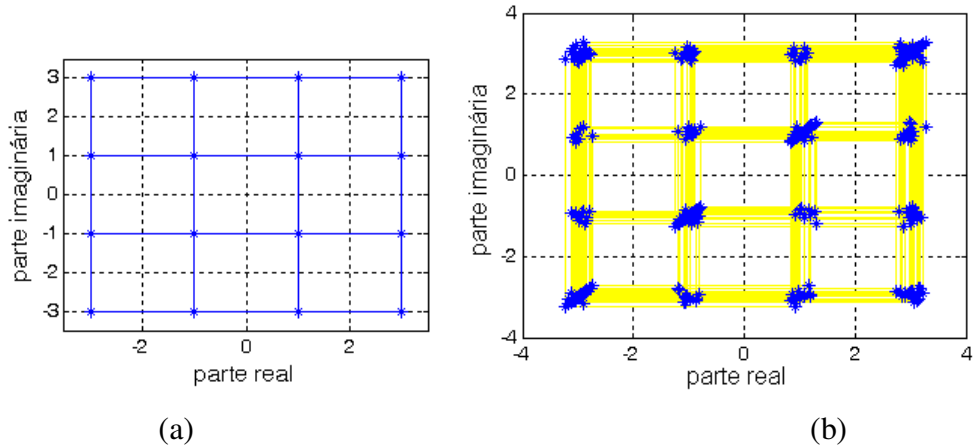


Figura 3.15: Constelação 16-QAM mostrando as transições (a) antes do canal, (b) equalizada. Relação sinal-ruído de 30 dB.

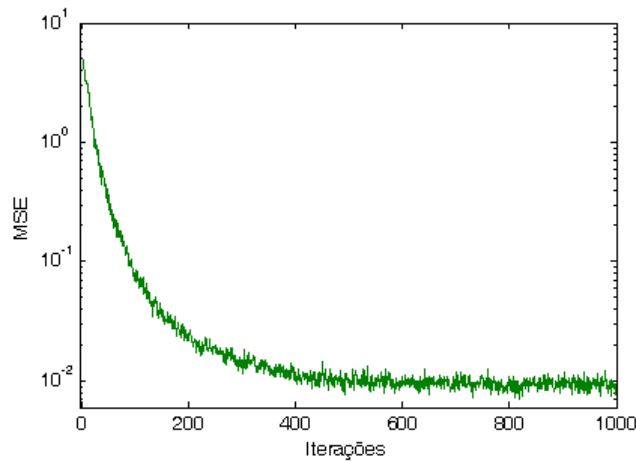


Figura 3.16: Curva de aprendizado do Exemplo 2.

### 3.10.3 Análise estatística: exemplos

Apresentaremos agora dois exemplos numéricos que comprovam a precisão do modelo estatístico desenvolvido na Seção 3.6. Em cada um destes dois exemplos, usaremos dois passos de adaptação, um “médio” e outro “pequeno” em relação à  $\mu_{\max}$ .

*Exemplo 1.* Consideramos novamente um sinal  $a[k]$  pertencente a uma constelação 4-PAM, uma relação sinal-ruído de 30 dB e um canal cuja resposta ao impulso é dada por  $\mathbf{h} = [0,495+0,155j; 0,477+0,040j; 0,277-0,411j; -0,346+0,044j; 0,324+0,174j]$ . Neste exemplo, estamos considerando que os filtros  $\mathbf{f}$  e  $\mathbf{g}$  têm seis coeficientes cada. A seqüência de treinamento é  $d[k] = a[k-k_0]$ , sendo  $k_0$  um atraso igual a 4 amostras. A Figura 3.17 apresenta

as curvas teóricas e simuladas (média de 500 realizações). Notamos que o comportamento previsto e o experimental apresentam uma excelente concordância. Já a Figura 3.18 mostra a evolução dos valores dos coeficientes para o filtro  $\mathbf{f}$  e os valores preditos. Podemos observar que existe uma boa concordância entre o modelo teórico e a simulação.

*Exemplo 2.* Neste exemplo, estamos considerando o canal cuja resposta ao impulso é dada por  $\mathbf{h} = [0,6394 - 0,1598j; 0,4715 + 0,1199j; 0,0519 - 0,0775j; -0,3257 + 0,3372j; 0,2587 - 0,1580j; -0,0527 + 0,0760j]$  e que os filtros  $\mathbf{f}$  e  $\mathbf{g}$  têm nove coeficientes cada. A sequência de treinamento é  $d[k] = a[k-k_0]$ , sendo  $k_0$  um atraso igual a 4 amostras. A Figura 3.19 apresenta as curvas teóricas e simuladas (média de 500 realizações). Novamente, consideramos uma relação sinal-ruído de 30 dB. Notamos que o comportamento previsto e o experimental apresentam uma excelente concordância. Já a Figura 3.20, mostra a evolução dos valores dos coeficientes para o filtro  $\mathbf{f}$  e os valores preditos. Novamente, temos uma boa concordância entre o modelo teórico e a simulação.

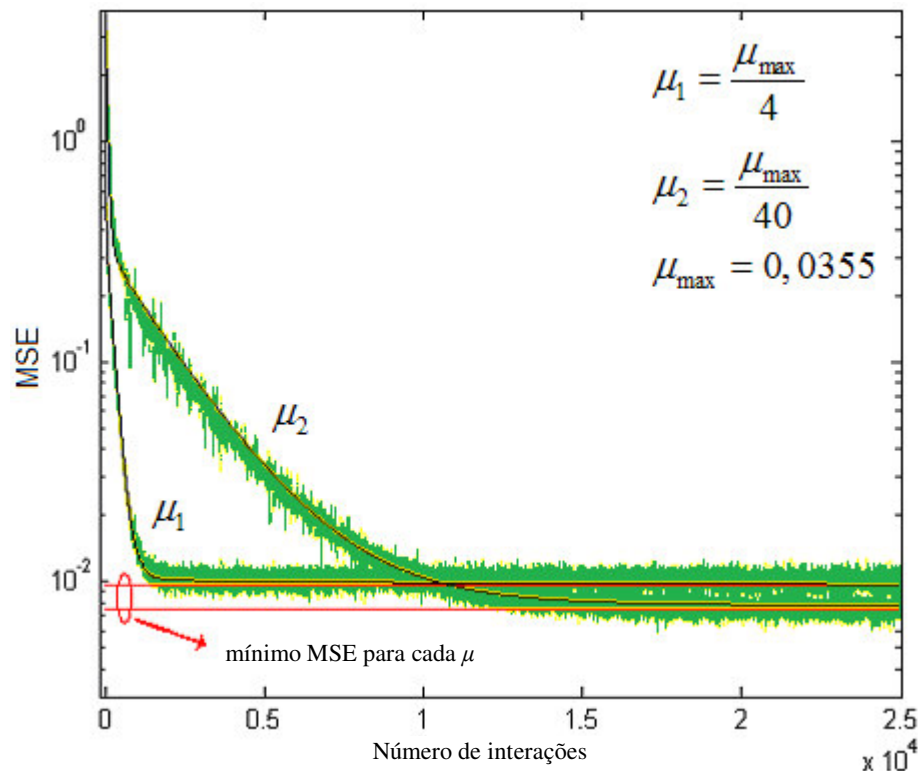


Figura 3.17: Comportamento do MSE, Exemplo 1.

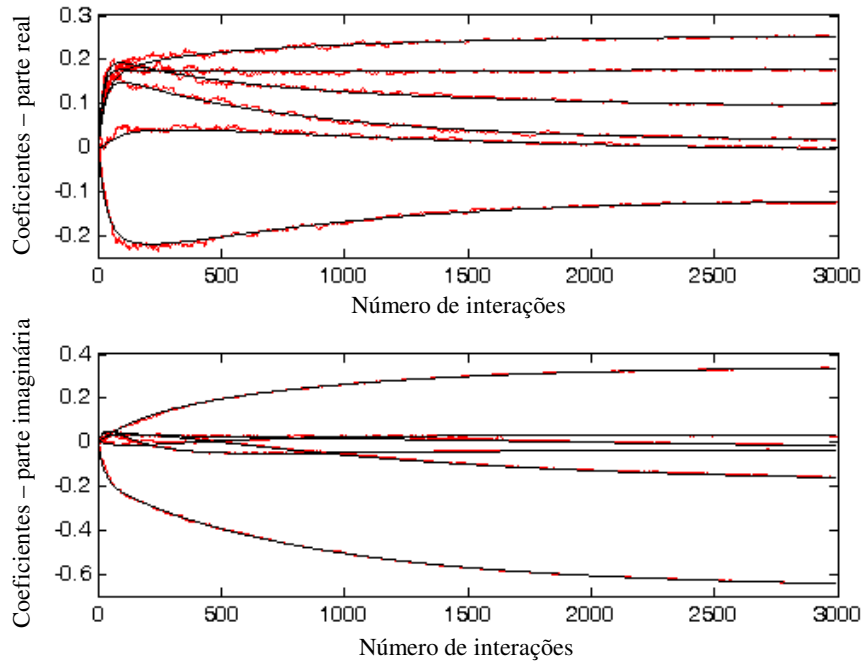


Figura 3.18 Comportamento teórico e simulado (média de apenas 3 realizações) para os coeficientes do filtro  $\mathbf{f}$ , Exemplo 1.

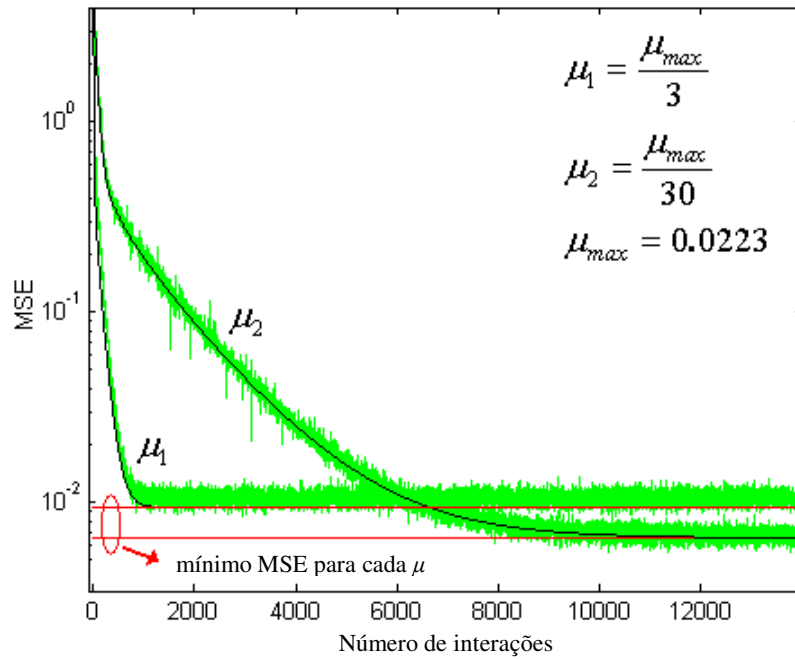


Figura 3.19 Comportamento do MSE, Exemplo 2.

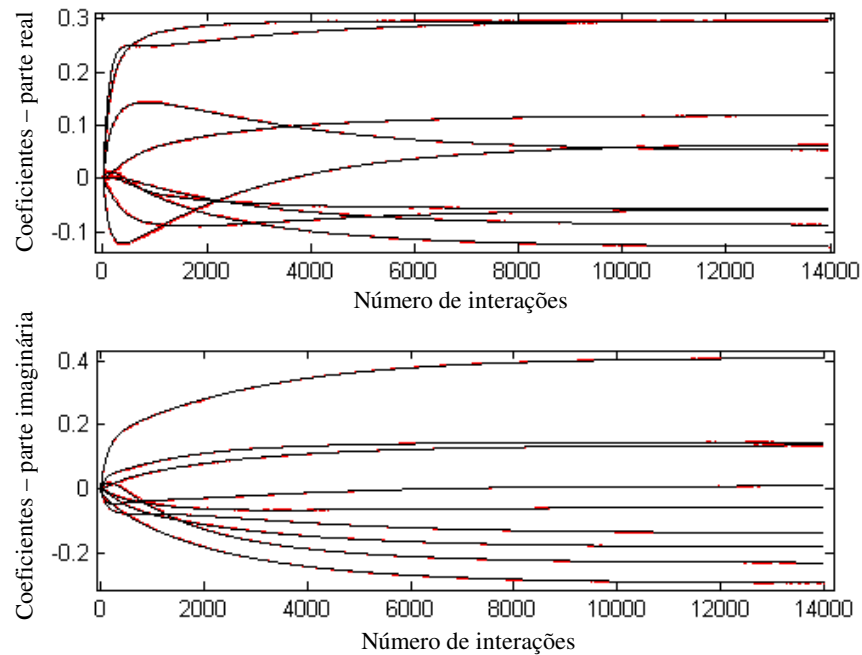


Figura 3.20 Comportamento teórico e simulado (média de apenas 3 realizações) para os coeficientes do filtro  $\mathbf{f}$ , Exemplo 2.

### 3.10.4 Algoritmo LMS-LL-MS: exemplos

Para as simulações de Monte Carlo a seguir, iremos considerar que o sinal transmitido pertence a uma constelação 4-PAM, que o canal é complexo, discreto, de ordem  $L$ , e que a relação sinal-ruído é de 30 dB. Todos os equalizadores considerados têm  $P = 8$  coeficientes. Em todas as simulações consideramos  $\varepsilon = 10^{-4}$  para o algoritmo LMS-LL-N e  $\gamma = 0,995$  para o algoritmo LMS-LL-MS. Escolhemos os passos de adaptação para que o erro quadrático médio  $\left( E\{|e_{LL}[k]|^2\}, E\{|e_{LL-N}[k]|^2\} \text{ e } E\{|e_{LL-MS}[k]|^2\} \right)$  após a convergência dos algoritmos sejam aproximadamente os mesmos. Todas as curvas de aprendizado foram obtidas pela mediação de 400 realizações independentes.

*Exemplo 1.* Nesta primeira simulação, consideramos um canal cuja resposta ao impulso é  $\mathbf{h}_1 = [0,601; 0,6309-0,2403j; 0,355+0,024j; -0,0348-0,047j; 0,156-0,1706j]$ . Este canal é de fase mista e apresenta um zero próximo ao círculo de raio unitário. A Figura 3.21 mostra os zeros deste canal, bem como a sua resposta em frequência, o sinal após passar pelo canal e o sinal equalizado. A Figura 3.22 apresenta as curvas de aprendizado para os algoritmos LMS-LL,

LMS-LL-N (normalizado) e LMS-LL-MS. Nesta simulação, escolhemos um atraso  $k_0 = 3$ . Notamos que os três algoritmos têm praticamente o mesmo erro quadrático médio final, mas o algoritmo LMS-LL-MS converge com um número menor de iterações. Os passos de adaptação usados foram:  $\mu_{LL} = 0,0045$ ;  $\mu_N = 0,22$  e  $\mu_{LL-MS} = 1/32$ .

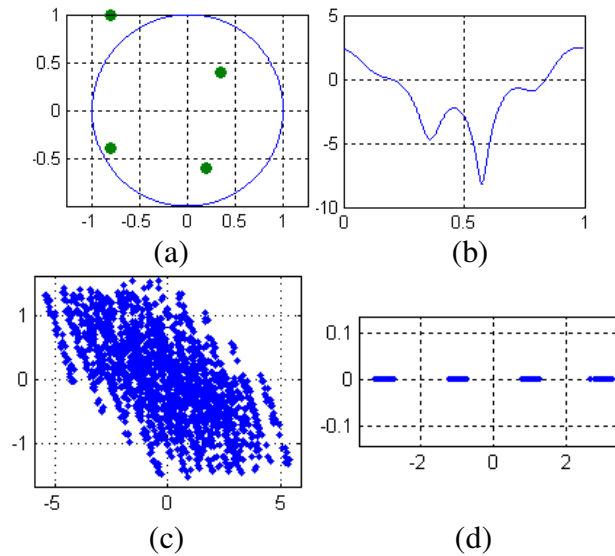


Figura 3.21 Canal  $\mathbf{h}_1$  (a) zeros, (b) resposta em frequência, (c) saída do canal, (d) constelação equalizada (igual para os três equalizadores).

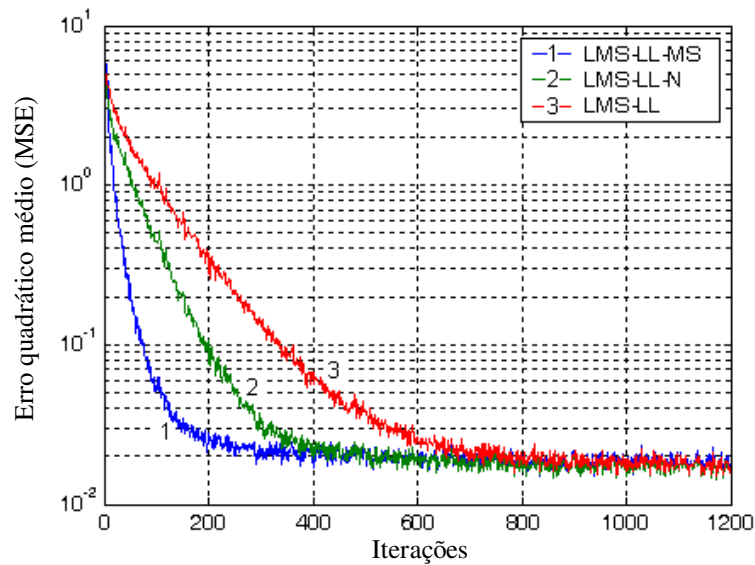


Figura 3.22 Curvas de aprendizado para o canal  $\mathbf{h}_1$ .

*Exemplo 2.* Nesta segunda simulação, consideramos um canal cuja resposta ao impulso é dada por  $\mathbf{h}_2 = [0,3921+0,3921j; 0,0392+0,745j; 0,051+0,2548j; 0,132+0,136j; 0,1068+0,1129j; 0,0423+0,0819j]$ . Este canal também apresenta fase mista e um zero próximo ao círculo de raio unitário. A Figura 3.23 mostra os zeros, a resposta em frequência e o sinal após passar

pelo canal. A Figura 3.24 apresenta as curvas de aprendizado para os algoritmos LMS-LL, LMS-LL-N e LMS-LL-MS. O atraso usado foi  $k_0 = 2$ . Novamente, observamos que os três algoritmos têm praticamente o mesmo erro quadrático médio após a convergência, mas o algoritmo LMS-LL-MS converge com um número menor de iterações. Os passos de adaptação usados foram:  $\mu_{LL} = 0,0050$ ;  $\mu_N = 0,25$  e  $\mu_{LL-MS} = 1/32$ .

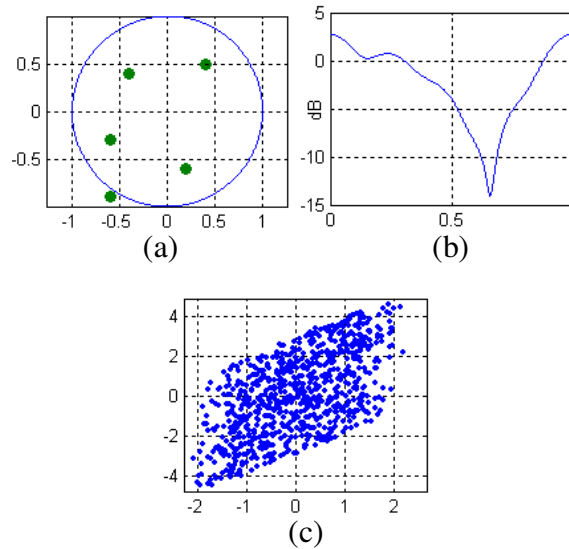


Figura 2.23 Canal  $\mathbf{h}_2$ : (a) zeros, (b) resposta em frequência, (c) sinal após o canal.

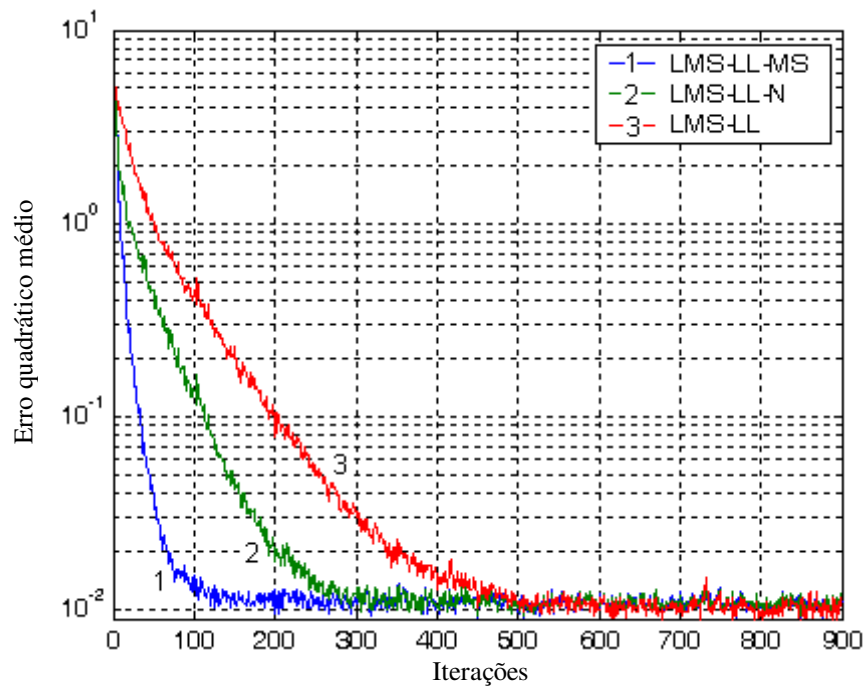


Figura 3.24 Curvas de aprendizado para o canal  $\mathbf{h}_2$ .

### 3.10.5 Implementação em FPGA: comparações e exemplos

*Comparação das Estruturas.* Para a avaliação das estruturas apresentadas, usamos um canal de fase não-mínima com dois zeros próximos ao círculo de raio unitário descrito pela resposta ao impulso  $\mathbf{h} = [0,6791; -0,1698; 0,1698 - 0,4176j; -0,0302 + 0,5531j]$ , como mostra a Figura 3.25. Nesta simulação, a constelação utilizada é uma 4-PAM, consideramos uma relação sinal-ruído de 30 dB e o passo de adaptação  $\mu_{LL} = 0,001$ . As curvas de aprendizado, mostradas na Figura. 3.25, foram obtidas com a mediação de 1000 realizações independentes. Esta figura também mostra o resultado da equalização deste canal de teste por um equalizador LL de “precisão infinita” simulado em Matlab®. Os resultados obtidos mostram que o equalizador com a estrutura paralela tem um desempenho muito próximo ao de precisão infinita. Os equalizadores com estrutura seqüencial e semi-paralela apresentam um desajuste aceitável, porém maior que o desajuste da estrutura paralela. Além disso, podemos observar que, para este canal, a taxa de convergência da estrutura paralela é maior, devido ao menor número de arredondamentos e aproximações realizados.

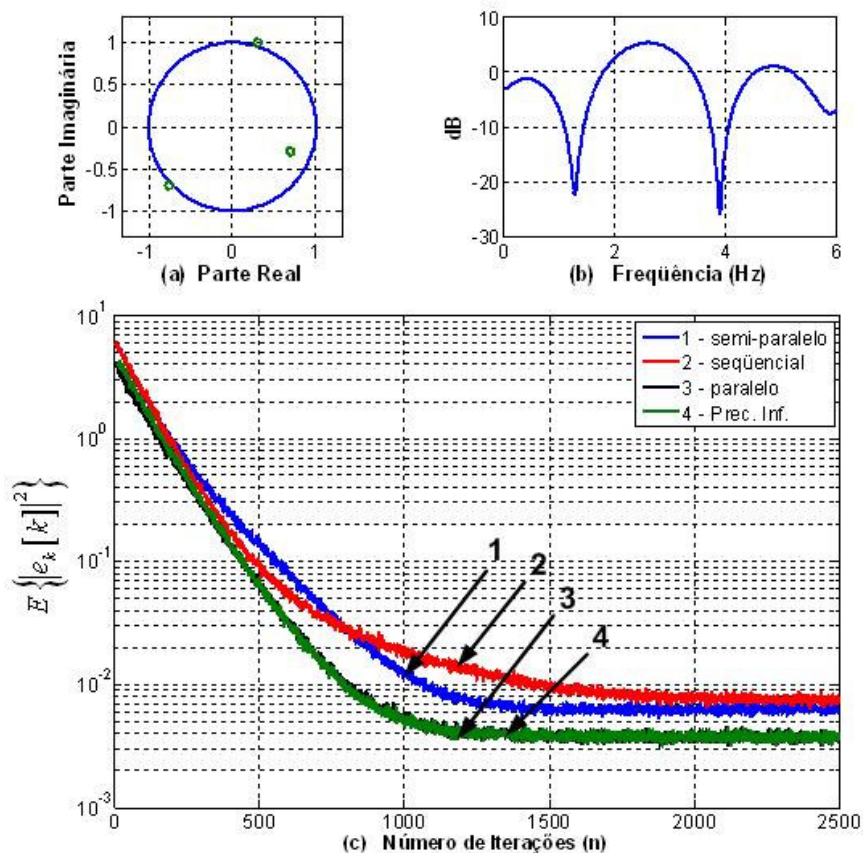


Figura 3.25 Comparação das estruturas paralela, semi-paralela e seqüencial: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado.

*Exemplo 1.* O canal A apresenta uma resposta ao impulso  $\mathbf{h} = [-0,1216-1,9799j; -0,1950+0,3102j; 0,2659-0,2216j; -0,1772+0,2393j]$ . Este canal é de fase mínima e seus zeros estão longe do círculo de raio unitário. A Figura 3.26 (a) e (b) mostra os zeros do canal e a sua resposta em frequência. A Figura 3.26 (c) apresenta as curvas de convergências do erro quadrático médio (MSE) para os algoritmos LMS e LMS largamente linear. A partir desta figura, pode-se observar que o algoritmo LMS largamente linear apresenta um menor MSE e uma melhor taxa de convergência.

*Exemplo 2.* O canal B tem a seguinte resposta ao impulso:  $\mathbf{h} = [-0,1375+0,2474j; 0,176-0,264j; 0,2749+0,33j; -0,22-0,385j; -0,44+0,4948j]$ . Este canal é de fase não-mínima e tem um zero próximo ao círculo de raio unitário, apresentando também uma forte distorção, como é mostrado pela Figura 3.27. As curvas de aprendizado mostram que o algoritmo LMS largamente linear apresenta um menor tempo para a convergência e um menor MSE. Na Tabela 3.5 temos uma comparação entre os algoritmos LMS e LMS-LL para os canais A e B.

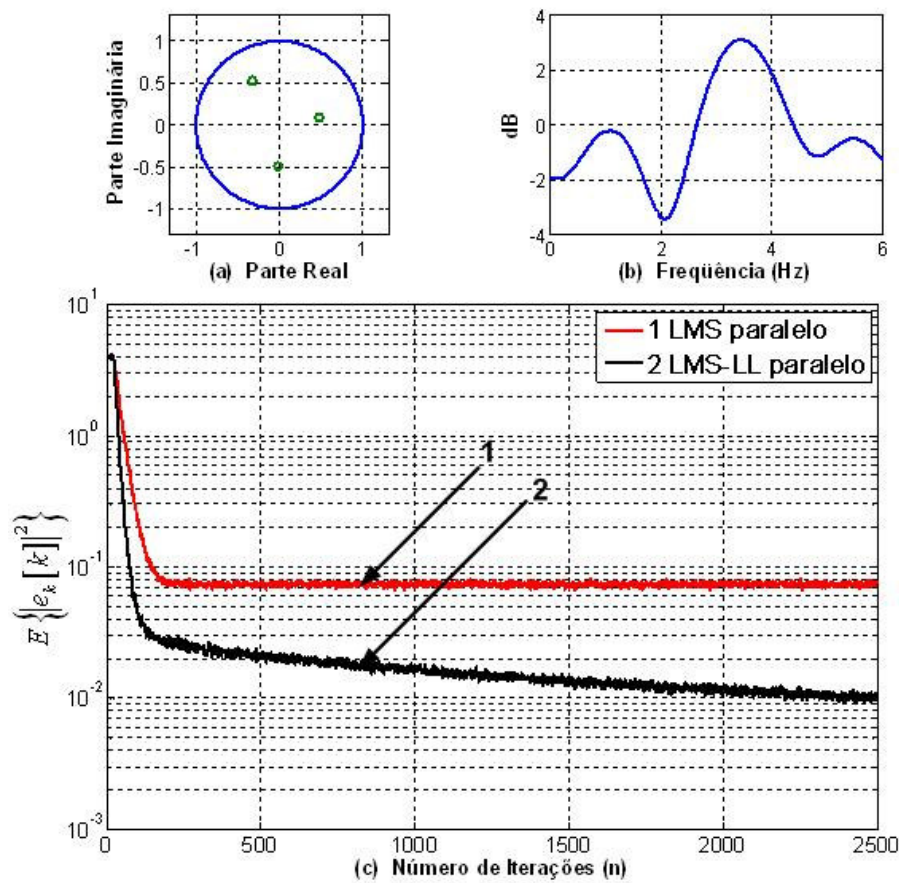


Figura 3.26 Simulação I: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado dos algoritmos para o canal A.



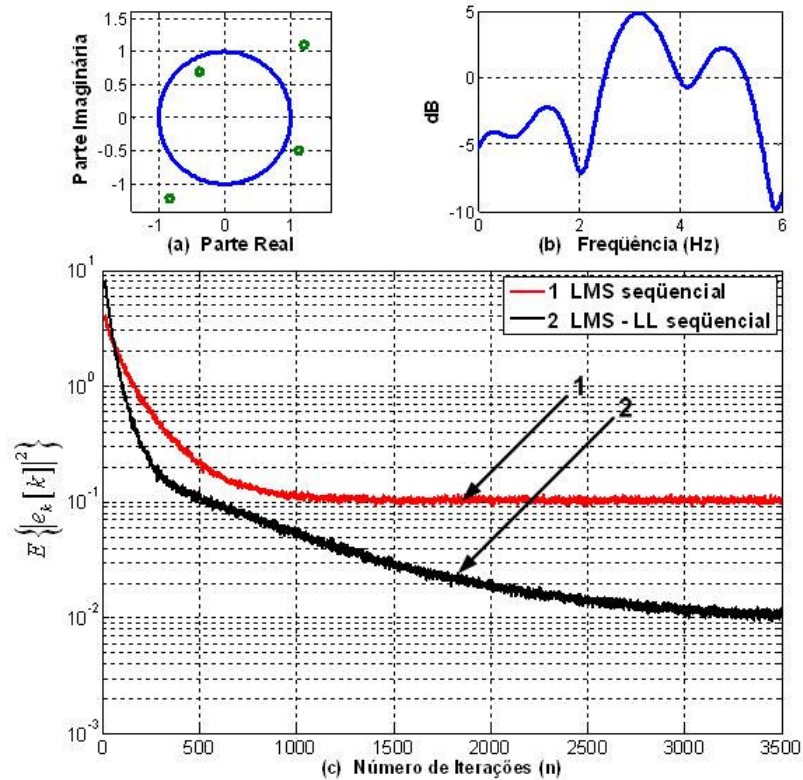


Figura 3.27 Simulação II: (a) raízes do canal, (b) resposta em frequência do canal, (c) curvas de aprendizado dos algoritmos para o canal B.

TABELA 3.5: COMPARAÇÃO DOS EQUALIZADORES

Canal	LMS	LMS-LL
A	Número de coef.: 4 Estrutura: paralela passo: 0,004	Número de coef.: 4 Estrutura: paralela passo: 0,004
B	Número de coef.: 56 Estrutura: seqüencial passo: 0,002	Número de coef.: 11 Estrutura: seqüencial passo: 0,002

### 3.11 Conclusão do capítulo

Nesse capítulo revisamos brevemente a necessidade de equalização de canais de comunicação. Concentramos a nossa atenção nos equalizadores lineares e largamente lineares. Em seguida, encontramos expressões para o cálculo do equalizador LL ótimo, pelos critérios MMSE e ZF. Na seqüência, mostramos um algoritmo LMS LL de passo fixo e realizamos a sua análise estatística. Aplicamos a transformada *multi-split* ao algoritmo LMS LL e

mostramos que é possível conseguir um ganho de convergência em relação ao LMS LL normalizado. Finalmente, implementamos em FPGA o equalizador LL de passo fixo.

No próximo capítulo abordaremos uma técnica de equalização cega baseada na predição largamente linear. Esta nova técnica apresenta várias vantagens quando comparada à técnica “elementar” de equalização cega por filtro de erro de predição (estritamente) linear, podendo ter, inclusive, menor complexidade computacional.

# Capítulo 4

---

## Filtro de Erro de Predição Largamente Linear

No capítulo anterior, focamos principalmente o processamento LL com o objetivo de equalizar canais de comunicação tendo uma seqüência de treinamento disponível. Em algumas situações, entretanto, o envio desta seqüência de treinamento não é possível ou se torna muito custoso em termos da banda desperdiçada devido à freqüente retransmissão de dados conhecidos. Neste ponto, técnicas de equalização autodidata ou cega podem ser muito interessantes, pois não utilizam seqüências de treinamento para equalizar o canal.

Neste capítulo, aplicaremos filtros de erro de predição largamente lineares ao problema de equalização cega<sup>11</sup>. Mostraremos que o filtro de erro de predição largamente linear ótimo, ao contrário da sua versão estritamente linear, pode compensar canais de fase não-mínima. E, de forma semelhante aos equalizadores LL, o número de coeficientes do filtro de erro de predição LL precisa ser apenas igual à ordem do canal para a sua inversão, a menos de uma constante multiplicativa, quando não há ruído aditivo. Em outras palavras, podemos ter uma grande redução na complexidade computacional usando esses novos equalizadores quando comparados aos seus equivalentes lineares. Esses resultados estão entre as nossas principais contribuições. Apresentamos a maior parte desses resultados nos congressos ICC'07 (Aquino *et al.*, 2007a), SBrT'07 (Aquino *et al.*, 2007c) e I2TS 2007 (Aquino *et al.*, 2007e).

Organizamos este capítulo como segue. A Seção 4.1 apresenta os fundamentos da predição linear e do filtro de erro de predição, incluindo as suas principais características e propriedades. A Seção 4.2 aborda o filtro de erro de predição largamente linear, destacando a sua característica de poder equalizar canais de fase não-mínima. Ainda nessa seção, calculamos os coeficientes ótimos segundo os critérios MMSE e ZF do equalizador. Na seção seguinte realizamos uma análise polifásica desse equalizador. Um filtro de erro de predição LL (FEPLL) adaptativo usando o algoritmo LMS de passo fixo é apresentado na Seção 4.4, e na Seção 4.5 apresentamos um algoritmo LMS de passo variável e o algoritmo RLS para a

---

<sup>11</sup> Podemos dizer que a expressão “equalização cega” (*blind equalization*) foi oficialmente introduzida por Benveniste e Goursat (1982), em um artigo com esse mesmo título. A expressão “*blind deconvolution*” remonta a um artigo de Stockham *et al.* (1975). Alguns autores usam o termo “equalização” para aplicações em tempo real e o termo “deconvolução” para processamento *off-line* (Belline, 1994).

atualização dos coeficientes do FEPLL. Desenvolvemos um FEPLL com atraso de equalização na Seção 4.6. Já na Seção 4.7, implementamos um FEPLL em FPGA. Na Seção 4.8 apresentamos vários resultados de simulação demonstrando o desempenho do equalizador e exemplificando os principais resultados teóricos obtidos. A Seção 4.9 conclui este capítulo.

## 4.1 Predição linear e filtro de erro de predição

O problema de predição linear consiste em prever o valor futuro de uma seqüência  $X_k$  estacionária aleatória discreta no tempo, dado um conjunto de amostras passadas desta mesma seqüência (Haykin, 1990). Portanto, dados os valores  $x[k]$ ,  $x[k-1]$ , ...,  $x[k-M+1]$  indicando as  $M$  amostras passadas da seqüência até e incluindo o instante  $k$ , o problema de predição de passo  $n$  consiste em calcular o valor  $\hat{x}[k+n|X_{k,k-M+1}]$  no instante de tempo  $k+n$ ,  $n \geq 1$ . Podemos expressar o valor predito como uma função das amostras passadas por:

$$\hat{x}[k+n|X_{k,k-M+1}] = F(x[k], \dots, x[k-M+1]). \quad (4.1)$$

A predição é linear se a função  $F(\cdot)$  pode ser expressa como uma combinação linear das amostras  $x[k]$ ,  $x[k-1]$ , ...,  $x[k-M+1]$ . Quando a operação de predição linear envolve as amostras  $x[k-1]$ ,  $x[k-2]$ , ...,  $x[k-M]$  para fazer uma estimativa de  $x[k]$ , dizemos que esta forma é uma predição linear direta de um passo (*forward one-step linear prediction*). Portanto, um preditor<sup>12</sup> linear de um passo pode ser descrito como um filtro FIR de ordem  $M-1$  pela expressão

$$\hat{x}[k] = \sum_{n=1}^M p_n^*[k] x[k-n], \quad (4.2)$$

onde  $p_1^*[k]$ ,  $p_2^*[k]$ , ...,  $p_M^*[k]$  são os coeficientes do preditor no instante  $k$ . A diferença entre a amostra verdadeira  $x[k]$  e sua predição  $\hat{x}[k|X_{k-1,k-M}]$  é dada por

$$e[k] = x[k] - \hat{x}[k|X_{k-1,k-M}], \quad (4.3)$$

<sup>12</sup> O termo “preditor” (adaptação do termo em inglês *predictor*), apesar de ser de uso corrente (Rocha, 1996), não é (ainda) um vocábulo da língua portuguesa, segundo o dicionário Aurélio Eletrônico Século XXI.

e é conhecido como resíduo ou erro de predição direto (*forward prediction error*). O preditor ótimo, no sentido MMSE, é aquele que minimiza o valor médio quadrático de  $e[k]$ , isto é, os coeficientes  $p_n^*[k]$ ,  $n = 1, \dots, M$ , do preditor são ajustados tal que a função custo  $J_e = E\{|e[k]|^2\}$  seja minimizada. Esta minimização da potência do erro de predição, no instante  $k$ , leva ao vetor de coeficientes ótimos dado por:

$$\mathbf{P}_0 = \mathbf{R}^{-1}\mathbf{r}, \quad (4.4)$$

onde  $\mathbf{P}_0 = [p_1^* \ p_2^* \ \dots \ p_M^*]^T$ ,  $\mathbf{r} = [r(1) \ r(2) \ \dots \ r(M)]^T$  é o vetor de autocorrelação, com  $r(i) = E\{x[k]x^*[k-i]\}$  e  $\mathbf{R}$  é a matriz de autocorrelação de  $x[k]$ .

#### 4.1.1. Filtro de erro de predição

A Figura 4.1 apresenta um diagrama de blocos mostrando um preditor linear com estrutura FIR e a construção de um *filtro de predição linear de um passo na direção direta* ou *filtro de erro de predição* a partir do preditor. Para simplificar um pouco, usaremos a expressão *filtro de predição* para o filtro de erro de predição. Podemos notar que a resposta ao impulso do filtro de predição é maior que a do preditor por uma amostra e que o erro  $e[k]$  na saída do filtro de predição pode ser expresso por

$$e[k] = x[k] - \sum_{n=1}^M f_n^*[k]x[k-n] = x[k] - \mathbf{f}^H[k]\mathbf{x}[k-1], \quad (4.5)$$

onde  $\mathbf{f}[k] = [f_1[k] \ f_2[k] \ \dots \ f_M[k]]^T$ ,  $\mathbf{x}[k] = [x[k] \ x[k-1] \ \dots \ x[k-M+1]]^T$  e os coeficientes  $f_n^*[k]$  são definidos por:

$$f_n^*[k] = \begin{cases} -p_n^*[k], & n = 1, \dots, M \\ 0, & n > M. \end{cases} \quad (4.6)$$

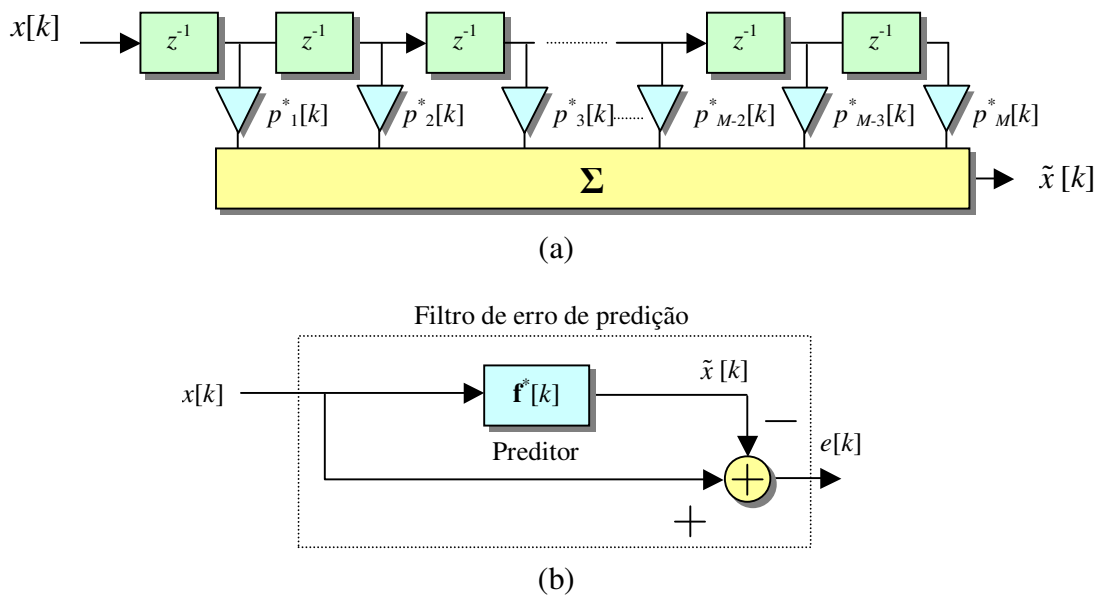


Figura 4.1: (a) Preditor linear e (b) filtro de erro de predição linear (FEPL).

### 4.1.2. Propriedades do filtro de erro de predição

Considere um filtro de predição direto de ordem  $M$ , com os seus coeficientes indicados por  $f_1^*[k]$ ,  $f_2^*[k]$ , ...,  $f_M^*[k]$ . Esta seqüência define a resposta ao impulso do filtro. A função de transferência  $F(z)$  deste filtro, para um dado instante  $k$ , é definida como a transformada  $z$  de sua resposta ao impulso:

$$F(z) = 1 - \sum_{n=1}^M f_n^* z^{-n}. \quad (4.7)$$

A função  $F(z)$  opera sobre o sinal de entrada  $x[k]$  para produzir o erro de predição  $e[k]$ , como mostra a Figura 4.1(b). Esta função de transferência apresenta as seguintes características (Haykin, 1990):

- para qualquer entrada estacionária  $x[k]$ ,  $F(z)$  não apresenta zeros fora do CRU no plano  $z$ ;
- a função de transferência  $F(z)$  tem zeros sobre o CRU somente quando o sinal  $x[k]$  consiste de um processo puramente senoidal;
- para um processo estacionário cujo espectro de potência não seja puramente discreto, a função de transferência  $F(z)$  apresenta zeros somente dentro do CRU.

A partir das características da função de transferência  $F(z)$  do filtro de predição, podemos afirmar que este filtro apresenta a propriedade de ser de *fase mínima* (todos os zeros estão dentro do CRU). Duas das conseqüências desta característica são as propriedades de *atraso de grupo mínimo*<sup>13</sup> e *atraso mínimo*, isto é, a energia da resposta ao impulso está concentrada próximo ao instante de chegada do impulso. Vale a pena destacar que o coeficiente de maior amplitude (valor absoluto) da resposta ao impulso não precisa ocorrer no instante  $k = 0$ , mas apenas que o *centro de energia* esteja mais próximo do instante  $k = 0$  do que do instante  $k = M$ . O centro de energia é definido por (Haykin, 1990):

$$\bar{k} = \frac{\sum_{n=1}^M n |f_n^*|^2}{\sum_{n=1}^M |f_n^*|^2}. \quad (4.8)$$

Uma outra propriedade importante do filtro de predição é a de “*branqueamento*”. Teoricamente, um filtro de predição de ordem  $M$  pode branquear qualquer processo estacionário de entrada, representado pela seqüência de entrada  $x[k], x[k-1], \dots, x[k-M]$ , desde que a ordem do filtro,  $M$ , seja suficientemente elevada. Ao incrementarmos a ordem do filtro de predição, reduzimos a correlação entre amostras adjacentes do processo aplicado na entrada do filtro. Por isso, muitas vezes este filtro é chamada de filtro branqueador.

Essa propriedade é uma conseqüência da minimização de  $J_e = E\{[e[k]]^2\}$  e do princípio da ortogonalidade aplicado ao filtro de predição. Na prática, como o comprimento do filtro é finito, a característica de branqueamento não pode ser completamente satisfeita (Rocha, 1996), exceto para o caso de sinais autoregressivos.

### 4.1.3. Predição linear e modelamento autoregressivo

A análise de predição linear direta e o modelamento autoregressivo de um processo aleatório são profundamente entrelaçados. Um processo autoregressivo (AR) de ordem  $M$  pode ser definido no domínio temporal como:

$$x[k] = -a_1 x[k-1] - a_2 x[k-2] - \dots - a_M x[k-M] + v[k], \quad (4.9)$$

<sup>13</sup> O atraso de grupo é definido como o negativo da derivada da resposta de fase em relação à frequência angular:  $\tau_g = -d\theta(\omega)/d\omega$ .

onde  $v[k]$  tem variância  $\sigma_v^2$  e representa a inovação no processo e os coeficientes  $a_i$  são os parâmetros do processo autoregressivo  $x[k]$ . Logo, o valor corrente do processo é igual a uma combinação de seus valores passados mais a inovação. Um modelo AR pode ser representado por uma função de transferência  $G(z)$  expressa por:

$$G(z) = \frac{1}{1 + \sum_{i=1}^M a_i z^{-i}}. \quad (4.10)$$

Considere um processo AR estável. Para isto ocorrer todos os pólos da função de transferência  $G(z)$  devem estar no interior do CRU. Podemos definir um filtro inverso (filtro de branqueamento) como  $H(z)=1/G(z)$ . Como o processo AR é estável, então podemos afirmar que todos os zeros de  $H(z)$  também estão no interior do CRU. Isto mostra que a função de transferência de um filtro de predição corresponde à inversa da função de transferência de um processo AR estável. Portanto, a análise da predição linear de um sinal aleatório estacionário no sentido amplo (WSS) e o modelamento AR podem ser vistos como um par de operações complementares, como mostra a Figura 4.2.

Dessa forma, um filtro de predição pode prover uma técnica de equalização cega “elementar” baseada em estatísticas de segunda ordem quando o canal SISO é de fase mínima. Esta é, infelizmente, uma restrição séria já que tais canais são raros na prática (Papadias e Slock, 1999).

Podemos perceber que a saída do filtro de predição só será branca (descorrelacionada) se a ordem do filtro for igual ou maior que a do modelo AR. Por outro lado, se a sequência  $x[k]$  for gerada por um filtro FIR de ordem  $N$ , teremos um processo de média móvel (MA - *moving average*) de ordem  $N$ :

$$x[k] = v[k] + b_1 v[k-1] + b_2 v[k-2] + \dots + b_N v[k-N], \quad (4.11)$$

onde os coeficientes  $b_i$  são os parâmetros do processo de média móvel. Neste caso, o filtro de predição precisa, em geral, ter um comprimento infinito para poder descorrelacionar as amostras de entrada, considerando que a função de transferência  $B(z) = 1 + \sum_{i=1}^M b_i z^{-i}$  correspondente ao modelo MA tenha todos os seus zeros dentro do CRU.



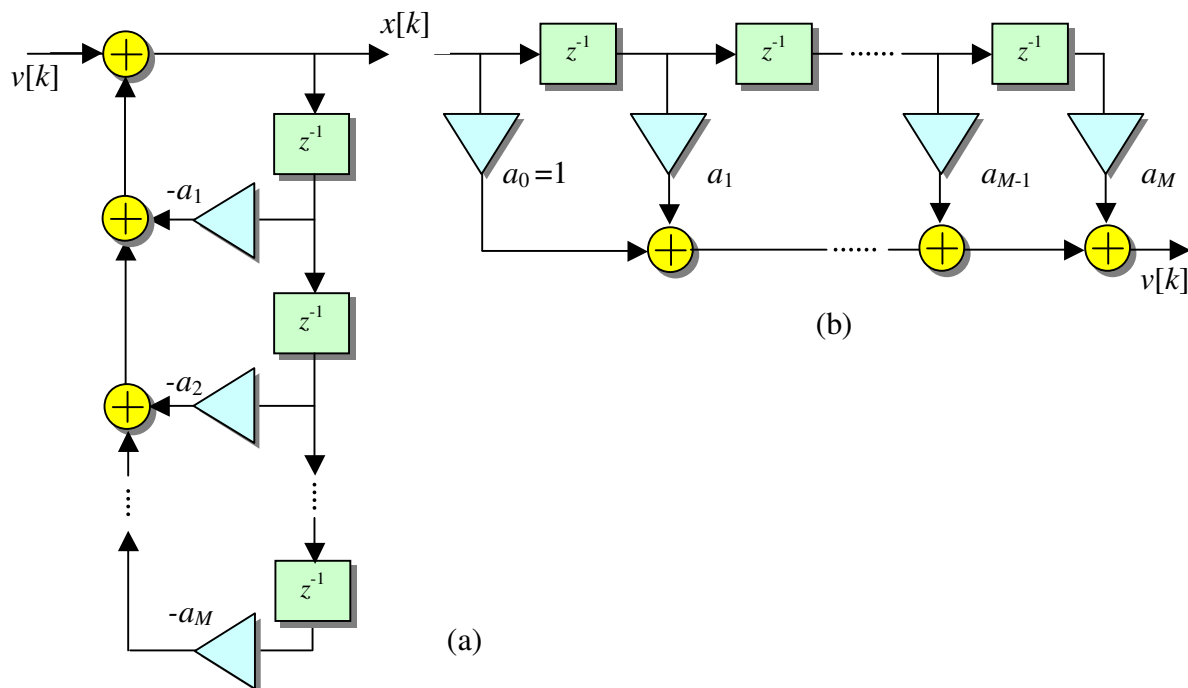


Figura 4.2: (a) Síntese de um processo AR com um filtro IIR, (b) análise de um processo AR usando um filtro de predição.

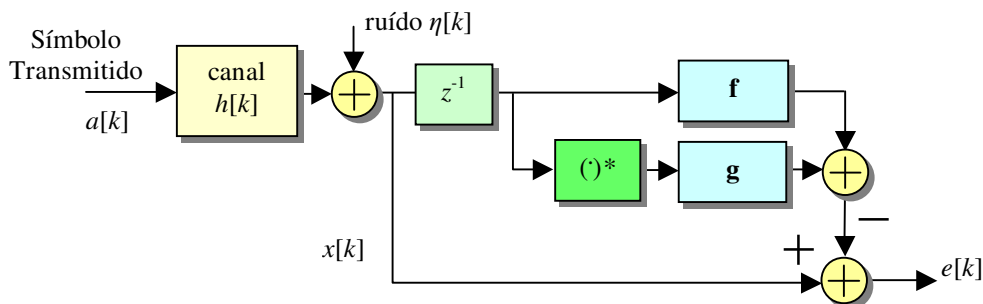


Figura 4.3: Filtro de erro de predição largamente linear (FEPLL).

## 4.2 Filtro de erro de predição largamente linear (FEPLL)

A Figura 4.3 mostra um filtro de erro de predição largamente linear (FEPLL). Este filtro, da mesma forma que o equalizador LL, apresenta a operação de conjugação complexa e os filtros  $f$  e  $g$  que atuam em paralelo. É interessante notar que a saída do FEPLL, da mesma forma que no caso do FEPL, é o próprio sinal de erro  $e[k]$ . O FEPLL e a sua análise é uma das nossas principais contribuições nesta tese.

O FEPLL pode, ao contrário do FEP estritamente linear, equalizar satisfatoriamente canais complexos de fase não-mínima ou com nulos espectrais, desde que os zeros do canal

não sejam zeros reais nem pares de zeros complexos conjugados próximos ao CRU, quando lidamos com sinais impróprios (Aquino *et al.*, 2007a).

O sinal de entrada  $a[k]$  deve ser descorrelacionado para que o FEPLL opere apropriadamente. Esta não é uma restrição importante, pois, em geral, nos sistemas de comunicação os símbolos transmitidos se distribuem de forma aproximadamente uniforme em um dado alfabeto. Além disso, em vários sistemas de comunicação são usados embaralhadores (*scramblers*) e desembaralhadores (*unscramblers*) que eliminam seqüências longas de bits iguais. Adicionalmente, lembramos que o sinal  $a[k]$  deve ser real, isto é, ele deve pertencer a uma constelação  $M$ -PAM ou do tipo OQAM (BPSK, MSK, GMSK ou OQAM, ver Capítulo 2, Seção 2.4). Em outras palavras: o sinal na saída do canal deve ser impróprio.

Podemos encontrar uma análise detalhada do FEPL, não adaptativo, em Haykin (1990). No contexto de equalização multicanal (FSE), o primeiro trabalho usando predição linear é apresentado por Slock (1994) e aprofundado por Abed-Meraim *et al.* (1997). Abed-Meraim *et al.* mostram que o FEPL em sistemas superamostrados é robusto ao problema de sobre-determinação da ordem do canal, o que não ocorre com outros métodos baseados em estatísticas de segunda ordem, tais como os métodos baseados em subespaço.

#### 4.2.1 Filtro de erro de predição LL ótimo – critério MSE

Agora, determinaremos os coeficientes ótimos para o filtro de predição largamente linear segundo o critério MSE. Para isso, precisaremos expressar o erro de predição largamente linear em função dos filtros  $\mathbf{f}$  e  $\mathbf{g}$ , do canal e do ruído. Neste cálculo, vamos considerar o canal fixo. Novamente, admitiremos que o canal complexo apresenta uma resposta ao impulso tipo FIR de comprimento  $L+1$ . O tamanho dos filtros  $\mathbf{f}$  e  $\mathbf{g}$  são  $M \geq L$ . Para um filtro de predição largamente linear (Figura 4.3), podemos expressar o erro de predição direto de um passo por:

$$\begin{aligned} e[k] &= x[k] - \sum_{n=1}^M f_n^*[k] x[k-n] - \sum_{n=1}^M g_n^*[k] x^*[k-n] \\ &= x[k] - \mathbf{f}^H[k] \mathbf{x}[k-1] - \mathbf{g}^H[k] \mathbf{x}^*[k-1], \end{aligned} \quad (4.12)$$

onde  $\mathbf{x}[k-1] = [x[k-1] \ x[k-2] \ \dots \ x[k-M]]^T$ ,  $x[k] = \mathbf{h}^T \mathbf{a}[k] + \eta[k]$ , com

$\mathbf{a}[k] = [a[k] \ a[k-1] \ \dots \ a[k-L]]^T$ ,  $\mathbf{h} = [h_0 \ h_1 \ \dots \ h_L]^T$ ,  $\mathbf{f}[k] = [f_1[k] \ f_2[k] \ \dots \ f_M[k]]^T$  e

$\mathbf{g}[k] = [g_1[k] \ g_2[k] \ \dots \ g_M[k]]^T$ . Omitindo a dependência temporal dos filtros  $\mathbf{f}$  e  $\mathbf{g}$ , podemos reescrever (4.12) como

$$e[k] = \mathbf{h}_e^T \mathbf{a}_e[k] - \mathbf{f}^H (\mathbf{H} \mathbf{a}_e[k] + \boldsymbol{\eta}[k-1]) - \mathbf{g}^H (\mathbf{H}^* \mathbf{a}_e[k] + \boldsymbol{\eta}^*[k-1]), \quad (4.13)$$

onde  $\boldsymbol{\eta}[k] = [\eta[k] \ \eta[k-1] \ \dots \ \eta[k-M]]^T$  é um vetor de ruído,  $\mathbf{h}_e = [h_0 \ \dots \ h_L \ 0 \ \dots \ 0]^T$  tem dimensão  $(M+L+1) \times 1$ ,  $\mathbf{a}_e = [a[k] \ a[k-1] \ \dots \ a[k-M-L]]^T$  tem a mesma dimensão de  $\mathbf{h}_e$ , e a matriz do canal de dimensão  $M \times (M+L+1)$  é definida por:

$$\mathbf{H} = \begin{bmatrix} 0 & h_0 & h_1 & \dots & h_L & 0 & \dots & 0 \\ 0 & 0 & h_0 & h_1 & \dots & h_L & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & h_0 & h_1 & \dots & h_L \end{bmatrix}. \quad (4.14)$$

A primeira coluna de zeros de  $\mathbf{H}$  em (4.14) corresponde ao atraso introduzido no filtro de erro de predição. Aplicando o critério MSE e as regras de derivação complexa (Haykin, 1995), podemos encontrar os filtros  $\mathbf{f}$  e  $\mathbf{g}$  ótimos pela resolução do seguinte sistema de equações:

$$\begin{bmatrix} \mathbf{H}\mathbf{H}^H + \xi\mathbf{I} & \mathbf{H}\mathbf{H}^T \\ \mathbf{H}^*\mathbf{H}^H & \mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{f}_{opt} \\ \mathbf{g}_{opt} \end{bmatrix} = \begin{bmatrix} \mathbf{H}\mathbf{h}_e^H \\ \mathbf{H}^*\mathbf{h}_e^H \end{bmatrix}, \quad (4.15)$$

onde  $\xi = \sigma_\eta^2 / \sigma_a^2$  e  $\mathbf{I}$  é uma matriz identidade de dimensão  $M \times M$ . Os vetores ótimos são dados por:

$$\mathbf{f}_{opt} = \left[ (\mathbf{H}\mathbf{H}^H + \xi\mathbf{I}) - \mathbf{H}\mathbf{H}^T (\mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I})^{-1} \mathbf{H}^*\mathbf{H}^H \right]^{-1} \left( \mathbf{H}\mathbf{h}_e^H - \mathbf{H}\mathbf{H}^T (\mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I})^{-1} \mathbf{H}^*\mathbf{h}_e^H \right), \quad (4.16)$$

$$\mathbf{g}_{opt} = \left[ \mathbf{H}\mathbf{H}^T - (\mathbf{H}\mathbf{H}^H + \xi\mathbf{I}) (\mathbf{H}^*\mathbf{H}^H)^{-1} (\mathbf{H}^*\mathbf{H}^T + \xi\mathbf{I}) \right]^{-1} \left( \mathbf{H}^*\mathbf{h}_e^H - (\mathbf{H}\mathbf{H}^H + \xi\mathbf{I}) (\mathbf{H}^*\mathbf{H}^H)^{-1} \mathbf{H}\mathbf{h}_e^H \right). \quad (4.17)$$

Podemos notar que os filtros ótimos *não* são complexos conjugados, como ocorre no caso do equalizador LL treinado. Isto ocorre porque na minimização do erro (ver em (4.13)), o sinal “desejado” ( $\mathbf{h}_e^T \mathbf{a}_e[k]$ ) sofre a influência do canal. O erro ótimo pode ser expresso por  $e_{opt}[k] = h_0 a[k] + \eta_{FES}[k]$ , onde  $h_0$  é primeiro coeficiente (possivelmente complexo) do canal e  $\eta_{FES}[k]$  representa o ruído filtrado pelo FEPLL mais a interferência residual entre símbolos.

#### 4.2.2 Critério ZF para o filtro de predição LL

O sistema de equações em (4.15) pode ser reescrito de forma mais conveniente quando  $\xi \rightarrow 0$  (canal sem ruído). Nesta situação, o filtro de predição LL segundo o critério MMSE se confunde com um filtro LL segundo o critério ZF, ou seja, o erro de predição LL se torna igual ao sinal transmitido a menos de uma constante multiplicativa ( $h_0$ ). Para que isso ocorra, o filtro de predição LL precisa ter apenas  $M = L$  coeficientes em cada um dos filtros  $\mathbf{f}$  e  $\mathbf{g}$ , como mostraremos a seguir. Note que para o FEPL isso não é verdade, mesmo quando o canal é de fase mínima e não apresenta qualquer zero próximo ao CRU, a não ser que o canal seja um sistema IIR.

A partir de (4.12) e desprezando o ruído, podemos expressar o erro de predição como

$$\begin{aligned}
 e[k] = & h_0 a[k] + h_1 a[k-1] + \dots + h_L a[k-L] - f_1^* (h_0 a[k-1] + h_1 a[k-2] + \dots + h_L a[k-L-1]) + \\
 & - f_2^* (h_0 a[k-2] + \dots + h_L a[k-L-2]) + \dots - f_L^* (h_0 a[k-L] + \dots + h_L a[k-2L]) + \\
 & - g_1^* (h_0^* a[k-1] + \dots + h_L^* a[k-L-1]) - g_2^* (h_0^* a[k-2] + \dots + h_L^* a[k-L-2]) + \dots \\
 & - g_L^* (h_0^* a[k-L] + h_1^* a[k-L-1] + \dots + h_L^* a[k-2L]).
 \end{aligned}
 \tag{4.18}$$

Aplicando o critério ZF ao filtro de predição LL, temos  $e[k] = h_0 a[k]$ , logo (4.18) irá resultar no seguinte sistema de equações lineares com  $2L (= 2M)$  variáveis:

$$\begin{bmatrix} h_0 & h_0^* & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ h_1 & h_1^* & h_0 & h_0^* & 0 & 0 & \dots & 0 & 0 \\ h_2 & h_2^* & h_1 & h_1^* & h_0 & h_0^* & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{L-1} & h_{L-1}^* & h_{L-2} & h_{L-2}^* & h_{L-3} & h_{L-3}^* & \dots & h_0 & h_0^* \\ h_L & h_L^* & h_{L-1} & h_{L-1}^* & h_{L-2} & h_{L-2}^* & \dots & h_1 & h_1^* \\ 0 & 0 & h_L & h_L^* & h_{L-1} & h_{L-1}^* & \dots & h_2 & h_2^* \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & h_L & h_L^* \end{bmatrix} \begin{bmatrix} f_1^* \\ g_1^* \\ f_2^* \\ g_2^* \\ \vdots \\ f_{L-1}^* \\ g_{L-1}^* \\ f_L^* \\ g_L^* \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ \vdots \\ h_L \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad (4.19)$$

Note que (4.19) é, essencialmente, uma reordenação de (4.15) na ausência de ruído. A partir do sistema de equações em (4.19) podemos fazer as seguintes considerações para a existência de uma solução ZF:

- o canal precisa ser complexo (ao menos um coeficiente precisa ser realmente complexo, isto é,  $\Re\{h_i\} \neq 0$  e  $\Im\{h_i\} \neq 0$ ) e os canais  $\mathbf{h}$  e  $\mathbf{h}^*$  não podem possuir zeros em comum (identidade de Bezout, (Papadias e Slock, 1997; Giannakis e Halford, 1997));
- os filtros ótimos  $\mathbf{f}_{opt}$  e  $\mathbf{g}_{opt}$  precisam ter apenas  $L$  coeficientes;
- os filtros ótimos  $\mathbf{f}_{opt}$  e  $\mathbf{g}_{opt}$  não são conjugados complexos, diferentemente do que ocorre com o equalizador LL, com isso não mais podemos simplificar os cálculos como fizemos para o equalizador LL (ver Capítulo 3).

Caso o filtro de erro de predição LL com  $\mathbf{f}_{opt}$  e  $\mathbf{g}_{opt}$  calculados a partir de (4.19) exista, o FEPLL torna-se um equalizador ZF (*zero forcing*), a menos de uma constante multiplicativa, isto é,  $e[k] = h_0 a[k]$ . A existência de uma solução ZF também indica que o problema de equalização é bem condicionado, no sentido de que pode ser garantido um bom desempenho do sistema se o nível de ruído se mantiver baixo, além de ser certa a existência de uma solução MSE (Tidestav *et al.*, 1999).

Portanto, canais de fase não-mínima, inclusive aqueles que possuem nulos espectrais podem ser equalizados pelo filtro de predição largamente linear. Como  $h_0$  pode ser um valor complexo, o filtro de predição LL precisa de uma compensação de ganho e de fase adicionais para recuperação do símbolo transmitido. Na Seção 4.8, apresentamos alguns resultados de simulação para ilustrar o desempenho desse equalizador.

### 4.3 Análise polifásica do filtro de erro de predição LL

Nesta seção, faremos uma análise do filtro de predição LL como um sistema polifásico. O objetivo é mostrar claramente que a solução MSE, quando o ruído é nulo ( $\xi = 0$ ), é equivalente à solução ZF. Esta análise é baseada em (Giannakis e Halford, 1997).

A operação de conjugação complexa equivale transformar um sistema SISO em um sistema SIMO com dois canais complexos conjugados. Portanto, da mesma forma que no caso do equalizador LL, podemos visualizar o filtro de predição LL como um sistema polifásico com duas fases, como mostrado na Figura 4.4. Mais uma vez, isto é possível porque o sinal transmitido pertence a uma constelação real ou pode ser interpretado como um sinal real.

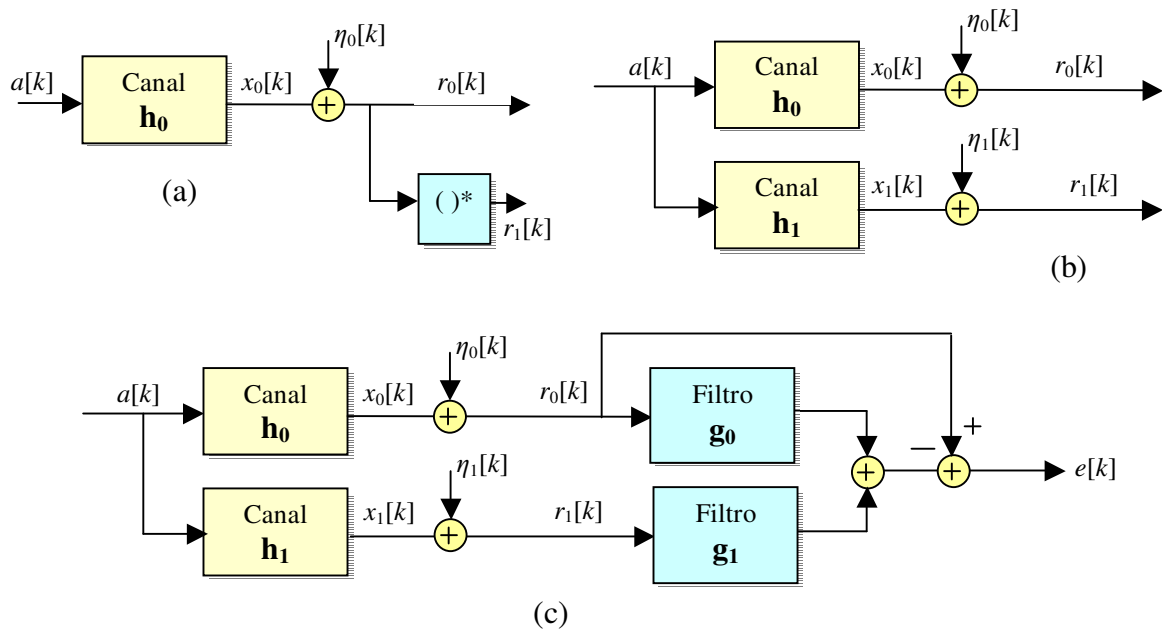


Figura 4.4. Equivalência entre o filtro de predição LL e um sistema polifásico: (a) sistema SISO mais operação de conjugação, (b) sistema SIMO equivalente, (c) filtro de erro de predição LL na forma polifásica equivalente.

Na Figura 4.4 valem as seguintes definições:  $\mathbf{h}_1 = \mathbf{h}_0^*$ , com  $\mathbf{h}_0 = [h_0 \ h_1 \ \dots \ h_L]^T$ ,  $\eta_1[k] = \eta_0^*[k]$  e  $r_1[k] = r_0^*[k]$ . Consideramos que o canal tem ordem  $L$  e os filtros  $\mathbf{g}_0$  e  $\mathbf{g}_1$  têm ordem  $M$ , sendo definidos por  $\mathbf{g}_0 = [0 \ f_1^* \ f_2^* \ \dots \ f_M^*]^T$  e  $\mathbf{g}_1 = [0 \ g_1^* \ g_2^* \ \dots \ g_M^*]^T$ . Podemos expressar o sinal recebido como

$$r_i[k] = \sum_{n=-\infty}^{\infty} a[n] h_i[k-n] + \eta_i[k] = x_i[k] + \eta_i[k], \quad i = 0, 1. \quad (4.20)$$

Podemos representar (4.20) vetorialmente como

$$\mathbf{r}[k] = \sum_n a[n] \mathbf{h}[k-n] + \boldsymbol{\eta}[k], \quad (4.21)$$

onde  $\mathbf{r}[k] \triangleq [r_0[k] \ r_1[k]]^T$ ,  $\mathbf{h}[k] \triangleq [h_0[k] \ h_1[k]]^T$  e  $\boldsymbol{\eta}[k] \triangleq [\eta_0[k] \ \eta_1[k]]^T$ . Para um vetor com  $N$  observações (amostras) da saída do canal podemos escrever:

$$\mathbf{r}_N[k] \triangleq [\mathbf{r}^T[k] \ \mathbf{r}^T[k-1] \ \dots \ \mathbf{r}^T[k-N+1]]^T. \quad (4.22)$$

Notamos que somente os símbolos  $a[m]$ , para  $m = k, k-1, \dots, k-N-M+1$ , irão afetar os dados observados  $\mathbf{r}_N[k]$ . A partir de (4.22) e fazendo  $m = k, k-1, \dots, k-N+1$ , podemos escrever:

$$\mathbf{r}_N[k] = \mathbf{H} \mathbf{a}_{N+L}[k] + \mathbf{n}_N[k], \quad (4.23)$$

onde  $\mathbf{a}_{N+L}[k]$  e  $\mathbf{n}_N[k]$  são definidos como

$$\begin{aligned} \mathbf{a}_{N+L}[k] &\triangleq [a[k] \ a[k-1] \ \dots \ a[k-N-L+1]]^T \\ \mathbf{n}_N[k] &\triangleq [\boldsymbol{\eta}^T[k] \ \boldsymbol{\eta}^T[k-1] \ \dots \ \boldsymbol{\eta}^T[k-N+1]]^T, \end{aligned} \quad (4.24)$$

a matriz  $\mathbf{H}$  tem dimensão  $2N \times (N+L)$  e é definida como

$$\mathbf{H} \triangleq \begin{bmatrix} \mathbf{h}[0] & \mathbf{h}[1] & \dots & \mathbf{h}[L] & \dots & 0 \\ 0 & \mathbf{h}[0] & \dots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \dots & \mathbf{h}[L-1] & \mathbf{h}[L] \end{bmatrix}. \quad (4.25)$$

A matriz de correlação de  $\mathbf{r}_N[k]$  definido em (4.22) é dada por:

$$\mathbf{R}_{\mathbf{r}_N} = E\{\mathbf{r}_N[k] \mathbf{r}_N^H[k]\} \quad (4.26)$$

$$\mathbf{R}_{r_N} = E \left\{ (\mathbf{H}\mathbf{a}_{N+L}[k] + \mathbf{n}_N[k]) (\mathbf{H}\mathbf{a}_{N+L}[k] + \mathbf{n}_N[k])^H \right\} \quad (4.27)$$

$$\mathbf{R}_{r_N} = \mathbf{H}E \left\{ \mathbf{a}_{N+L}[k] \mathbf{a}_{N+L}^T[k] \right\} \mathbf{H}^H + E \left\{ \mathbf{n}_N[k] \mathbf{n}_N^H[k] \right\} \quad (4.28)$$

$$\mathbf{R}_{r_N} = \mathbf{H}\mathbf{R}_a\mathbf{H}^H + \mathbf{R}_n, \quad (4.29)$$

uma vez que  $h_i[k]$  tem média zero. Usaremos este resultado posteriormente. Também usaremos a seguinte notação nas próximas subseções: para um vetor  $\mathbf{q} = [q_0 \ q_1 \ \dots \ q_N]^T$ ,  $\mathbf{q}(d)$  indicará o elemento  $q_d$ . Essa notação simplificará o desenvolvimento matemático dessas subseções e ajudará a mantermos uma certa compatibilidade com a notação usada por Giannakis e Halford (1997).

### 4.3.1 Equalização ZF para o FEPLL

Na ausência de ruído, uma escolha natural para a saída de um filtro de erro de predição é requerer  $e[k] = a[k]\mathbf{h}_0(0) = a[k]h_0$ . Observando a Figura 4.4(c) e admitindo filtros com a mesma ordem  $M$ , podemos escrever para a condição ZF:

$$-\sum_{\ell=0}^L \mathbf{h}_0(\ell) \mathbf{g}_0(m-\ell) - \sum_{\ell=0}^L \mathbf{h}_1(\ell) \mathbf{g}_1(m-\ell) + \mathbf{h}_0(m) = \mathbf{h}_0(m) \delta[m], \quad (4.30)$$

onde  $\delta[m]$  é a função impulso unitário. Agrupando os somatórios, podemos escrever (4.30) como

$$\sum_{i=0}^1 \sum_{\ell=0}^L \mathbf{h}_i(\ell) \mathbf{g}_i(m-\ell) = \mathbf{h}_0(m), \quad m \geq 1. \quad (4.31)$$

A partir de (4.31), podemos equacionar o seguinte sistema:

$$\mathbf{H}_z^T \mathbf{g} = \mathbf{h}_e, \quad (4.32)$$

onde  $\mathbf{g} = [\mathbf{g}_0(1) \ \mathbf{g}_1(1) \ \mathbf{g}_0(2) \ \mathbf{g}_1(2) \ \dots \ \mathbf{g}_0(M) \ \mathbf{g}_1(M)]^T$ ,  $\mathbf{h}_e = [h_1 \ h_2 \ \dots \ h_L \ 0 \ \dots \ 0]^T$  e

$\mathbf{H}_z \triangleq \mathbf{H}|_{N=L}$  tem dimensão  $2L \times (L+M)$ , sendo indicada em (4.33):



$$\mathbf{H}_Z^T \triangleq \begin{bmatrix} \mathbf{h}^T[0] & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{h}^T[1] & \mathbf{h}^T[0] & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}^T[L] & \mathbf{h}^T[L-1] & \ddots & \vdots \\ \vdots & \vdots & \ddots & \mathbf{h}^T[L-1] \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{h}^T[L] \end{bmatrix}. \quad (4.33)$$

Se a matriz  $\mathbf{H}_Z^T$  for quadrada, ou seja, fazendo  $N = L = M$ ,  $\mathbf{g}$  pode ser determinado de forma única calculando a inversa de  $\mathbf{H}_Z^T$ :

$$\mathbf{g} = (\mathbf{H}_Z^T)^{-1} \mathbf{h}_e. \quad (4.34)$$

No caso de  $M > L$ , a solução ZF não é única. Neste caso, empregando a restrição de norma mínima, podemos resolver (4.32) usando a pseudo-inversa, isto é,

$$\mathbf{g} = (\mathbf{H}_Z^T)^+ \mathbf{h}_e, \quad (4.35)$$

onde  $(\cdot)^+$  indica a pseudo-inversa. Aqui, destacamos a exigência do não compartilhamento de zeros em comum dos canais  $\mathbf{h}_0$  e  $\mathbf{h}_1$ , que é uma restrição de todos os métodos baseados em estatísticas de segunda ordem.

### 4.3.2 FEPLL como um equalizador cego

Focaremos agora a equalização cega com o filtro de erro de predição LL para os critérios ZF e MSE. Estaremos, portanto, considerando apenas o conjunto dos dados observados  $\mathbf{r}_N[k]$  em (4.23) para o cálculo dos equalizadores.

Nesta subseção, mostraremos que as soluções ZF e MSE levam ao mesmo equalizador quando o ruído tende a zero. Esse é um resultado esperado, mas que podemos considerar como um resultado formal novo para o FEPLL. Exemplificamos esse resultado no apêndice C através de um exemplo numérico.

### A. Critério ZF

Considerando que os dados  $a[k]$  têm média nula e são i.i.d. ( $\mathbf{R}_a = \sigma_a^2 \mathbf{I}$ ), e o canal livre de ruído ( $\mathbf{R}_n = \mathbf{0}$ ), a matriz de correlação em (4.29) se reduz a:

$$\mathbf{R}_{r_N} = \mathbf{R}_{x_N} = \sigma_a^2 \mathbf{H} \mathbf{H}^H. \quad (4.36)$$

Tomando o conjugado complexo de  $\mathbf{R}_{x_N}$  ( $\mathbf{R}_{x_N}^* = \sigma_a^2 \mathbf{H}^* \mathbf{H}^T$ ) e multiplicando à direita pelo vetor de coeficientes do equalizador indicado em (4.35), obtemos

$$\mathbf{R}_{x_N}^* \mathbf{g} = \sigma_a^2 \mathbf{H}^* \mathbf{h}_e. \quad (4.37)$$

Notando que  $(\mathbf{R}_{x_N}^*)^{-1} = 1/\sigma_a^2 (\mathbf{H}^T)^+ (\mathbf{H}^*)^+$ , a solução de (4.37) é

$$\mathbf{g} = \sigma_a^2 (\mathbf{H}^T)^+ (\mathbf{H}^*)^+ \mathbf{H}^* \mathbf{h}_e = (\mathbf{H}^T)^+ \mathbf{h}_e. \quad (4.38)$$

Quando a matriz de canal  $\mathbf{H}$  é uma matriz quadrada, isto é, o número de coeficientes do equalizador é igual ao comprimento do canal, trocamos a pseudo-inversa pela matriz inversa, e a solução encontrada em (4.38) é única. Note que a solução dada em (4.38) é a mesma indicada em (4.35), a menos de uma constante multiplicativa. O efeito dessa constante multiplicativa pode ser compensado por um circuito de controle automático de ganho, por exemplo. Havendo ambigüidade de fase, esta pode ser combatida pelo emprego de uma codificação diferencial.

Devemos notar que essa solução remove toda a IES (idealmente), mas, caso o sinal esteja contaminado por ruído, este pode ser amplificado e se torna colorido.

### B. Critério MMSE

Calcularemos agora  $\mathbf{g}$  baseado na minimização do valor esperado do módulo do erro  $e[k]$  ao quadrado,  $(|e[k]|^2)$ . Novamente, podemos expressar o erro  $e[k]$  como:

$$e[k] = r_0[k] - \sum_{i=0}^1 \sum_{\ell=0}^M \mathbf{g}_i(\ell) r_i[k-\ell]. \quad (4.39)$$

Nosso objetivo é encontrar  $\mathbf{g}$  tal que a função custo  $J \triangleq E\{|e[k]|^2\}$  seja minimizada.

Para isso, tomamos a derivada complexa em relação aos coeficientes desconhecidos do equalizador e igualamos o resultado a zero:

$$\frac{\partial J}{\partial \mathbf{g}_s^*[m]} = \frac{\partial}{\partial \mathbf{g}_s^*[m]} E \left\{ \left| r_0[k] - \sum_{i=0}^1 \sum_{\ell=0}^M \mathbf{g}_i(\ell) r_i[k-\ell] \right|^2 \right\} = 0, \quad (4.40)$$

$s = 0, 1$  e  $m = 1, 2, \dots, M$ . Levando em conta os critérios de ortogonalidade, após algumas simplificações, obtemos

$$E \left\{ r_0[k] r_s^*[k-m] - \sum_{i=0}^1 \sum_{\ell=0}^M \mathbf{g}_i(\ell) r_i[k-\ell] r_s^*[k-m] \right\} = 0, \quad (4.41)$$

$$E \{ r_0[k] r_s^*[k-m] \} - \sum_{i=0}^1 \sum_{\ell=0}^M \mathbf{g}_i(\ell) E \{ r_i[k-\ell] r_s^*[k-m] \} = 0. \quad (4.42)$$

Assumindo que o ruído é nulo e que os símbolos são i.i.d, temos

$$\begin{aligned} E \{ r_0[k] r_s^*[k-m] \} &= E \left\{ \left( \sum_{\ell} a[\ell] \mathbf{h}_0(k-\ell) \right) \left( \sum_{\ell} a[\ell] \mathbf{h}_s^*(k-\ell-m) \right) \right\} \\ &= \sigma_a^2 \sum_{\ell} \mathbf{h}_0(k-\ell) \mathbf{h}_s^*(k-\ell-m). \end{aligned} \quad (4.43)$$

Logo, podemos equacionar

$$\sum_{i=0}^1 \sum_{\ell=0}^M \mathbf{g}_i(\ell) E \{ x_i[k-\ell] x_s^*[k-m] \} = \sigma_a^2 \sum_{\ell} \mathbf{h}_0(k-\ell) \mathbf{h}_s^*(k-\ell-m), \quad (4.44)$$

que pode ser colocada na forma matricial para  $m = 1, 2, \dots, M$  e  $s = 0, 1$ , como:

$$\mathbf{R}_{\mathbf{X}_N}^* \mathbf{g} = \sigma_a^2 \mathbf{H}^* \mathbf{h}_e, \quad (4.45)$$

que é, portanto, idêntica a (4.37). Conseqüentemente, a minimização do erro quadrático do FEPLL leva a uma equalização do tipo ZF quando o ruído tende a zero.

No Apêndice C, Seção C.2, mostramos alguns resultados numéricos que ilustram os resultados teóricos apresentados acima.

## 4.4 FEPLL adaptativo

Apresentaremos agora um algoritmo adaptativo para o filtro de predição LL. Este algoritmo é, em essência, o algoritmo LMS aplicado ao preditor LL. Na próxima seção, apresentaremos estratégias para a aceleração da convergência do FEPLL.

A Figura 4.5 apresenta o esquema para o filtro de predição LL adaptativo, fazendo o uso de um dispositivo de decisão do tipo *hard* (decisão símbolo a símbolo). As equações (4.46) e (4.47) definem o algoritmo LMS adaptado para o filtro de predição LL. Ao contrário do algoritmo adaptativo apresentado para o equalizador, precisamos deixar os filtros  $\mathbf{f}$  e  $\mathbf{g}$  evidenciados, pois estes não são mais complexos conjugados um do outro. Assim, podemos escrever o algoritmo LMS como:

$$e[k] = x[k] - \mathbf{f}^H[k-1]\mathbf{x}[k-1] - \mathbf{g}^H[k-1]\mathbf{x}^*[k-1], \quad (4.46)$$

$$\begin{bmatrix} \mathbf{f}[k] \\ \mathbf{g}[k] \end{bmatrix} = \begin{bmatrix} \mathbf{f}[k-1] \\ \mathbf{g}[k-1] \end{bmatrix} + \mu e^*[k] \begin{bmatrix} \mathbf{x}^*[k-1] \\ \mathbf{x}[k-1] \end{bmatrix}, \quad (4.47)$$

onde  $\mu$  é o passo de adaptação. Podemos realizar as correções de fase e de ganho (CAG – controle automático de ganho) para compensar o efeito multiplicativo de  $h_0$ , usando as seguintes expressões (Rocha, 1996):

$$\phi[k] = \phi[k-1] + \mu_\phi \text{angle}(e[k] \exp(j\phi[k-1])), \quad (4.48)$$

$$G[k] = G[k-1] + \mu_G \left( E\{|a[k]|^2\} - |\tilde{a}[k]|^2 \right), \quad (4.49)$$

$$\gamma_k = \sqrt{|G[k]|}, \quad (4.50)$$

onde  $\mu_\phi$  e  $\mu_G$  correspondem aos passos de adaptação da fase e do ganho, respectivamente. Para a inicialização dos algoritmos podemos escolher  $\mathbf{f}[0] = \mathbf{g}[0] = \mathbf{0}$ ,  $\phi[0] = 0$  e  $G[0] = 1$ . Apresentaremos alguns resultados de simulação que ilustram o desempenho deste esquema de equalização cega no final deste capítulo.

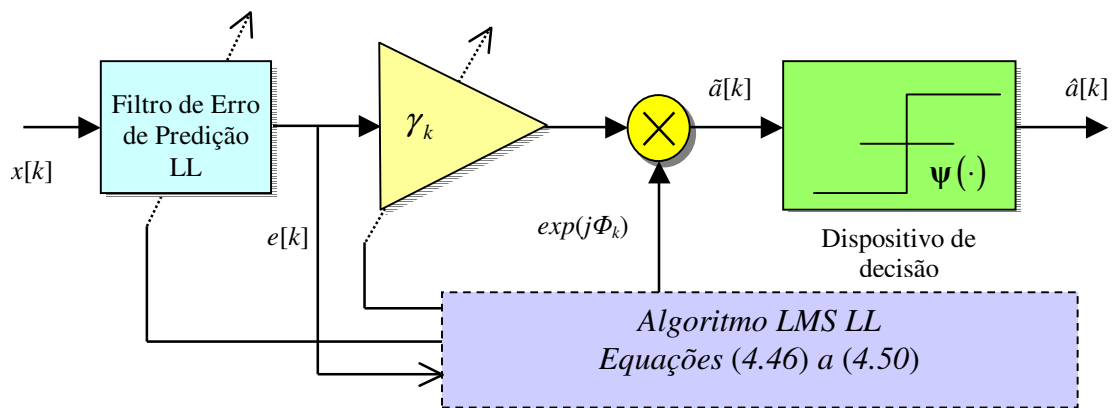


Figura 4.5. Esquema de equalização cega usando o filtro de predição LL.

## 4.5 FEPLL com a convergência acelerada

A maior vantagem do algoritmo LMS é a sua grande simplicidade computacional, pois possui como parâmetros de ajuste o passo de adaptação  $\mu$  e o número de coeficientes somente. No algoritmo LMS, a matriz de correlação dos dados e o vetor de correlação cruzada entre os dados e a resposta desejada são trocados por suas estimativas instantâneas. Entretanto, o preço a ser pago por essa reduzida complexidade computacional é a baixa taxa de convergência deste algoritmo, especialmente para o caso de equalização ou estimação cegas.

Uma forma de obtermos uma melhor taxa de convergência é empregar um algoritmo com um maior número de parâmetros e, conseqüentemente, uma maior complexidade computacional. Uma estratégia, com um custo computacional relativamente baixo, é o emprego de técnicas de passo variável aplicadas ao algoritmo LMS, como já utilizamos no Capítulo 3. Uma outra possibilidade é a utilização do algoritmo RLS (*Recursive Least Squares*). Na próxima subseção, apresentamos um algoritmo LMS de passo variável e na subseção seguinte, um algoritmo RLS para o FEPLL.

### 4.5.1 FEPLL de passo variável

Dentre as técnicas de passo variável comparadas em (Aquino *et al.*, 2006), o algoritmo VSS-LMS (*Variable Step-Size LMS algorithm*) proposto por Kwong e Johnston (1992) é a que se aplica mais facilmente ao algoritmo LMS apresentado na seção anterior. No estudo comparativo apresentado por Zipf *et al.* (2007), este é também um dos algoritmos de passo

variável que se destaca por sua robustez e elevada taxa de convergência. Aplicando o algoritmo VSS ao FEPLL, os passos de adaptação  $\mu$ ,  $\mu_G$  e  $\mu_\phi$  também se tornam variáveis. Com isso podemos esperar uma melhora no desempenho do algoritmo em relação à taxa de convergência e no desajuste após a convergência. Apresentamos os resultados desta subseção no I2TS-2007 (Aquino *et al.*, 2007e).

No algoritmo VSS o passo de adaptação é ajustado de acordo com quadrado do erro instantâneo:

$$\mu'_x[k] = \alpha\mu_x[k-1] + \beta_x |\varepsilon_x[k-1]|^2, \quad (4.51)$$

com  $0 \ll \alpha < 1$ ,  $\beta_x > 0$  e

$$\mu_x[k] = \begin{cases} \mu_{\max}, & \text{se } \mu'_x[k] > \mu_{\max} \\ \mu_{\min}, & \text{se } \mu'_x[k] < \mu_{\min} \\ \mu'_x[k], & \text{outros valores} \end{cases} \quad (4.52)$$

onde  $0 < \mu_{\min} < \mu_{\max}$  e  $\varepsilon_x[\cdot]$  é um *erro* que depende do passo que está sendo adaptado, como explicaremos a seguir.

O valor inicial de  $\mu[0]$  é usualmente feito igual a  $\mu_{\max}$ . Note que o valor de  $\mu_{\max}$  deve ser tal que a estabilidade do algoritmo esteja garantida. Como podemos observar a partir de (4.51), o passo de adaptação é sempre positivo e é controlado pelos parâmetros  $\alpha$  e  $\beta_x$ . Tipicamente, 0,98 é um valor adequado para  $\alpha$ , enquanto que o valor de  $\beta_x$  é usualmente pequeno (da ordem de  $10^{-5}$  nas nossas simulações). Naturalmente, a adoção do passo variável acarreta em uma maior complexidade computacional do algoritmo.

Para cada passo ( $\mu$ ,  $\mu_G$  ou  $\mu_\phi$ ), o erro  $\varepsilon_x$  é calculado de forma diversa:

$$\begin{aligned} \mu &\rightarrow \varepsilon_\mu[k] = \Psi(y[k]) - y[k], \\ \mu_G &\rightarrow \varepsilon_G[k] = \Psi(y[k]) - \Re\{y[k]\}, \\ \mu_\phi &\rightarrow \varepsilon_\phi[k] = \Im\{y[k]\}, \end{aligned} \quad (4.53)$$

onde  $\Re$  indica a parte real e  $\Im$  a parte imaginária. Apresentaremos diversos resultados de simulação que mostram o bom funcionamento do algoritmo LMS largamente linear de passo variável proposto ao final deste capítulo, na seção de resultados.

### 4.5.2 FEPLL com algoritmo RLS

Os filtros adaptativos LS (*least-squares*) são projetados de forma que os ajustes de seus coeficientes sempre minimizem o erro quadrático total desde o início da operação de filtragem até o instante atual. Portanto, os coeficientes do filtro, no instante  $k$ , são escolhidos de forma a minimizar a função custo:

$$J_{RLS}[k] = \sum_{j=0}^k \lambda^{k-j} |e[j]|^2 = \sum_{j=0}^k \lambda^{k-j} |d[j] - \mathbf{f}^H \mathbf{x}[j]|^2, \quad (4.54)$$

onde  $e[j]$  é o erro instantâneo,  $d[j]$  o sinal desejado e a constante  $\lambda, 0 \ll \lambda \leq 1$ , é um *fator de esquecimento* (*forgetting factor*). Este fator de esquecimento é importante em situações em que o ambiente é não-estacionário, pois, desta forma, os dados mais antigos têm menor peso com relação aos dados mais recentes. Isto possibilita um melhor acompanhamento do canal pelo filtro em um ambiente não-estacionário. Além disto, quando  $\lambda < 1$ , o algoritmo RLS se torna numericamente mais estável, especialmente em relação aos efeitos de precisão numérica finita. Podemos observar em (4.54) que os coeficientes do filtro permanecem os mesmos durante a janela de observação  $0 \leq j \leq k$ .

Como o algoritmo RLS é bem documentado na literatura (ver, por exemplo, Proakis, 1995; Sayed, 2003; Manolakis *et al.*, 2000; Haykin, 2002) optamos por apresentá-lo na Tabela 4.1, onde  $\delta$ <sup>14</sup> é um número real positivo chamado de parâmetro de regularização e  $\mathbf{I}$  é uma matriz identidade. A adição desta matriz tem o efeito de tornar a matriz de correlação não singular em todas as etapas de cálculo, partindo de  $k = 0$ .

Apresentaremos agora um algoritmo RLS largamente linear, tendo por base o algoritmo RLS tradicional descrito na Tabela 4.1, já incluindo os ajustes de fase e de ganho, para uso no filtro de predição LL.

Apesar de termos dois filtros para adaptar ( $\mathbf{f}$  e  $\mathbf{g}$ ), a complexidade computacional pode ser menor que para um filtro de predição estritamente linear. Isto ocorre porque a ordem do filtro de predição LL tende a ser menor (da mesma ordem do canal). Além disso, podemos aproveitar alguns cálculos que envolvem matrizes e vetores para cálculo dos coeficientes do filtro  $\mathbf{f}$ , aplicando apenas a operação de conjugação, para o cálculo dos coeficientes do filtro  $\mathbf{g}$ , em lugar de refazer estes cálculos. Este algoritmo RLS modificado é mostrado de forma detalhada na Tabela 4.2.

<sup>14</sup> Este valor é “pequeno” para uma alta relação sinal-ruído e “grande” para uma baixa relação sinal-ruído.

TABELA 4.1: ALGORITMO RLS

<i>Inicialização</i>	$k = 1, 0 \ll \lambda \leq 1,$ $\hat{\mathbf{f}}[0] = \mathbf{0}, \mathbf{P} = \delta^{-1} \mathbf{I}_{M \times M}.$
<i>Laço de repetição</i> Passo – 1	$\beta[k] = 1 / (1 + \lambda^{-1} \mathbf{x}[k] \mathbf{P}[k-1] \mathbf{x}[k]^H)$ $\mathbf{b}[k] = \lambda^{-1} \mathbf{P}[k] \mathbf{x}^H[k] \beta[k]$ $\zeta[k] = d[k] - \hat{\mathbf{f}}^H[k-1] \mathbf{x}[k]$ $\hat{\mathbf{f}}[k] = \hat{\mathbf{f}}[k-1] + \mathbf{b}[k] \zeta^*[k]$ $\mathbf{P}[k] = \lambda^{-1} \mathbf{P}[k-1] - \mathbf{b}[k] \mathbf{b}[k]^H / \beta[k]$
<i>Fim da repetição,</i> <i>voltar ao Passo – 1</i>	$k = k + 1$

TABELA 4.2: ALGORITMO RLS PARA O FILTRO DE ERRO DE PREDIÇÃO LL

<i>Inicialização</i>	$k = 1, p = 0, 0 \ll \lambda < 1, G_{-1} = 2, \phi_{-1} = 0$ $\mathbf{f} = \mathbf{0}_{1 \times M}, \mathbf{g} = \mathbf{0}_{1 \times M}, \mathbf{x}_f = \mathbf{0}_{1 \times M}, \mathbf{w} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix},$ $\mathbf{P} = \delta^{-1} \mathbf{I}_{M \times M}.$
<i>Laço de repetição</i> Passo – 1	$p = 0$ <b>enquanto</b> ( $p < M$ ) <b>se</b> ( $k-p > 0$ ) $\mathbf{x}_f[p] = x[k-p]$ <b>fim se</b> $p = p+1$ <b>fim enquanto</b> $\beta[k] = 1 / (1 + \lambda^{-1} \mathbf{x}_f[k] \mathbf{P}[k] \mathbf{x}_f^H[k])$ $\mathbf{B}[k] = \lambda^{-1} \mathbf{P}[k] \mathbf{x}_f^H[k] \beta[k]$ $\zeta[k] = x[k] - [\mathbf{x}_f^T[k] \quad \mathbf{x}_f^H[k]] \mathbf{w}[k]$ $\mathbf{w}[k] = \mathbf{w}[k] + \begin{bmatrix} \mathbf{B}[k] \\ \mathbf{B}^*[k] \end{bmatrix} \zeta[k]$ $\mathbf{P}[k] = \lambda^{-1} \mathbf{P}[k] - \mathbf{B}[k] \mathbf{B}^H[k] / \beta[k]$
<i>Ajuste de ganho e de fase</i> Passo – 2	$\phi_k = \phi_{k-1} + \mu_\phi \text{angle}(\zeta[k] \exp(j\phi_{k-1}))$ $\zeta[k] = \zeta[k] \exp(j\phi_k)$ $G_k = G_{k-1} + \mu_G (E\{ a[k] ^2\} - G_{k-1}  \zeta[k] ^2)$ $\zeta[k] = \zeta[k] \sqrt{G_k}$
<i>Fim da repetição,</i> <i>voltar ao Passo – 1</i>	$k = k + 1$



Apresentaremos um exemplo de equalização aplicando esse algoritmo RLS largamente linear na seção de resultados.

## 4.6 Equalização com atraso usando um conjunto de FEPLL's

O FEPLL, inerentemente, apresenta atraso zero, ou seja, o símbolo na saída do equalizador no instante discreto  $k$  é uma estimativa do símbolo transmitido no instante  $k$ . Isto, entretanto, pode não ser a melhor solução, especialmente quando o canal considerado é de fase não-mínima. A seleção de um atraso mais apropriado, que reflita a distribuição de energia do canal, pode resultar em um melhor desempenho do equalizador. Uma solução encontrada na literatura para esse problema, no contexto de equalização multicanal, é a utilização de predição multi-atraso (*prediction multi-step*) (Gesbert e Duhamel, 1997; Slock e Papadias, 1995). Uma outra possível solução pode ser baseada em equalizadores (filtros) mutuamente referenciados (Gesbert *et al.*, 1997). Também encontramos na literatura a combinação de FEPL direto e FEPL reverso (BPEF – *backward prediction error filter*) como solução para o problema de equalização cega, ver Barbosa *et al.* (2000) e as referências lá citadas.

Nesta seção, apresentamos uma nova solução usando um conjunto de FEPLL's. Na saída de cada FEPLL é gerado um símbolo estimado com um atraso distinto, isto é, a saída desse conjunto de FEPLL's forma o vetor de dados estimados  $[\tilde{a}[k], \tilde{a}[k-1], \dots, \tilde{a}[k-D]]^T$ . A cada instante é selecionada a melhor saída do conjunto de FEPLL's, que pode ser usada como uma seqüência de treinamento para um equalizador largamente linear *treinado*. Seguindo a nomenclatura sugerida por Gesbert *et al.* (1999), o equalizador proposto também poderia ser chamado de “*super equalizador baseado em FEPLL*”. Apresentamos essa nova estrutura no SBrT 2007 (Aquino *et al.*, 2007c). A seguir, detalhamos a estrutura proposta.

### 4.6.1 Predição multi-atraso

A predição multi-passo (ou multi-atraso) pode ser usada para equalização cega com um atraso de decisão arbitrário em sistemas SIMO (Gesbert e Duhamel, 1997; Slock e Papadias, 1995). Na ausência de ruído, o erro de predição multi-passo pode ser modelado como a saída de um canal truncado. Assim, para um FEPLL com um passo (atraso)  $\Delta > 1$  (FEPLL “ $\Delta$ -step”) podemos calcular:

$$e_{\Delta}[k] = \sum_{m=0}^{\Delta-1} h[m]a[k-m]. \quad (4.55)$$

Se for usado, em cascata, um filtro de erro de predição reverso (*backward linear prediction*) largamente linear de ordem suficiente, o erro de predição ótimo torna-se:

$$e_B[k] = h[\Delta-1]a[k-\Delta+1]. \quad (4.56)$$

A grande vantagem desse método é que torna possível selecionar o coeficiente de maior energia do canal e, assim, facilitar a correta estimação do símbolo transmitido (com um atraso igual a  $\Delta - 1$ ). A solução expressa por (4.56) pode ser obtida de uma forma alternativa, como será descrito a seguir, usando-se apenas FEPLLs.

#### 4.6.2 Esquema proposto

Considerando-se uma alta relação sinal ruído, para o FEPLL descrito nas seções anteriores (com  $\Delta = 1$ , atraso zero), podemos conseguir uma boa estimativa de  $h[0]$  e  $\hat{a}[k]$ , sendo o erro de decisão direta expresso por  $\varepsilon_{DD0} = \hat{a}[k] - \Re(y[k])$ , onde  $\Re$  indica a parte real. Quando ocorre a convergência do algoritmo, após um número suficiente de amostras (o que vai depender do canal a ser equalizado), o erro de decisão direta se torna pequeno,  $\hat{a}[k] \approx a[k]$ , e o valor estimado de  $h[0]$  no instante  $k$  é dado por (ver Figura 4.6):

$$\hat{h}[0] = \exp(-j\Phi_k) / \sqrt{\gamma_k} \approx h[0] \quad (4.57)$$

e a saída do FEPLL, sem a correção de ganho ou fase, é

$$e_{LL0}[k] = h[0]a[k] + n_1[k], \quad (4.58)$$

onde  $n_1[k]$  é o ruído na saída do filtro. O símbolo decidido pode ser calculado por:

$$\hat{a}[k] = \Psi(y_0[k]) = \Psi(e_{LL0}[k] / \hat{h}[0]), \quad (4.59)$$

onde  $y_0[k]$  é a saída deste primeiro FEPLL. A Figura 4.6 ilustra o esquema proposto e o equacionamento já realizado. Na saída do FEPLL com  $\Delta = 2$ , obviamente sem qualquer ajuste de fase ou ganho, temos

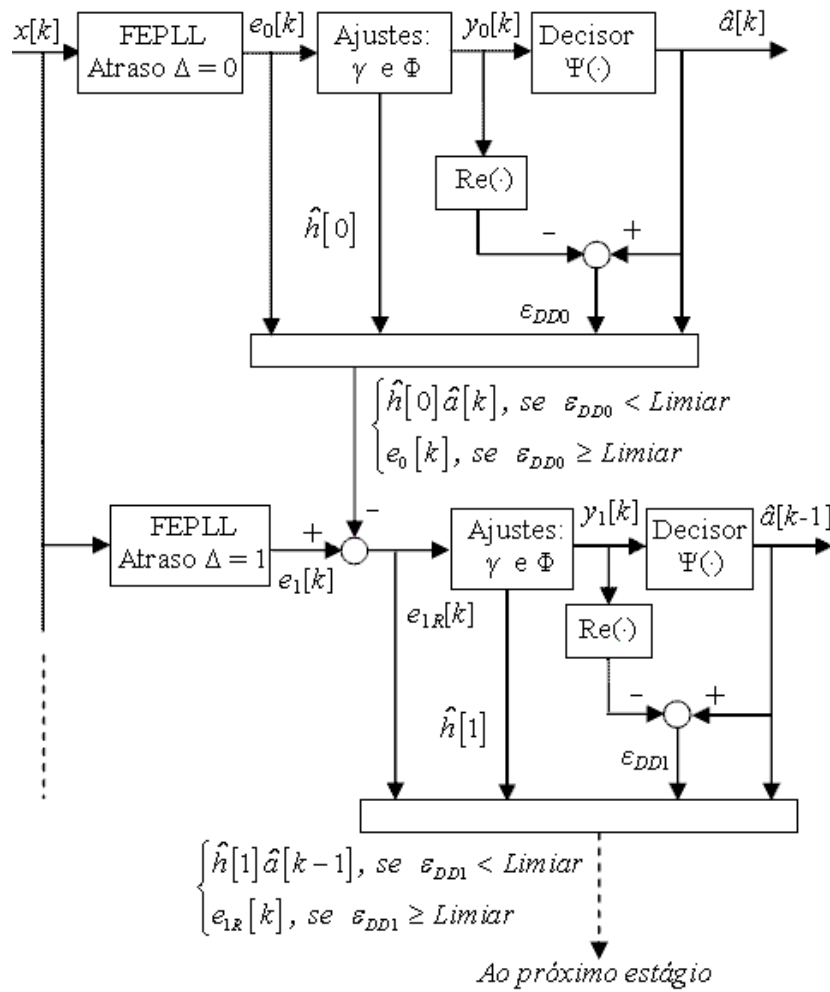


Figura 4.6: Diagrama de blocos dos dois primeiros estágios do esquema de equalização robusto proposto usando FEPLL adaptativo.

$$e_{LL1}[k] = h[0]a[k] + h[1]a[k-1] + n_2[k]. \quad (4.60)$$

Logo, calculando

$$e_{1R}[k] = e_{LL1}[k] - e_{LL0}[k], \quad (4.61)$$

resulta em  $h[1]a[k-1]$  mais ruído. Se o erro de decisão direta ( $\varepsilon_{DD}$ ) do primeiro FEPLL estiver abaixo de um certo limiar, é mais conveniente calcular

$$\begin{aligned} e_{1R}[k] &= e_{LL1}[k] - \hat{h}[0]\hat{a}[k] \\ &= h[1]a[k-1] + n_{1R}[k], \end{aligned} \quad (4.62)$$

pois não será adicionado o ruído do estágio anterior. Na saída deste FEPLL, o símbolo será decidido com base em  $e_{1R}[k]$  e  $\hat{h}[1]$ :

$$\hat{a}[k-1] = \Psi(y_1[k]) = \Psi(e_{1R}[k]/\hat{h}[1]), \quad (4.63)$$

onde  $y_1[k]$  é a saída deste segundo FEPLL e  $\Psi(\cdot)$  representa o dispositivo de decisão. O erro de decisão para este estágio, no instante  $k$ , é expresso por:

$$\varepsilon_{DD1} = \hat{a}[k-1] - \Re(y_1[k]), \quad (4.64)$$

onde  $\hat{h}[1]$  é uma estimativa de  $h[1]$ . Todo esse procedimento está esquematizado na Figura 4.6. Na saída de um FEPLL com  $\Delta = 3$  tem-se:

$$e_{LL2}[k] = \sum_{m=0}^2 h[0]a[k-m] + n_2[k]. \quad (4.65)$$

Usando-se os resultados e procedimentos anteriores, podemos estimar  $h[2]$  e  $a[k-2]$ :

$$\begin{aligned} e_{2R}[k] &= e_{LL2}[k] - \hat{h}[0]a[k] - \hat{h}[1]a[k-1] \\ &= h[2]a[k-2] + n_{2R}[k]. \end{aligned} \quad (4.66)$$

Logo, com estes três filtros pode-se estimar  $a[k]$ ,  $a[k-1]$ ,  $a[k-2]$  e os coeficientes  $h[0]$ ,  $h[1]$  e  $h[2]$  do canal. De uma forma geral, tendo-se vários FEPLL's com atrasos 0, 1, ...,  $D$ , podemos estimar os coeficientes  $h[0]$ , ...,  $h[D]$ , e os símbolos  $a[k]$ , ...,  $a[k-D]$ . Os símbolos estimados, depois de alinhados apropriadamente, podem ser combinados para gerar uma estimativa mais robusta de  $a[k-D]$ . Por exemplo, no instante  $k$ , pode-se calcular:

$$y_R[k] = \frac{\frac{e_{LL0}[k-D]}{\hat{h}[0]} + \frac{e_{1R}[k-D+1]}{\hat{h}[1]} + \dots + \frac{e_{DR}[k]}{\hat{h}[D]}}{D+1}, \quad (4.67)$$

que leva em conta a contribuição de cada FEPLL. Após o alinhamento de todas as saídas, podemos, ainda, usar o seguinte critério para tornar o procedimento mais robusto: em cada instante  $k$ , a saída *selecionada* será aquela que apresentar menor erro de decisão direta dentre as saídas disponíveis.

Uma outra idéia que poderíamos explorar é tornar a soma em (4.67) ponderada, isto é, atribuir um peso em cada parcela da soma proporcional ao valor absoluto de coeficiente estimado do canal. Com isto, a influência do ruído poderia ser minimizada.

Finalmente, esta saída selecionada (saída *ótima*) é usada como uma *seqüência de treinamento* para um equalizador LMS largamente linear treinado de passo fixo  $\mu_i$ . Quando o módulo do erro de decisão direta deste equalizador treinado ficar abaixo de um certo valor limiar (0,85 nas nossas simulações<sup>15</sup>), ele passa a operar no modo de decisão direta. Desta forma, o desempenho final do sistema, para uma relação sinal-ruído apenas moderada (~ 15 dB), pode se aproximar muito daquele que seria obtido por uma equalização largamente linear treinada convencional, como mostraremos na seção de resultados de simulação.

A Figura 4.7 mostra o esquema final de equalização proposto, com a omissão de alguns detalhes já apresentados na Figura 4.6. Notamos que o sinal equalizado é  $y_i[k]$  e o símbolo decidido é  $\hat{a}[k] = \Psi(y_i[k])$ .

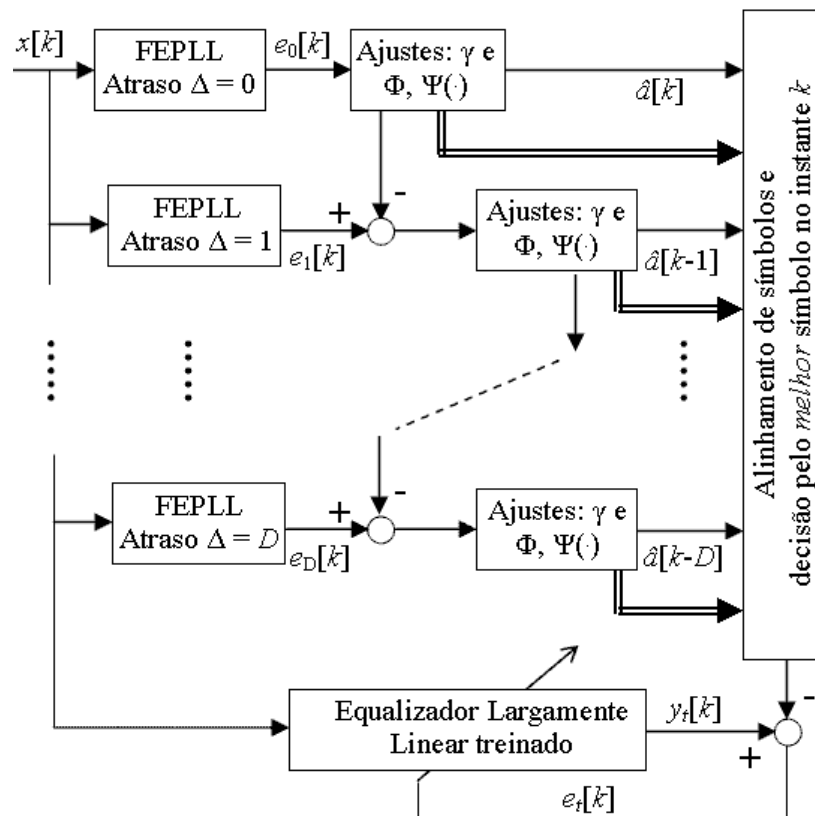


Figura 4.7: Diagrama de blocos completo do esquema de equalização proposto.

<sup>15</sup> Outros valores poderiam ser usados, por exemplo, 0,60. Quanto menor o valor do limiar, maior a confiabilidade na decisão e mais demorada se torna a convergência.

Em relação à complexidade computacional desta estrutura, podemos estimar, *grosso modo*, que ela é dada por:  $(N+1)O(P)$ , sendo  $N$  o número de FEPLL usados e  $O(P)$  o número de operações (somadas e multiplicações) para um único FEPLL.

## 4.7 Implementação do FEPLL em FPGA

Finalmente, apresentamos nesta seção uma implementação em FPGA de um FEPLL de cinco coeficientes. Esta implementação inclui a correção de fase e ganho, mas com o algoritmo modificado em relação ao descrito pelas equações (4.46) a (4.50). As modificações foram necessárias para manter uma baixa complexidade e porque todas as operações (somadas, multiplicações) são realizadas com números reais e não com valores complexos. Outro detalhe é que, devido à implementação realizada ser do tipo paralela e a latência decorrente das operações de multiplicação, o algoritmo realizado é um LMS atrasado (Delayed LMS). Isso, entretanto, não é crítico para esta aplicação, pois o passo de adaptação é pequeno (Haimi-Cohen *et al.*, 1990; Long *et al.*, 1989; Tobias *et al.*, 2000). Na Figura 4.8 vemos detalhes da implementação do ajuste de ganho e na Figura 4.9 temos uma visão geral do FEPLL implementado.

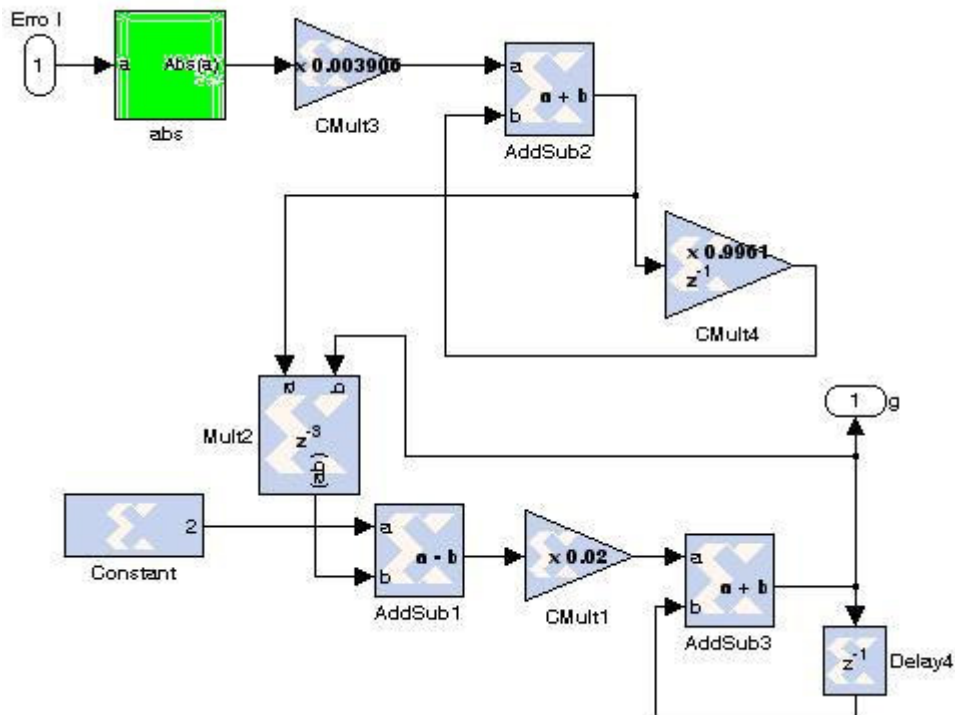


Figura 4.8. Implementação em FPGA: detalhe do ajuste do ganho.

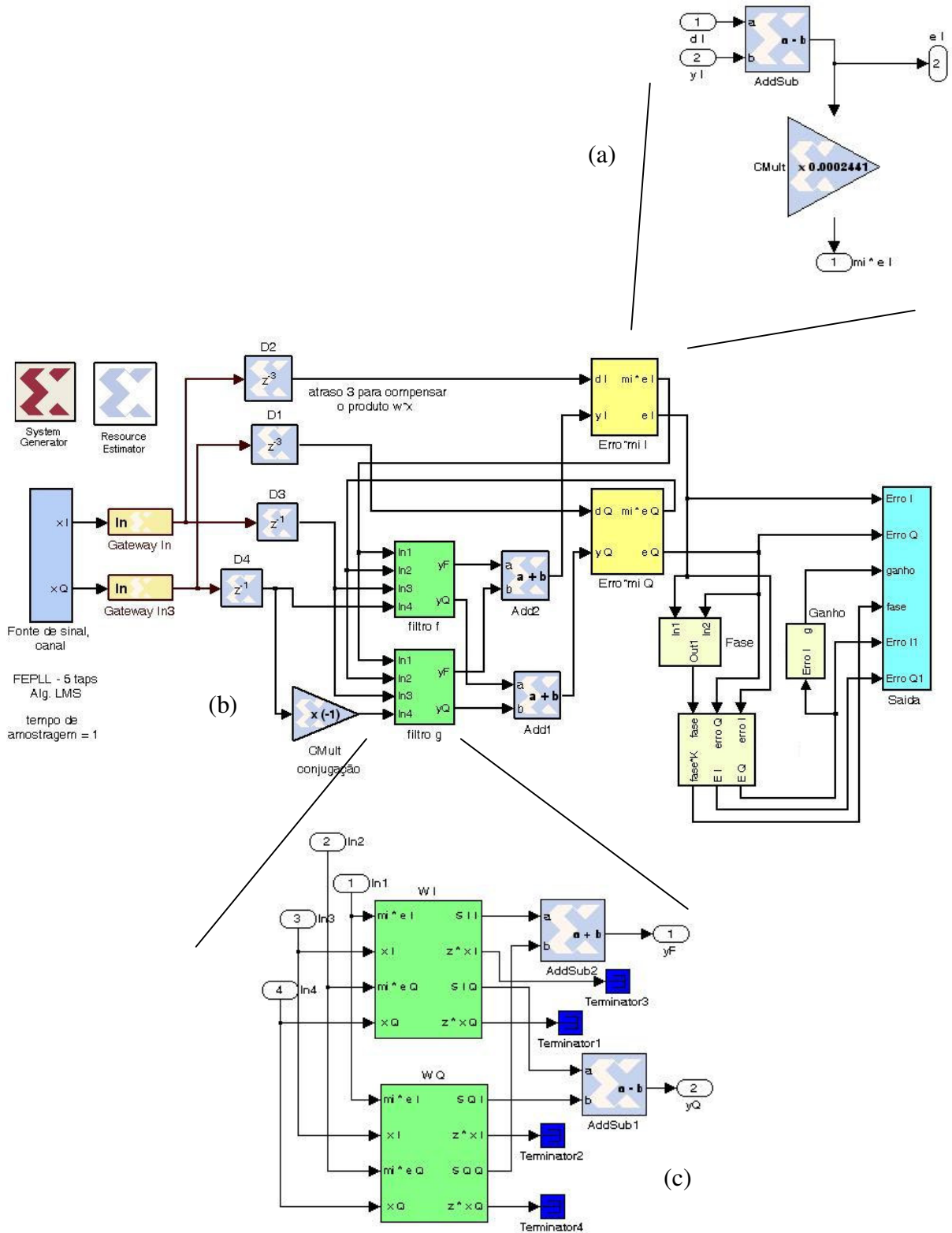


Figura 4.9. Implementação do FEPLL em FPGA: (a) detalhe do cálculo do erro, (b) visão geral do FEPLL, (c) detalhamento de um dos filtros transversais.

O ajuste de ganho, de acordo com a Figura 4.8, é dado por:

$$g[k] = g[k-1] + (2^{-6} + 2^{-8}) \cdot (2 - g[k-3] E\{|e_l[k]|\}), \quad (4.87)$$

com

$$E\{|e_l[k]|\} \cong 2^{-8} |e_l[k]| + (1 - 2^{-8}) |e_l[k-1]|. \quad (4.88)$$

Observamos que o uso de constantes em potência de 2 permite que as multiplicações sejam facilmente implementadas apenas por deslocamentos (baixo custo em *hardware*) e com latência nula. Outros detalhes de implementação e resultados de simulação, o leitor pode conferir no Apêndice C.

## 4.8 Resultados de Simulação

Nesta seção apresentaremos diversos resultados de simulação para ilustrar e complementar toda a teoria abordada neste capítulo. Cada subseção contempla uma seção correspondente já apresentada. A menos que indicado de forma diferente, em todas as simulações os símbolos a serem equalizados pertencem a uma constelação 4-PAM e o ruído é gaussiano, com igual variância nos eixos real e imaginário, próprio e de média nula.

### 4.8.1 FEPLL ótimos

*Canal complexo de fase não-mínima e com um nulo: equalização ZF.* Seja o canal definido por  $\mathbf{h}_c = [0,5408+0,0854j; 0,0882-0,2078j; -0,2300-0,6848j; -0,2676-0,2362j]$ , cuja resposta em frequência está indicada na Figura 4.10. Podemos calcular os filtros ótimos, segundo o critério ZF, usando (4.19), obtendo os seguintes valores:  $\mathbf{f}_{opt} = [0,2754j; 0,0251+0,6116j; 0,1573+0,3600j]$  e  $\mathbf{g}_{opt} = [0,1335+0,0878j; -0,3843+0,7115j; -0,3767+0,1115j]$ . Este equalizador consegue compensar perfeitamente este canal, a menos do ganho complexo  $0,5408+0,0854j$ , como mostra a Figura 4.10.



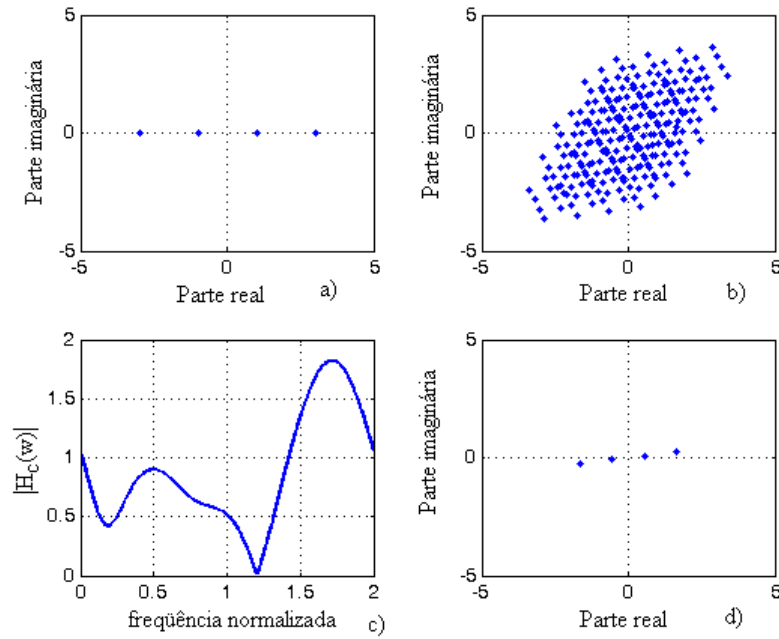


Figura 4.10. Equalização com filtro de predição largamente linear (critério ZF), sem compensação do ganho complexo do canal. a) Sinal transmitido, b) constelação após o canal, c) resposta em frequência do canal, d) sinal equalizado.

*Canal complexo de fase não-mínima e com um nulo espectral: equalização MMSE.* Seja o canal definido por  $\mathbf{h}_c = [0,5503+0,0869j \quad 0,5310-0,2069j \quad 0,4114-0,4059j \quad 0,1178-0,1301j]$ , cuja resposta em frequência está indicada na Figura 4.11(c). Consideramos uma relação sinal ruído de 30 dB. Os filtros ótimos, segundo o critério MSE, são  $\mathbf{f}_{opt} = [0,4487-0,3576j; -0,1298-0,1750j; \quad 0,0193+0,2065j]$  e  $\mathbf{g}_{opt} = [0,4607-0,0165j; \quad 0,1262-0,1646j; -0,1971+0,0248j]$ . Mostramos o resultado desta equalização na Figura 4.11, já levando em conta o ganho complexo introduzido pelo canal compensado. Podemos observar que ocorre uma amplificação significativa do ruído. Neste exemplo, a variância do erro na saída do filtro de predição LL é cerca de 5,6 vezes o valor da variância do ruído na entrada deste equalizador. Para este canal, se o critério de equalização fosse o ZF, a variância do erro na saída do filtro de predição seria cerca de 16,5 vezes maior que na entrada.

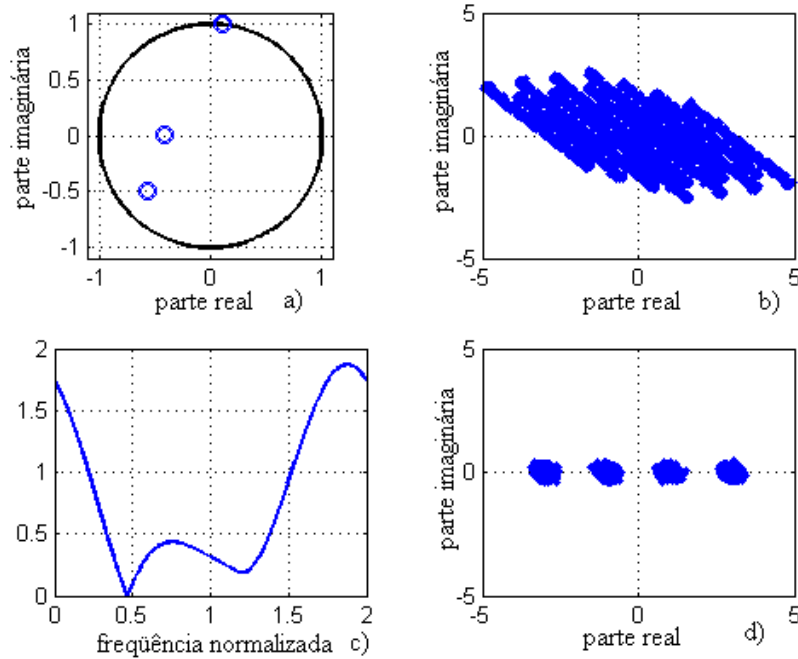


Figura 4.11. Equalização usando o critério MSE, com compensação do ganho complexo introduzido pelo canal: a) zeros do canal, b) constelação após o canal, c) resposta em frequência do canal, d) sinal equalizado. Relação sinal-ruído de 30 dB.

#### 4.8.2 FEPLL adaptativo com passo fixo

*Equalização LMS.* Para o canal  $\mathbf{h}_c$  já definido na segunda simulação desta seção e uma relação sinal ruído de 30 dB, os coeficientes dos filtros  $\mathbf{f}$  e  $\mathbf{g}$  encontrados adaptativamente, depois de 15000 iterações (símbolos transmitidos), usando o algoritmo LMS (4.46 – 4.50) são:  $\mathbf{f} = [0,4528-0,3398j; -0,1398-0,1852j; 0,0383+0,1966j]$  e  $\mathbf{g} = [0,4675-0,0445j; 0,1031-0,1334j; -0,1614+0,0286j]$ . Podemos observar que estes valores estão próximos dos valores ótimos calculados na simulação anterior. A Figura 4.12 (a) e (b) apresenta a constelação equalizada (últimos 1500 símbolos transmitidos) e a evolução do diagrama de olho. Observamos que o olho abre com cerca de 6000 iterações. Os passos usados foram:  $\mu = 0,0001$ ;  $\mu_\phi = 0,0007$ ; e  $\mu_G = 0,004$ . A Figura 4.12 (c) e (d) mostra a evolução de  $\Phi_k$  (em radianos) e de  $G_k$ .

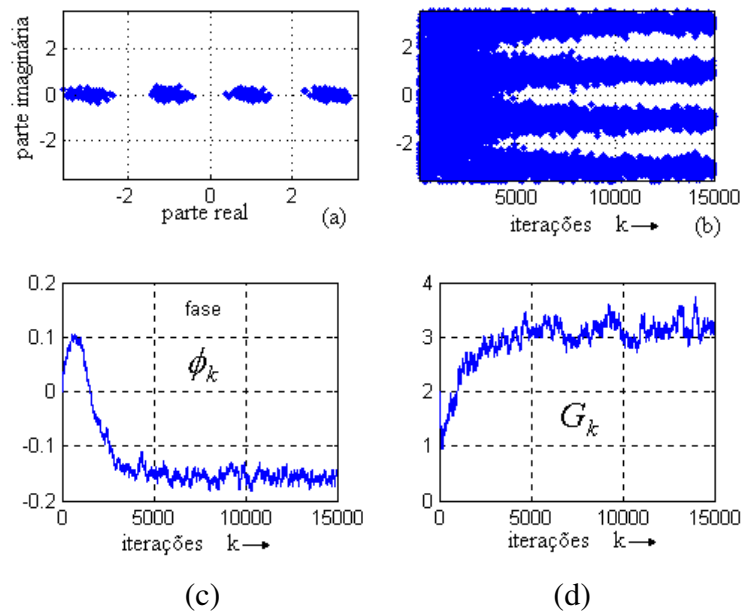


Figura 4.12. Equalização cega – (a) diagrama de olho após a equalização, (b) evolução da constelação, (c) compensações de fase e (d) de ganho. Relação sinal-ruído de 30 dB.

*Equalização do canal definido por* (Porat e Friedlander, 1991). Usaremos agora o filtro de predição LL para equalizar o canal definido por Porat e Friedlander (1991) e uma relação sinal-ruído de 40 dB. Os resultados estão indicados na Figura 4.13. Nesta figura, podemos ver novamente a constelação equalizada (últimos 1500 símbolos transmitidos) e a evolução do diagrama de olho. Observamos que o olho abre com cerca de 12000 iterações (símbolos transmitidos) e que este canal é mais difícil de ser equalizado do que o da simulação anterior. Os passos usados foram:  $\mu = 0,0001$ ;  $\mu_\phi = 0,0005$ ; e  $\mu_G = 0,004$ .

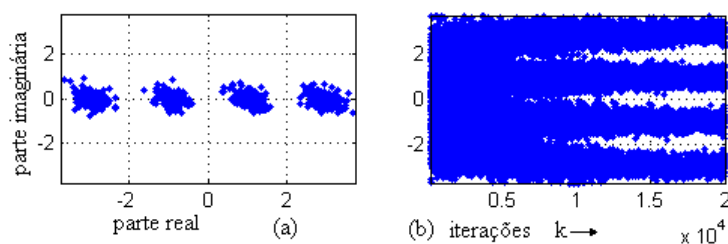


Figura 4.13. Equalização cega: (a) diagrama de olho após a equalização para o canal definido por Porat e Friedlander, (b) evolução da constelação. Relação sinal-ruído de 40 dB.

*Comparação com outras técnicas cegas.* Faremos agora uma comparação do FEPLL com um filtro de erro de predição linear e várias outras técnicas cegas lineares em um canal de fase mínima definido como  $\mathbf{h}_m = [0,7069; -0,1909-0,1767j; 0,1612+ 0,5433j; 0,1420-0,3021j]$ . Este canal é de fase mínima e não apresenta nulo espectral, mas dois de seus zeros estão

próximos do CRU. Consideramos uma relação sinal ruído de 30 dB. Novamente, a constelação usada será uma 4-PAM. Escolhemos este tipo de canal para que o filtro de erro de predição linear e os demais equalizadores cegos conseguissem “abrir o olho” e tivessem um erro residual pequeno após a convergência. As técnicas cegas consideradas foram: os algoritmos de Sato, Godard (CMA), Benveniste-Goursat, todos da família Bussgang de algoritmos cegos (Haykin, 2002), e um FEPL. Os principais algoritmos da família Bussgang são detalhados e explicados no Apêndice C.

Os parâmetros usados nos algoritmos e outras informações para a comparação dessas técnicas estão resumidos na Tabela 4.3. A Figura 4.13 mostra o diagrama de olho para uma realização destes algoritmos e a Figura 4.14 mostra a média de 1000 realizações para a energia do erro de decisão direta. Claramente, os melhores resultados para a equalização desse canal são obtidos com o FEPLL e o equalizador cego usando o algoritmo de Benveniste-Goursat. Entretanto, em relação à complexidade computacional e ao número de iterações necessárias à convergência, o FEPLL apresenta o melhor desempenho entre todas as técnicas simuladas para o canal considerado.

Convém dizer que, após a convergência, cada um dos algoritmos poderia passar do modo cego para o modo de decisão direta. Por exemplo, no modo de decisão direta para o FEPLL teríamos o seguinte algoritmo:

$$\begin{aligned}
 e[k] &= x[k] - \mathbf{f}^H[k-1]\mathbf{x}[k-1] - \mathbf{g}^H[k-1]\mathbf{x}^*[k-1], \\
 e_{DD}[k] &= \Psi(e[k]) - e[k], \\
 \begin{bmatrix} \mathbf{f}[k] \\ \mathbf{g}[k] \end{bmatrix} &= \begin{bmatrix} \mathbf{f}[k-1] \\ \mathbf{g}[k-1] \end{bmatrix} + \mu_{DD} e_{DD}[k] \begin{bmatrix} \mathbf{x}^*[k-1] \\ \mathbf{x}[k-1] \end{bmatrix},
 \end{aligned} \tag{4.87}$$

sendo  $e_{DD}[k]$  o erro de decisão direta,  $\Psi(\cdot)$  é um dispositivo de decisão e  $\mu_{DD}$  é o passo de adaptação no modo de decisão direta. Esse passo de adaptação pode ser bem maior (10 vezes ou mais) que o passo  $\mu$  do modo autodidata.

Com esse chaveamento, o erro residual se tornaria bem menor do que o indicado na Figura 4.14, ver apêndice C. Em geral, o critério para passar de um modo para o outro é feito tendo como parâmetro a comparação entre o valor do módulo do erro de decisão direta e um certo valor *limiar*, como, por exemplo, o valor 0,71.

TABELA 4.3. PARÂMETROS DOS ALGORITMOS CEGOS E RESULTADOS DE SIMULAÇÃO

Canal $\mathbf{h}_m$ (fase mínima)					
Algoritmo:	FEPL	(CMA)	Sato	Benveniste-Goursat	FEPLL
Número de iterações para abrir o olho	~ 10000	~ 3100	~ 2700	~ 2300	~ 2000
Passo $\mu$	$3,8 \cdot 10^{-4}$	$6,2 \cdot 10^{-5}$	$4,5 \cdot 10^{-4}$	$5,0 \cdot 10^{-4}$	$1,2 \cdot 10^{-4}$
Número de coeficientes do equalizador	37	25	25	25	5
MSE após a convergência (dB)	-7,00	-13,36	-13,93	-15,09	-15,11

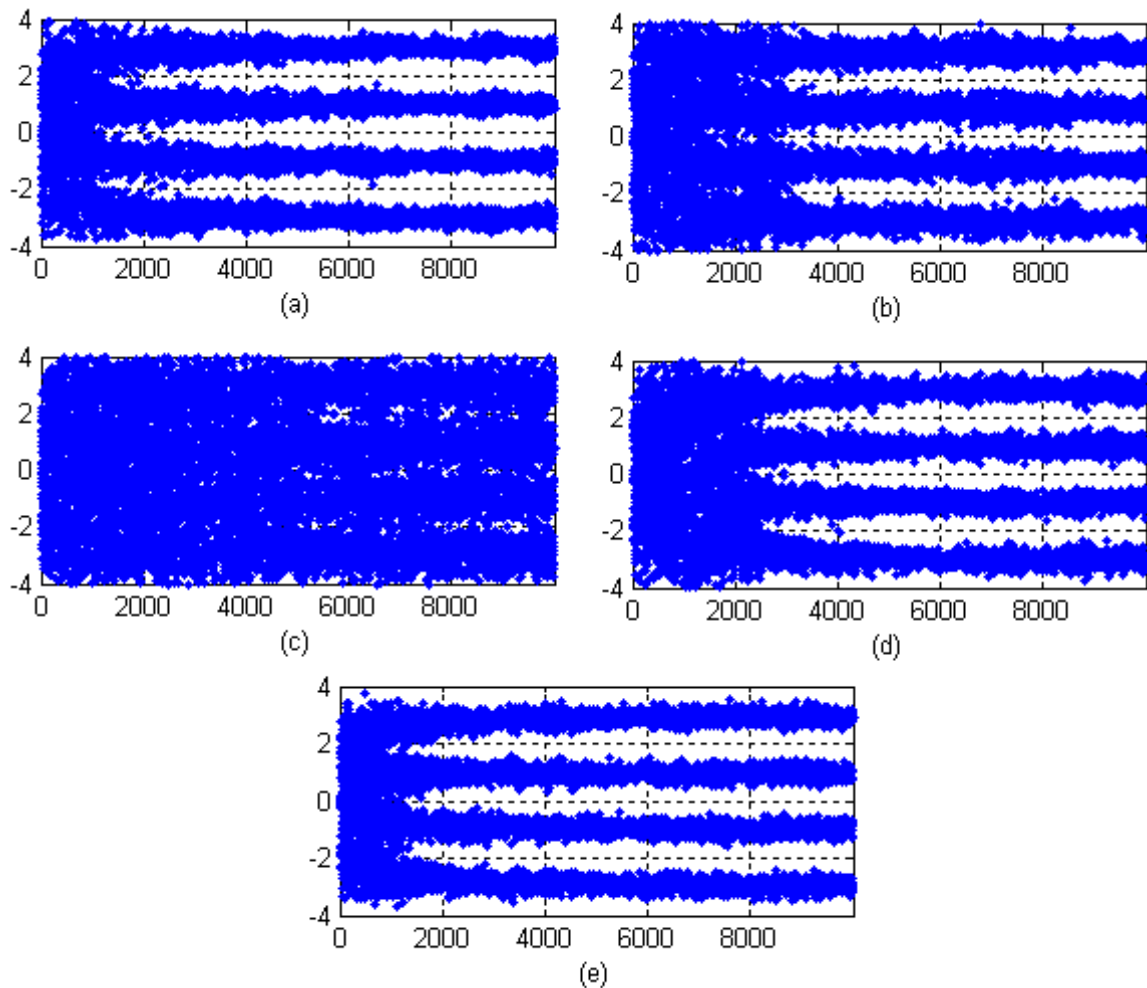


Figura 4.13. Diagrama de olho de uma realização para as diversas técnicas cegas para o canal  $\mathbf{h}_m$  e uma relação sinal-ruído de 30 dB: (a) FEPLL, (b) CMA, (c) FEPL, (d) Sato, (e) Benveniste-Goursat.

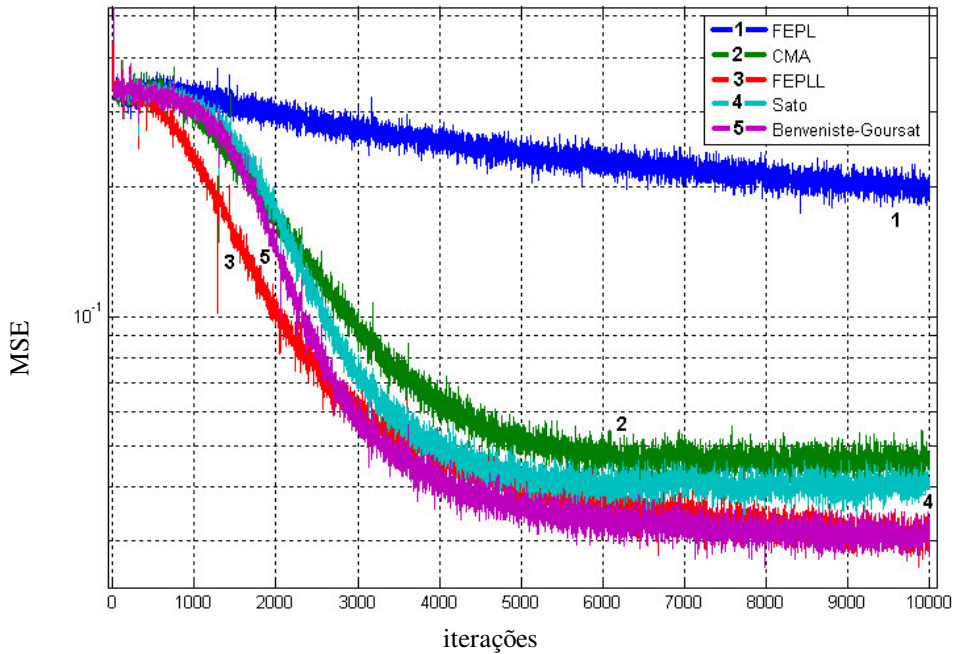


Figura 4.14: Comparação dos algoritmos cegos: MSE do erro de decisão direta. Relação sinal-ruído de 30 dB.

### 4.8.3 FEPLL adaptativo com passo variável

*Exemplo 1.* Inicialmente, consideramos o canal  $\mathbf{h}_1 = [0,5832+0,2187j; 0,0073+0,3353j; 0,2781-0,5509j; 0,1768-0,2959j]$ . Este canal é de fase não-mínima, apresenta um nulo espectral e um zero real dentro do círculo de raio unitário. Como podemos observar pela Figura 4.15, ele é corretamente equalizado pelo FEPLL após cerca de 600 iterações apenas. Os parâmetros usados estão listados na Tabela 4.4. Esses parâmetros foram estabelecidos por meio de várias simulações. Neste exemplo consideramos uma SNR de 30 dB. Uma observação importante é que o equalizador apresenta apenas 3 coeficientes em cada filtro transversal  $\mathbf{f}$  e  $\mathbf{g}$ . Além disso, o desempenho seria semelhante se fossem usados 4 ou até mesmo 5 coeficientes. Isto mostra que o FEPLL pode trabalhar bem mesmo quando ocorre um erro na estimativa do comprimento do canal. A próxima simulação reforça este ponto.

TABELA 4.4: PARÂMETROS DO FEPLL ADAPTATIVO PARA O CANAL  $\mathbf{h}_1$ .

$\mu :$	$\mu_{min} = 9 \times 10^{-5}$ $\mu_{max} = 1,15 \times 10^{-3}$	$\beta_{\mu} = 5 \times 10^{-6}$
$\mu_G :$	$\mu_{min} = 2,25 \times 10^{-4}$ $\mu_{max} = 1,8 \times 10^{-3}$	$\beta_G = 1 \times 10^{-5}$
$\mu_{\Phi} :$	$\mu_{min} = 5 \times 10^{-5}$ $\mu_{max} = 9 \times 10^{-4}$	$\beta_{\Phi} = 8 \times 10^{-5}$
$G_0 = 1, \Phi_0 = 0$		$\alpha = 0,98 ; 3$ coeficientes

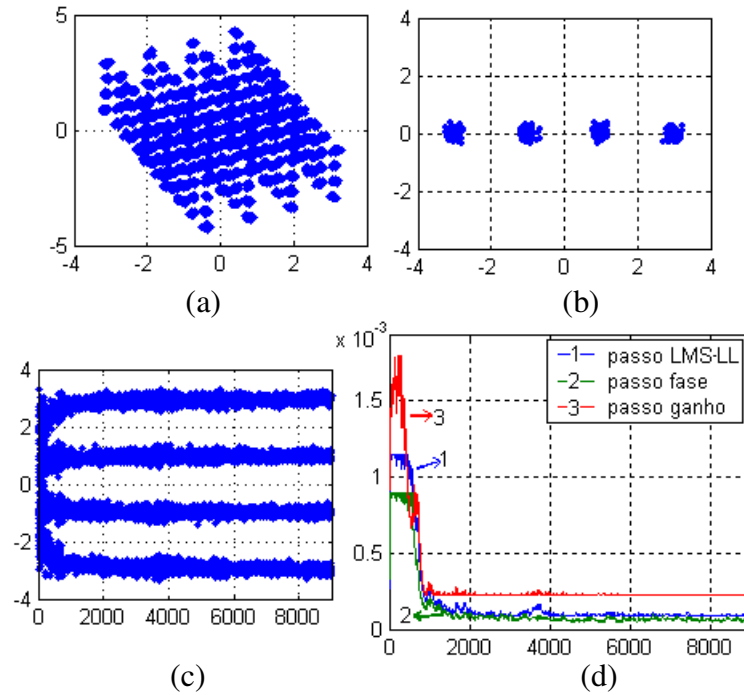


Figura 4.15. Equalização do canal  $\mathbf{h}_1$  com FEPLL de passo variável: (a) constelação após o canal, (b) sinal após o equalizador (últimas 500 amostras), (c) evolução da constelação na saída do equalizador, (d) passos de adaptação em função no tempo.

*Exemplo 2.* Iremos considerar agora o canal  $\mathbf{h}_2 = [0,5218-0,1648j; 0,2197+0,2746j; 0,1373; 0,1098-0,6591j; 0,0549-0,3295j]$ . Este canal é de fase não-mínima e apresenta um nulo espectral. Como podemos observar pela Figura 4.16, este canal também é corretamente equalizado pelo FEPLL após cerca de 1300 iterações. Os parâmetros usados estão listados na Tabela 4.5. Novamente, a relação sinal-ruído considerada foi de 30 dB.

TABELA 4.5: PARÂMETROS DO FEPLL ADAPTATIVO PARA O CANAL  $\mathbf{h}_2$ .

$\mu :$	$\mu_{min} = 7,00 \times 10^{-5}$ $\mu_{max} = 1,05 \times 10^{-3}$	$\beta_{\mu} = 5 \times 10^{-6}$
$\mu_G :$	$\mu_{min} = 1,33 \times 10^{-4}$ $\mu_{max} = 8,0 \times 10^{-4}$	$\beta_G = 1 \times 10^{-5}$
$\mu_{\Phi} :$	$\mu_{min} = 4,0 \times 10^{-5}$ $\mu_{max} = 4,8 \times 10^{-4}$	$\beta_{\Phi} = 8 \times 10^{-5}$
$G_0 = 1,5, \Phi_0 = 0$		$\alpha = 0,98 ; 6$ coeficientes

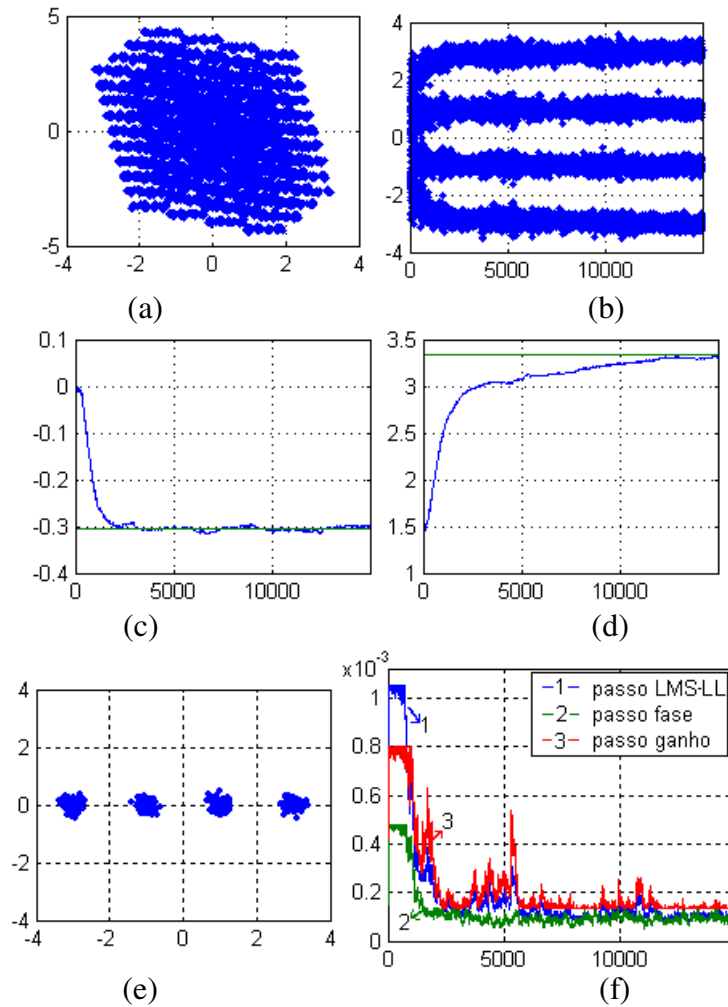


Figura 4.16. Equalização do canal  $\mathbf{h}_2$ : (a) constelação após o canal, (b) evolução do sinal na saída do FEPLL, (c) ajuste de fase em função do tempo, (d) ajuste de ganho, (e) últimos 500 símbolos equalizados, (f) evolução dos passos de adaptação.

*Exemplo 3.* Para mostrarmos a capacidade de convergência do algoritmo no caso de uma mudança brusca do canal, iremos considerar novamente os canais  $\mathbf{h}_1$  e  $\mathbf{h}_2$ . Inicialmente o FEPLL compensa o canal  $\mathbf{h}_1$  quando, na amostra 5000, o canal muda bruscamente para  $\mathbf{h}_2$ . Nesta simulação, usamos valores intermediários entre os mostrados nas Tabelas 4.4 e 4.5 para os passos de adaptação. O FEPLL usado nesta simulação tem 5 coeficientes. Podemos observar pela Figura 4.17 que o FEPLL com o algoritmo de passo variável conseguiu se recuperar rapidamente após a mudança do canal. A IES residual (definida em (1.11)) na saída do FEPLL é apresentada na Figura 4.18 (média de 15 realizações independentes).



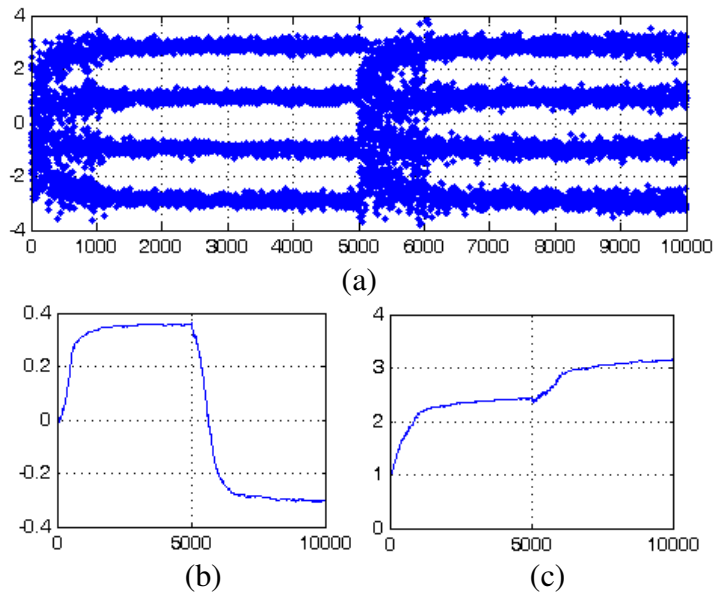


Figura 4.17. Diagramas de olho para o caso de variação brusca do canal, (b) evolução da fase, (c) evolução da correção do ganho em função do número de símbolos recebidos.

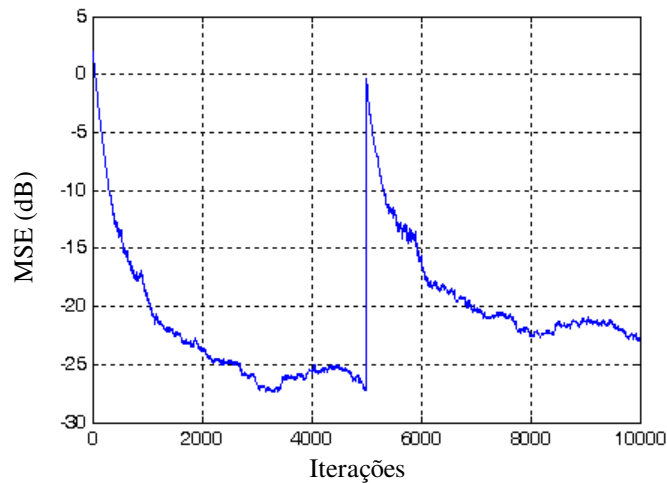


Figura 4.18. IES residual para os canais  $\mathbf{h}_1$  e  $\mathbf{h}_2$ , mudança brusca de  $\mathbf{h}_1$  para  $\mathbf{h}_2$  na amostra de número 5000. Relação sinal-ruído de 30 dB.

*Exemplo 4.* Finalmente, usaremos um canal de fase mínima  $\mathbf{h}_3 = [0,7553-0,1888j; 0,3588+0,1700j; 0,1907-0,3857j; 0,0328-0,2235j]$ , para compararmos o FEPLL usando o algoritmo de LMS de passo fixo, o FEPLL com o algoritmo de passo variável proposto e um equalizador cego usando o algoritmo de equalização de Godard (1980), também conhecido como CMA (*Constant Modulus Algorithm*), ver Apêndice C. Nesta simulação, o equalizador de Godard tem 21 coeficientes, enquanto que os filtros de erro de predição de passo fixo e de passo variável têm apenas 6 coeficientes ao todo. A Tabela 4.6 apresenta os demais parâmetros adotados. A Figura 4.19 apresenta a mediação de 200 realizações independentes

do erro médio quadrático de decisão direta e o diagrama de olho de uma realização para estes equalizadores. Como podemos observar, o FEPLL com passo variável apresenta o melhor desempenho, conseguindo abrir o olho com cerca de 400 amostras, enquanto que o FEPLL com passo fixo precisa de aproximadamente 1000 amostras e o equalizador com o algoritmo de Godard com cerca de 10.000 amostras. Convém observar que em (Lin, 2002) e (Xiong *et al.*, 2004) são propostas modificações no algoritmo CMA original tornando-o também de passo variável e, ao mesmo tempo, mais robusto que a versão simulada aqui.

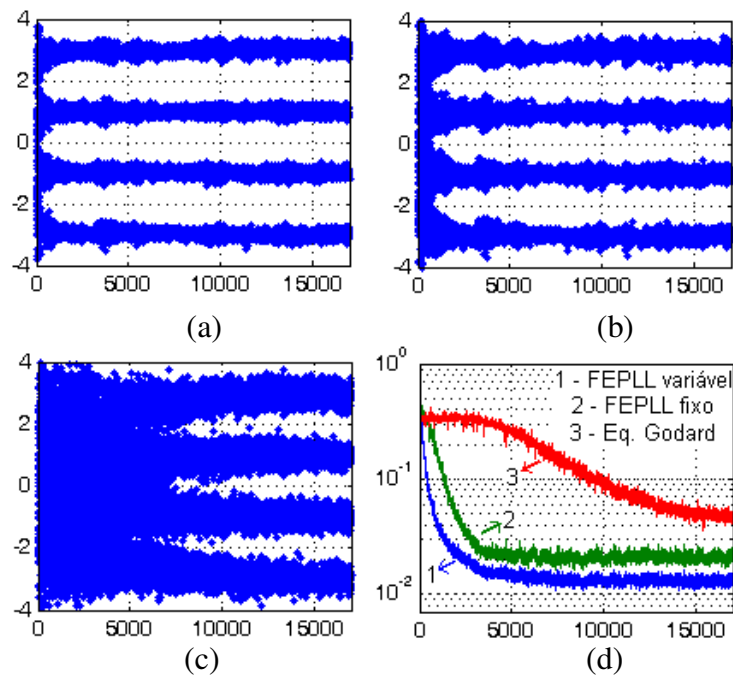


Figura 4.19. Diagramas de olho: (a) FEPLL de passo variável, (b) FEPLL de passo fixo, (c) equalizador de Godard, (d) valor médio do erro de decisão quadrático direto.

TABELA 4.6. PARÂMETROS DOS EQUALIZADORES PARA O CANAL  $h_3$ .

FEPLL com passo variável	$\mu$ :	$\mu_{min} = 7,00 \times 10^{-5}$ $\mu_{max} = 1,05 \times 10^{-3}$	$\beta_{\mu} = 5 \times 10^{-6}$
	$\mu_G$ :	$\mu_{min} = 1,33 \times 10^{-4}$ $\mu_{max} = 8,0 \times 10^{-4}$	$\beta_G = 1 \times 10^{-5}$
	$\mu_{\Phi}$ :	$\mu_{min} = 4,0 \times 10^{-5}$ $\mu_{max} = 4,8 \times 10^{-4}$	$\beta_{\Phi} = 8 \times 10^{-5}$
FEPLL com passo fixo:	$\mu = 2 \times 10^{-4}, \mu_G = 6 \times 10^{-4},$ $\mu_{\Phi} = 1 \times 10^{-3}$		
Equalizador de Godard: (CMA)	$\mu = 2 \times 10^{-5}, \mu_{\Phi} = 2,4 \times 10^{-5}$		

#### 4.8.4 FEPLL usando o algoritmo RLS

*Equalização com o algoritmo RLS.* Usaremos mais uma vez o canal definido por Porat e Friedlander (1991) e uma relação sinal ruído de 40 dB. Realizando o algoritmo indicado na Tabela 4.2 e uma constelação 4-PAM, obtemos os resultados indicados na Figura 4.20. Nesta figura vemos a constelação após o canal, após o equalizador (últimos 500 símbolos transmitidos) e a evolução do ganho, da fase e do diagrama de olho. Observamos que o olho abre com cerca de 150 iterações, que é um tempo significativamente menor em relação à equalização deste mesmo canal usando o algoritmo LMS. Podemos observar também que o olho está mais aberto nesta simulação. Os parâmetros usados foram:  $\delta = 2,5$ ;  $\lambda = 0,9995$ ;  $\mu_\phi = 0,0012$ ; e  $\mu_G = 0,003$ . O ganho  $G[0]$  foi inicializado em 3,0 e fase inicial foi considerada nula,  $\phi[0] = 0$ .

A Figura 4.21 mostra a equalização do canal definido por  $\mathbf{h}_c = [0,5408+0,0854j; 0,0882-0,2078j; -0,2300-0,6848j; -0,2676-0,2362j]$  e uma relação sinal-ruído de 40 dB pelo FEPLL usando o algoritmo RLS. Este canal apresenta um nulo espectral. Podemos observar que olho já está aberto com cerca de 120 iterações apenas. Os parâmetros do algoritmo RLS foram:  $\delta = 0,5$ ;  $\lambda = 0,9995$ ;  $\mu_\phi = 0,002$ ; e  $\mu_G = 0,005$ . O ganho  $G[0]$  foi inicializado em 3,0 e fase inicial foi considerada nula,  $\phi[0] = 0$ .

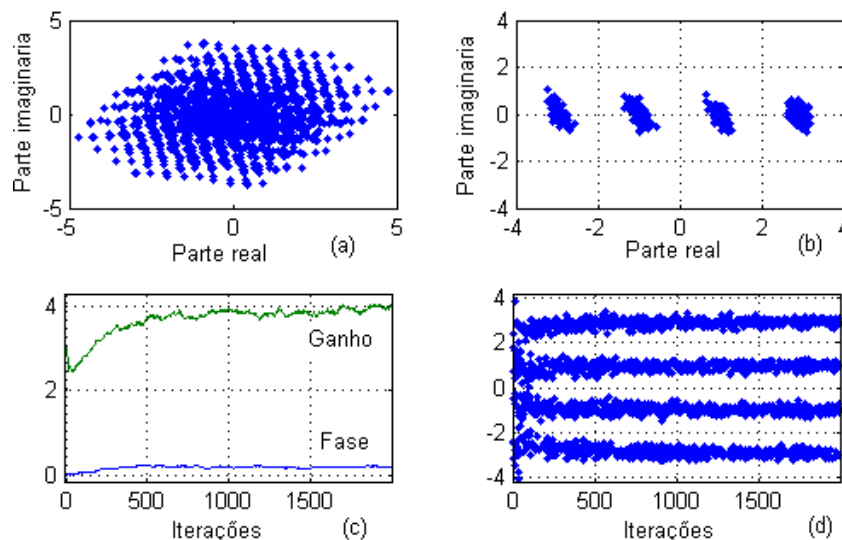


Figura 4.20. Equalização usando o algoritmo RLS, canal de Porat: (a) constelação após o canal, (b) sinal equalizado, (c) ganho e fase, (d) evolução da constelação após o equalizador.

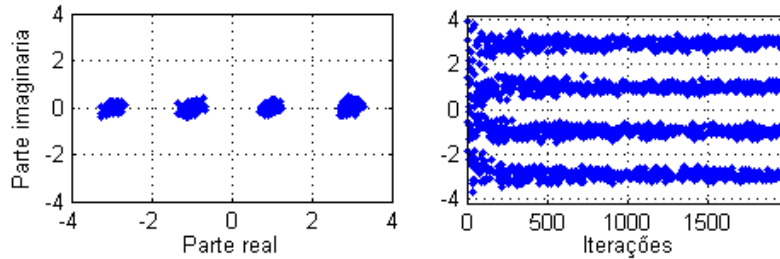


Figura 4.21. Equalização usando o algoritmo RLS, canal com nulo espectral.

#### 4.8.5 Equalização com atraso usando um conjunto de FEPLL's

Os canais e os parâmetros ( $\mu_{LL}$ ,  $\mu_{\Phi}$ ,  $\mu_G$  e  $\mu_i$ ) dos algoritmos usados nas simulações a seguir estão indicados na Tabela 4.7. Estes canais são complexos, de fase não-mínima e apresentam um ou mais zeros próximos ao círculo de raio unitário.

O banco de FEPLL's simulado usa apenas 3 FEPLL's (atrasos de 0, 1 e 2 amostras) e um equalizador LL treinado com a saída ótima dos FEPLL's, sendo esta, após o alinhamento dos símbolos, uma versão atrasada de 4 amostras em relação ao sinal original. Tanto os FEPLL's quanto o equalizador LL têm 12 coeficientes ( $M = 6$ ). Para efeitos comparativos, foi realizada a equalização treinada dos mesmos canais por um equalizador LL, também com um atraso de equalização de 4 amostras. É sabido que este atraso pode influenciar no desempenho do sistema (Gerstacker *et al.*, 2003; Endres *et al.*, 1996), mas para uma equalização com um equalizador largamente linear, da mesma forma que uma equalização usando super-amostragem (sistemas polifásicos), um atraso de equalização igual a  $(L+M)/2$  é razoável (Gesbert *et al.*, 1999).

A Figura 4.22 mostra o efeito do atraso de equalização para o Canal 1 (ver também Tabela 4.7) na variância do erro na saída do equalizador treinado, definida como

$$V_{erro} = 10 \log_{10} \left( E \left\{ \left| a[k] - y_i[k] \right|^2 \right\} \right), \quad (4.87)$$

na saída de um equalizador LL treinado de 12 coeficientes considerando-se relações sinal-ruído de 10, 20 e 30 dB. Para os demais canais simulados, o resultado é bastante similar. A partir da Figura 4.22, é fácil verificar que o atraso de equalização poderia ficar entre 3 e 7 amostras, sendo 4 amostras o valor ótimo para o Canal 1.

A avaliação da taxa de erro de símbolo (TES) é feita após um período de *treinamento* realizado com as primeiras 7500 amostras ótimas geradas pelo banco de FEPLL's. O equalizador largamente linear trabalha com um algoritmo LMS de passo fixo, já que esta configuração é a mais simples e apresenta um desajuste final menor que sua versão de passo normalizado (Aquino *et al.*, 2006a). A TES para os três canais é mostrada na Figura 4.22. A equalização dos canais 1 e 2 apresenta um comportamento semelhante. Já o canal 3 é mais difícil de ser equalizado e somente quando a relação sinal-ruído é igual ou maior que 15 dB é que a TES se torna essencialmente igual ao caso treinado.

TABELA 4.7: COEFICIENTES DOS CANAIS 1, 2 E 3 PARA FEPLL COM ATRASO DE EQUALIZAÇÃO

<b>h</b>	Canal 1	Canal 2	Canal 3
$h[0]$	0,4133+ 0,4724j	0,6950 - 0,0695j	0,2783 - 0,2087
$h[1]$	0,2952+0,4133j	0,1286+ 0,2328j	0,3478 - 0,2783j
$h[2]$	0,3543 - 0,3897j	0,3708- 0,4688j	0,5218+ 0,3478j
$h[3]$	0,1181+ 0,1771j	0,1078- 0,2168j	-0,3478 - 0,2609j
$h[4]$	-0,1181+ 0,1063j	0,1136- 0,1119j	0,1739+ 0,2609j
<i>Passos de adaptação</i>	$\mu_{LL} = 0,0001$	$\mu_{LL} = 0,0001$	$\mu_{LL} = 0,0002$
	$\mu_{\Phi} = 0,001$	$\mu_{\Phi} = 0,001$	$\mu_{\Phi} = 0,001$
	$\mu_G = 0,01$	$\mu_G = 0,01$	$\mu_G = 0,01$
	$\mu_i = 0,001$	$\mu_i = 0,001$	$\mu_i = 0,001$

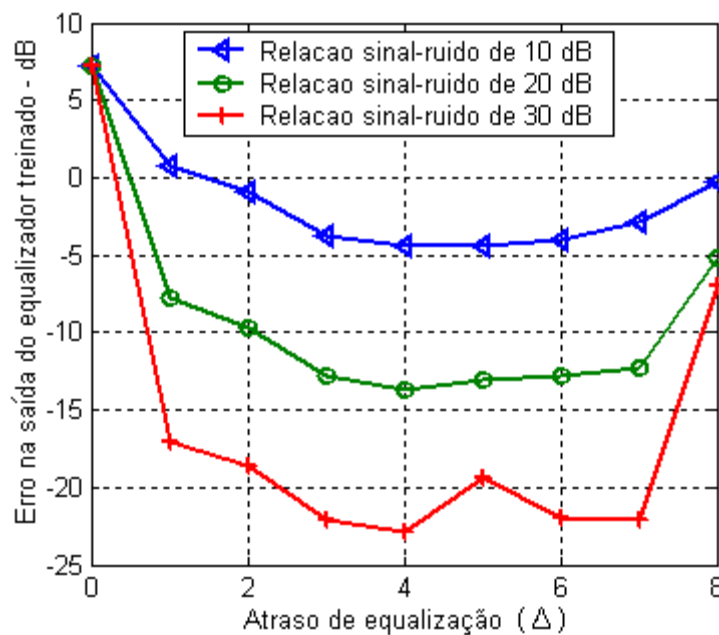


Figura 4.22. Variância do erro (em dB) na saída do equalizador LL treinado em função do atraso de equalização para o Canal 1.

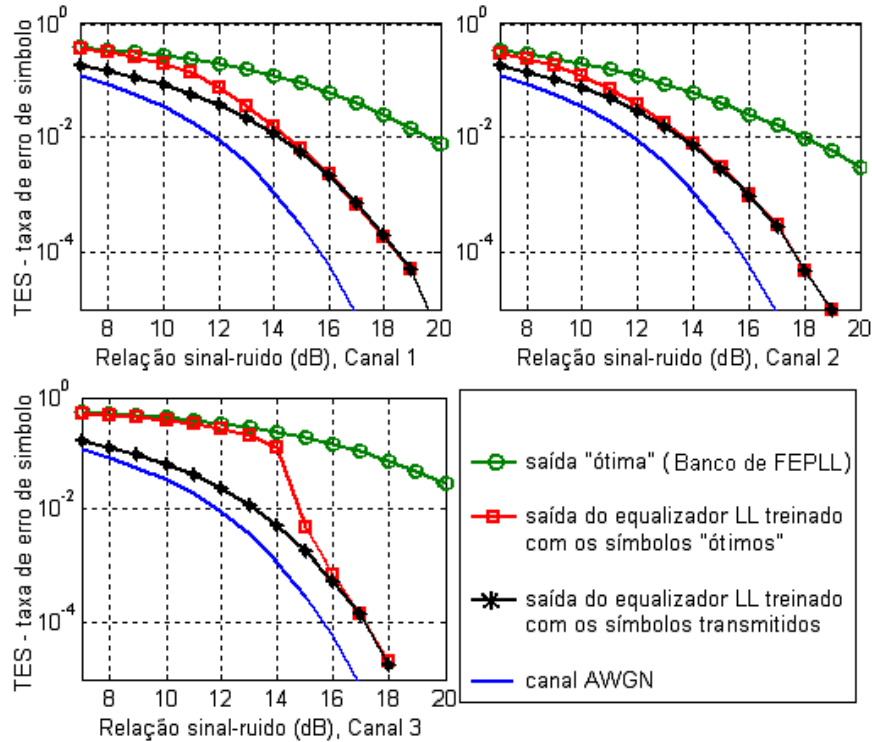


Figura 4.23. Taxa de erro de símbolo (TES) para os canais simulados.

## 4.9 Conclusão do capítulo

Neste capítulo apresentamos o filtro de erro de predição linear e a sua versão largamente linear. Mostramos que o filtro de predição LL é capaz de compensar canais de fase não-mínima, inclusive aqueles com nulos espectrais, desde que o canal não apresente zeros reais ou zeros complexos conjugados (canais  $\mathbf{h}$  e  $\mathbf{h}^*$  primos entre si). Esta característica torna o filtro de erro de predição LL interessante na aplicação do problema de equalização cega. Mostramos também que o número de coeficientes de um filtro de predição LL precisa ser apenas igual à ordem do canal, o que é uma grande vantagem computacional quando comparado a outras técnicas de equalização cega.

Nesse capítulo também aplicamos algoritmos LMS de passo fixo e de passo variável e o algoritmo RLS ao FEPLL. Desenvolvemos também um equalizador robusto com atraso de equalização tendo por base o FEPLL. Finalmente, mostramos uma implementação em FPGA do FEPLL. Diversos resultados numéricos mostraram o bom desempenho destes filtros como equalizadores adaptativos cegos.

# Capítulo 5

---

## Conclusões e Trabalhos Futuros

Neste capítulo final, apresentamos nossas conclusões, destacamos as nossas principais contribuições e, baseados nos resultados alcançados nesta tese e na literatura técnica consultada, propomos várias linhas de pesquisas para a continuidade do trabalho.

Este capítulo é estruturado como segue. Na Seção 5.1, apresentamos uma breve revisão dos capítulos que compõem esta tese, focando principalmente nas nossas contribuições originais. Em seguida, apresentamos idéias para a continuidade deste trabalho de pesquisa, incluindo alguma argumentação para dar suporte a tais idéias.

### 5.1 Contribuições desta tese

De forma bem compacta, podemos dizer que neste documento apresentamos a fundamentação do processamento largamente linear e a sua aplicação ao problema de equalização de canais com resposta impulsiva complexa usando equalizadores largamente lineares. Também apresentamos o desenvolvimento de um novo equalizador cego baseado em um filtro de erro de predição largamente linear. Este equalizador apresenta um desempenho superior ao de outras técnicas cegas lineares com uma carga computacional menor. Os equalizadores estudados ou propostos, usando o algoritmo LMS-LL de passo fixo, foram também implementados em FPGA. A seguir, detalhamos o conteúdo e contribuições de cada um dos capítulos desta tese.

Na introdução à tese, Capítulo 1, apresentamos, em linhas gerais, o processamento largamente linear, explicando as suas principais vantagens em relação ao processamento estritamente linear, algumas das aplicações encontradas na literatura e os nossos objetivos específicos com essa pesquisa. Nesse capítulo também introduzimos a notação matemática usada ao longo da tese e alguns conceitos essenciais para a leitura deste documento. Também listamos as nossas contribuições mais relevantes, em forma de artigos, durante o período de doutorado.

No Capítulo 2, apresentamos os principais conceitos envolvidos no processamento largamente linear, tais como o conceito de circularidade e de pseudo-covariância. Incluímos uma descrição do esquema de codificação espaço-temporal de bloco (STBC) proposto por Alamouti e a sua aplicação ao processamento largamente linear para sistemas MIMO. Tal descrição é encarada como uma contribuição uma vez que não existia tal fundamentação em português e a mesma se encontra dispersa na literatura.

Já no Capítulo 3, apresentamos um estudo detalhado do equalizador largamente linear e a sua aplicação ao problema de equalização. Nesse capítulo discutimos sobre a ordem mínima e as condições necessárias para um equalizador LL poder inverter perfeitamente um canal FIR. Apresentamos também um estudo sobre o comportamento estatístico do algoritmo LMS LL e a aplicação do processamento LL ao equalizador *multi-split* complexo. Ainda no Capítulo 3, incluímos uma seção mostrando uma implementação do equalizador LL em FPGA.

No Capítulo 4, o nosso foco foi o problema de equalização cega, tendo como uma condição implícita manter uma baixa complexidade computacional do equalizador. Para isto, propusemos um novo equalizador cego baseado em um filtro de erro de predição largamente linear (FEPLL). Realizamos um estudo detalhado desse equalizador (análise polifásica, filtro ótimo) e expandimos a idéia básica em duas direções: FEPLL com passo variável e FEPLL com atraso de equalização. Da mesma forma que no Capítulo 3, implementamos o FEPLL, usando um algoritmo LMS de passo fixo, em FPGA. Nesse capítulo residem nossas principais contribuições originais.

Tanto no Capítulo 3 quanto no Capítulo 4, incluímos uma extensa seção de resultados de simulação com diversos exemplos ilustrando os resultados teóricos.

## 5.2 Continuidade do trabalho

Equalização de canal de comunicação é um tópico de pesquisa fascinante e o processamento largamente linear pode ser usado com uma ferramenta poderosa para a obtenção de bons resultados a um custo computacional baixo. Tendo em vista o trabalho por nós desenvolvido, os desafios que vislumbramos e os resultados já conhecidos na literatura técnica, podemos citar as seguintes propostas para a continuidade desta pesquisa.

- *Estudo comparativo entre o equalizador LL e um equalizador LL-DFE. A utilização de realimentação (DFE) pode certamente melhorar o desempenho de um*



equalizador LL, mas praticamente inexitem na literatura estudos sobre o quanto podemos obter de ganho e em quais condições isso ocorre.

- *Análise de convergência do algoritmo LMS para o FEPLL.* O estudo de convergência de um algoritmo LMS para o FEPLL, especialmente na sua versão de passo variável, parece ser um problema de análise estatística interessante e poderia, por exemplo, esclarecer quais seriam as melhores estratégias para os ajustes dos passos de adaptação.
- *Estudo e projeto de um equalizador cego híbrido usando um FEPLL e um equalizador usando um algoritmo CMA-LL.* Essa proposta surge da observação de que um FEPLL apresenta, em geral, uma convergência inicial mais rápida que um equalizador cego CMA. Entretanto, após a convergência o equalizador CMA tende a ter um menor MSE, logo, a combinação desses dois equalizadores em uma estrutura híbrida poderia ser uma solução interessante.
- *Combinação do processamento LL com superamostragem.* Da mesma forma que na equalização linear, a equalização largamente linear pode também se beneficiar da técnica de superamostragem, sendo possível alcançar resultados ainda melhores. Novamente, essa idéia é pouco explorada na literatura.
- *Estudo do processamento LL em formatação de feixe.* Já o problema de formatação de feixe usando processamento largamente linear recebeu, até agora, pouca atenção, mas apresenta resultados interessantes.
- *Exploração do processamento LL em equalização cega de canais MIMO ou em sistemas OFDM/OQAM.* O futuro parece apontar para uma combinação MIMO/OFDM nos sistemas de comunicação. Sistemas OFDM/OQAM necessitam de equalizadores já que não usam portadoras pilotos, sendo por isso possível à utilização do processamento largamente linear. Também já é conhecido que em sistemas MIMO o processamento largamente linear pode aumentar o desempenho, portanto, sistemas MIMO/OFDM podem ser beneficiados com o uso do processamento largamente linear.

## Apêndice A – Complementos do Capítulo 2

Neste apêndice apresentamos alguns conceitos essenciais da teoria da probabilidade que são necessários à compreensão de parte dessa tese, principalmente o Capítulo 2. O enfoque é sobre variáveis aleatórias discretas. Na última seção deste apêndice, apresentamos um resumo do artigo de Schreier *et al.* (2003). Naquele artigo, os autores apresentam uma análise sobre o ganho do processamento LL sobre o processamento linear para os casos de estimação e detecção de sinal.

### A.1 Definições iniciais

- *Espaço de amostras*: conjunto  $\Omega$  de todos os pontos amostra que uma variável aleatória pode assumir. Um exemplo clássico de um espaço de amostras é o conjunto  $\Omega = \{1, 2, 3, 4, 5, 6\}$  que representa todos os possíveis resultados do lançamento de um dado não viciado.
- *Evento*: é a ocorrência de um elemento, um subconjunto, ou um conjunto de subconjuntos de  $\Omega$ . As operações com eventos obedecem às mesmas regras das operações com conjuntos. Exemplo: ocorrência de um número maior que três no lançamento de um dado gera o conjunto  $A = \{4, 5, 6\}$ .
- *Evento nulo*: não contém pontos amostra (conjunto vazio),  $A = \Phi$ .
- *Evento elementar*: consiste de um único ponto amostra.
- Um conjunto de eventos  $F$  é uma  $\sigma$ -álgebra  $F$  quando satisfaz as seguintes condições:
  - a. Se  $A \in F$ ,  $\bar{A} \in F$ , sendo  $\bar{A}$  o complemento do evento  $A$ .
  - b. Se  $A \in F$ ,  $B \in F$ , então  $A \cup B \in F$ . Além disso, se  $A_i \in F$ ,  $1 \leq i \leq \infty$ , então  $\bigcup_{i=1}^{\infty} A_i \in F$ .
- A cada *evento*  $A \in F$  está associada uma probabilidade  $P$  para a sua ocorrência. Define-se  $P$  através de três axiomas:
  - a.  $P(A) \geq 0$ ,  $\forall A \in F$ .
  - b.  $P(\Omega) \geq 0$ .
  - c. Se  $A_i \in F$ ,  $1 \leq i \leq \infty$ , são eventos disjuntos ( $A_i \cap A_j = \emptyset$ , para  $i \neq j$ ), então 
$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i).$$

O conjunto de eventos  $\Omega$ , a álgebra  $F$  e a probabilidade  $P$  formam a tripla  $(\Omega, F, P)$  do espaço probabilístico da teoria axiomática de Kolmogorov.

Neste ponto podemos fazer a seguinte observação: qualquer medida  $P$  satisfazendo os axiomas acima é matematicamente aceitável, entretanto, para que ela seja fisicamente aceitável e satisfatória no mundo real, a medida de probabilidade deve representar o “comportamento médio” do evento. Um “comportamento médio” de interesse nos leva ao conceito de frequência relativa de ocorrência de um evento.

- *Frequência relativa.* Seja um experimento que tenha sido repetido  $N$  vezes, resultando na ocorrência de um dado evento  $n(A)$  vezes. A razão  $n(A)/N$  é denominada frequência relativa de ocorrência de  $A$  para  $N$  realizações do experimento. Observa-se que, quando  $N$  cresce, a frequência relativa tende a se estabilizar em torno de um determinado valor. Uma medida atraente para medida de probabilidade  $P(A)$  é:

$$P(A) = \lim_{N \rightarrow \infty} \frac{n(A)}{N}. \quad (\text{A.1})$$

Entretanto, a expressão (A.1) não é matematicamente aceitável como definição da probabilidade de ocorrência do evento  $A$ , pois  $N$  sempre será um número finito e  $P(A)$  nunca será calculada exatamente. Apesar disso, para valores suficientemente elevados de  $N$ , a frequência relativa pode ser usada como uma boa aproximação de  $P(A)$ . As seguintes propriedades são conseqüências dos axiomas:

- a. A probabilidade de um evento vazio é zero, isto é,  $P(\emptyset) = 0$ .
  - b. A probabilidade do evento complementar é  $P(\bar{A}) = 1 - P(A)$ .
  - c. Para quaisquer eventos  $A$  e  $B$ ,  $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ .
- *Probabilidade Condicional.* A probabilidade condicional de um dado evento  $B$  dado que o evento  $A$ , com  $P(A) > 0$ , ocorreu é definida por:

$$P(B|A) = \frac{P(A \cap B)}{P(A)}. \quad (\text{A.2})$$

## A.2 Variável aleatória discreta

Uma variável aleatória é discreta quando  $\Omega_X = \{X_1, X_2, \dots\}$  é um conjunto contável. Cada valor  $X_i$  corresponde a um evento no espaço amostral  $\Omega_X$ . A cada evento  $X_i$  da variável aleatória podemos associar uma medida de probabilidade:

$$p_i \triangleq P(x = X_i) = P(\{\omega \in \Omega_X \mid x(\omega) = X_i\}), \quad i = 1, 2, 3, \dots \quad (\text{A.3})$$

A condição  $\sum_i p_i = 1$  deve ser satisfeita. O conjunto de probabilidades  $\{p_1, p_2, \dots\}$  é denominado de distribuição de probabilidade. Para um conjunto de números reais  $\zeta \subset \mathbb{R}$ , a probabilidade do evento  $x \in \zeta$  e dada por

$$P(x \in \zeta) = P(\{\omega \in \Omega_X \mid x(\omega) \in \zeta\}) = \sum_{i: X_i \in \zeta} p_i. \quad (\text{A.4})$$

O valor médio ou esperado de uma variável aleatória discreta é dado por:

$$m_x = E\{X\} = \sum_k X_k p_k. \quad (\text{A.5})$$

A variância de uma v.a. discreta é definida por:

$$\begin{aligned} \sigma_x^2 &= E\{(X - m_x)^2\} \\ &= E\{X^2\} - 2E\{Xm_x\} + E\{m_x^2\} \\ &= E\{X^2\} - 2E\{X\}m_x + m_x^2 \\ &= E\{X^2\} - m_x^2, \end{aligned} \quad (\text{A.6})$$

sendo  $E\{X^2\} = \sum_k X_k^2 p_k$  o valor quadrático médio da v.a.

## A.3 Covariância e pseudo-covariância

A covariância entre duas variáveis aleatórias complexas  $z$  e  $w$  é definida por:

$$r_{zw} \triangleq E\{(z - m_z)(w - m_w)^*\}, \quad (\text{A.8})$$

sendo  $m_z$  e  $m_w$  os valores médios de  $z$  e  $w$ . Note que o valor médio (valor esperado) de uma variável aleatória complexa  $z = x + jy$  é calculado por:  $m_z = E\{x\} + jE\{y\}$ . A pseudo-covariância é definida como:

$$c_{zw} \triangleq E\{(z - m_z)(w - m_w)\}. \quad (\text{A.9})$$

Seja uma variável aleatória complexa  $z = x + jy$ , com  $x$  e  $y$  independentes. O valor esperado de  $z^2$  é calculado como:

$$\begin{aligned} E\{zz\} &= E\{(x + jy)(x + jy)\} \\ &= E\{x^2\} + E\{2jxy\} + E\{-y^2\} \\ &= E\{x^2\} + 2jE\{x\}E\{y\} + E\{-y^2\}, \end{aligned} \quad (\text{A.10})$$

para  $x$  e  $y$  com médias nulas, teremos

$$E\{zz\} = E\{x^2\} - E\{y^2\}. \quad (\text{A.11})$$

Para  $x$  e  $y$  com mesma variância:  $E\{zz\} = 0$ .

Agora, ampliaremos os conceitos acima para o caso de vetores de variáveis aleatórias. Inicialmente, podemos definir a covariância entre dois vetores reais como:  $\text{Cov}\{\mathbf{u}, \mathbf{v}\} = E\{(\mathbf{u} - E\{\mathbf{u}\})(\mathbf{v} - E\{\mathbf{v}\})^T\}$ . Note que para o caso complexo precisaremos de quatro matrizes de covariância:  $\text{Cov}\{\mathbf{z}_R, \mathbf{w}_R\}$ ,  $\text{Cov}\{\mathbf{z}_I, \mathbf{w}_R\}$ ,  $\text{Cov}\{\mathbf{z}_R, \mathbf{w}_I\}$  e  $\text{Cov}\{\mathbf{z}_I, \mathbf{w}_I\}$ , onde  $\mathbf{z}_R = \text{Re}\{\mathbf{z}\}$ ,  $\mathbf{w}_R = \text{Re}\{\mathbf{w}\}$ ,  $\mathbf{z}_I = \text{Im}\{\mathbf{z}\}$ ,  $\mathbf{w}_I = \text{Im}\{\mathbf{w}\}$ . A matriz de covariância entre dois vetores aleatórios complexos pode ser definida como:

$$R_{zw} \triangleq E\{(\mathbf{z} - \mathbf{m}_z)(\mathbf{w} - \mathbf{m}_w)^H\}, \quad (\text{A.12})$$

onde  $\mathbf{z}$  e  $\mathbf{w}$  são vetores coluna com  $\mathbf{m}_z = E\{\mathbf{z}\}$  e  $\mathbf{m}_w = E\{\mathbf{w}\}$ . A matriz de pseudo-covariância é então definida como:

$$C_{zw} \triangleq E\{(\mathbf{z} - \mathbf{m}_z)(\mathbf{w} - \mathbf{m}_w)^T\}. \quad (\text{A.13})$$

A partir das definições acima, é fácil percebermos que

$$\begin{aligned}
Cov\{\mathbf{z}_R, \mathbf{w}_R\} &= \frac{1}{2} \Re\{R_{zw} + C_{zw}\}, \\
Cov\{\mathbf{z}_I, \mathbf{w}_I\} &= \frac{1}{2} \Re\{R_{zw} - C_{zw}\}, \\
Cov\{\mathbf{z}_I, \mathbf{w}_R\} &= \frac{1}{2} \Im\{R_{zw} + C_{zw}\}, \\
Cov\{\mathbf{z}_R, \mathbf{w}_I\} &= -\frac{1}{2} \Im\{R_{zw} - C_{zw}\}.
\end{aligned} \tag{A.14}$$

Como consequência, dois vetores complexos aleatórios são descorrelacionados somente se as suas matrizes de covariância e pseudo-covariância são nulas.

#### A.4 Comparação de desempenho entre o processamento linear e largamente linear para os casos de estimação e detecção

Apresentamos aqui um resumo do artigo de Schreier *et al.* (2003). Nesse artigo, os autores calculam o ganho do processamento largamente linear em relação ao linear para os casos de estimação e detecção. Por conveniência, reescrevemos nesta seção algumas expressões e definições já apresentadas no Capítulo 2 e também adaptamos parte da notação do artigo original para que ficasse consistente com a notação usada nesta tese.

Seja  $\mathbf{s}$  um vetor coluna de média nula cujos elementos são variáveis aleatórias complexas. A sua matriz de covariância é  $\mathbf{R} = E\{\mathbf{s}\mathbf{s}^H\}$  e a matriz de covariância complementar é calculada por  $\mathbf{C} = E\{\mathbf{s}\mathbf{s}^T\}$ . Podemos definir o vetor aumentado  $\mathbf{s}_{LL} = [\mathbf{s}^T \ \mathbf{s}^H]^T$ . A matriz de covariância aumentada de  $\mathbf{s}_{LL}$  é dada por:

$$\mathbf{\Gamma} = E\{\mathbf{s}_{LL}\mathbf{s}_{LL}^H\} = \begin{bmatrix} \mathbf{R} & \mathbf{C} \\ \mathbf{C}^* & \mathbf{R}^* \end{bmatrix}, \tag{A.15}$$

a matriz de covariância aumentada dá uma descrição de segunda ordem completa de  $\mathbf{s}$ . Suponha agora que nós queremos detectar um sinal real  $\mathbf{x}$  que é transmitido sobre um canal que o gira por alguma fase aleatória  $\phi$  e soma um ruído gaussiano branco complexo  $\boldsymbol{\eta}$ . As observações são então determinadas por  $\mathbf{r} = \mathbf{x}e^{j\phi} + \boldsymbol{\eta}$ , onde nós assumiremos independência mútua entre  $\mathbf{x}$ ,  $\boldsymbol{\eta}$  e  $\phi$ . Além disso, denote o sinal girado por  $\mathbf{s} = \mathbf{x}e^{j\phi}$ . Sua matriz de covariância é  $\mathbf{R} = E\{\mathbf{x}\mathbf{x}^T\}$  e sua matriz de covariância complementar é  $\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\} \cdot E\{e^{2j\phi}\}$ .

Existem dois casos especiais: quando a fase  $\phi$  é uniformemente distribuída entre 0 e  $2\pi$  e quando a fase é um valor conhecido. Na primeira hipótese,  $\mathbf{C} = \mathbf{0}$  e a detecção será não coerente. Na segunda hipótese,  $\mathbf{C} = e^{2j\phi}\mathbf{R}$  e a detecção será coerente. Para o caso de detecção coerente,  $\Re\{\mathbf{r}e^{-j\phi}\}$  tem uma estatística suficiente para este problema e não existe a necessidade de usar álgebra complexa. No caso de detecção não-coerente, a informação está inserida igualmente nas partes real e imaginária de  $\mathbf{r}$  e a detecção pode ser realizada usando o processamento estritamente linear. É bem conhecido que o processamento coerente traz um ganho de 3 dB sobre o processamento não-coerente.

A fase  $\phi$ , entretanto, pode não ser conhecida exatamente, ou podemos estar interessados em uma aproximação adaptativa da detecção coerente. Estes são casos onde as partes real e imaginária de  $\mathbf{r}$  precisam ser processadas independentemente, o que leva ao uso do processamento largamente linear.

#### A.4.1 Definições necessárias para a comparação

Como há uma grande variedade de aplicações, existem, obviamente, várias formas de avaliação e comparação de desempenhos. Muitas destas medidas de desempenho, porém, são da classe  $M(\{\lambda_n\}, \mathbf{p})$ ; isto é, elas são funções dos autovalores  $\{\lambda_n\}$  da matriz de covariância aumentada de  $\mathbf{s}$  e um vetor de parâmetros  $\mathbf{p}$ . Tipicamente,  $\mathbf{p}$  contém ao menos o nível de ruído  $N_0$ . Para essa classe particular de medidas de desempenho, há um resultado geral que compara o processamento LL com o estritamente linear, o qual apresentamos agora.

Seja  $\mathbf{\Gamma}$  uma matriz de covariância aumentada de  $\mathbf{s}$  com  $\mathbf{R}$  fixo e  $\mathbf{C}$  livre. O conjunto de todas as matrizes  $\mathbf{\Gamma}$  será chamado de  $\mathcal{F}_c$ . Desde que toda matriz  $\mathbf{\Gamma}$  deve ser definida semi-positiva, o conjunto  $\mathcal{F}_c$  é convexo e compacto. Assumindo que quanto maior  $M(\cdot)$ , melhor o desempenho, podemos expressar o ganho de desempenho máximo do processamento LL sobre o processamento linear como

$$\max_{\mathbf{p}} M_{LL}/M_L = \max_{\mathbf{p}} \left[ \max_{\mathbf{\Gamma} \in \mathcal{F}_c} M(\text{av}(\mathbf{\Gamma}), \mathbf{p}) / M(\text{av}(\mathbf{\Gamma}_0), \mathbf{p}) \right], \quad (\text{A.16})$$

onde  $\text{av}(\mathbf{\Gamma})$  denota os autovalores de  $\mathbf{\Gamma}$  e  $\text{av}(\mathbf{\Gamma}_0)$  os autovalores de  $\mathbf{\Gamma}$  com  $\mathbf{C} = \mathbf{0}$ . É fácil perceber que  $\text{av}(\mathbf{\Gamma}_0) = [\boldsymbol{\lambda}, \boldsymbol{\lambda}]$ , onde  $\boldsymbol{\lambda} = \text{av}(\mathbf{R})$ . A questão é: qual escolha de  $\mathbf{C}$  maximiza o

critério de desempenho  $M(\cdot)$ ? Para responder a essa questão precisamos de algumas definições e resultados intermediários.

**Definição 1 – Majoração.** Um vetor real  $\mathbf{x}$  com dimensão  $N \times 1$  é dito ser majorado por um vetor real  $\mathbf{y}$  ( $N \times 1$ ), escrito como  $\mathbf{x} \prec \mathbf{y}$ , se  $\sum_{k=1}^n x_{[k]} \leq \sum_{k=1}^n y_{[k]}$ ,  $n = 1, \dots, N$ , com a igualdade para  $n = N$ . O operador  $[\cdot]$  permuta  $\mathbf{x}$  tal que  $x_{[1]} \geq \dots \geq x_{[N]}$ .

Intuitivamente, se  $\mathbf{x} \prec \mathbf{y}$ , então os componentes de  $\mathbf{x}$  “menos espalhados” ou “mais iguais” que os componentes de  $\mathbf{y}$ .

Para uma dada matriz de covariância  $\mathbf{R}$ , o vetor aleatório cuja matriz de covariância aumentada tem um menor espalhamento de autovalores precisa ser próprio, isto é,  $\mathbf{C} = 0$ . Quanto mais impróprio um vetor se torna, mais espalhado se torna os autovalores de sua matriz aumentada. Finalmente, o vetor mais impróprio, novamente para uma determinada matriz  $\mathbf{R}$ , corresponde a um ponto extremo  $\Gamma_{\mathbf{x}}$  do conjunto convexo  $\Gamma_c$ . Todos os pontos extremos no conjunto  $\Gamma_c$  são caracterizados pelo complemento de Schur nulo:  $\mathbf{R}^* - \mathbf{C}^* \mathbf{R}^{-1} \mathbf{C} = \mathbf{0}$ . Em consequência, os seus autovalores formam o vetor  $[2\boldsymbol{\lambda}, \mathbf{0}_N]$ , sendo  $\mathbf{0}_N$  um vetor nulo de dimensão  $N \times 1$ . Usando a noção de majoração, podemos estabelecer o seguinte lema.

**Lema.** Dentro do conjunto  $\Gamma_c$ , temos o seguinte ordenamento parcial:

$$av(\Gamma_0) = [\boldsymbol{\lambda}, \boldsymbol{\lambda}] \prec av(\Gamma) \prec [2\boldsymbol{\lambda}, \mathbf{0}_N] = av(\Gamma_{\mathbf{x}}), \quad (\text{A.17})$$

onde  $\Gamma_{\mathbf{x}}$  é um ponto extremo qualquer do conjunto  $\Gamma_c$ . O leitor interessado pode conferir a demonstração deste lema no artigo original de Schreier *et al.* (2003).

**Definição 2.** Uma função real  $g$  definida sobre um conjunto  $D \subset \mathbb{R}^N$  é dita ser Schur convexa sobre  $D$  se  $\mathbf{x} \prec \mathbf{y}$  sobre  $D$  implica que  $g(\mathbf{x}) \leq g(\mathbf{y})$ . De forma similar, uma função é chamada de Schur côncava sobre  $D$  se  $\mathbf{x} \prec \mathbf{y}$  sobre  $D$  implica que  $g(\mathbf{x}) \geq g(\mathbf{y})$ .

**Teorema.** Seja  $\Gamma$  a matriz de covariância aumentada de um vetor a ser tratada por algum sistema  $S$  e seja  $M(av(\Gamma), \mathbf{p})$  uma medida de desempenho de  $S$ . Se  $M(\cdot)$  é uma função Schur convexa, sobre o conjunto de autovalores de  $\Gamma$ , com  $\mathbf{p}$  sendo um conjunto de parâmetros,



então o ganho de desempenho máximo do processamento LL sobre o processamento estritamente linear é, sob o critério  $M(\cdot)$ ,

$$\max M_{LL}/M_L = \max_{\mathbf{p}} \left[ M([\mathbf{2}\boldsymbol{\lambda}, \mathbf{0}_N], \mathbf{p}) / M([\boldsymbol{\lambda}, \boldsymbol{\lambda}], \mathbf{p}) \right]. \quad (\text{A.18})$$

Mantendo as condições acima, mas com  $M(av(\boldsymbol{\Gamma}), \mathbf{p})$  sendo uma função Schur côncava, o máximo ganho de desempenho é dado por

$$\max M_L/M_{LL} = \max_{\mathbf{p}} \left[ M([\boldsymbol{\lambda}, \boldsymbol{\lambda}], \mathbf{p}) / M([\mathbf{2}\boldsymbol{\lambda}, \mathbf{0}_N], \mathbf{p}) \right]. \quad (\text{A.19})$$

#### A.4.2 Ganho na estimação

Aplicaremos, agora, os conceitos desenvolvidos para o problema de estimação. Seja um vetor randômico  $\mathbf{s}$  de média nula com matriz de covariância aumentada  $\boldsymbol{\Gamma}$ , distorcido por um ruído aditivo AWGN (próprio)  $\mathbf{n}$  com matriz de covariância aumentada  $N_0\mathbf{I}$ . O sinal observado é dado por  $\mathbf{r} = \mathbf{s} + \mathbf{n}$ , podemos definir o vetor  $\mathbf{r}_{LL} = [\mathbf{r}^T \mathbf{r}^H]^T$ . A estimativa de  $\mathbf{s}$  usando o critério MSE nos leva à expressão

$$\tilde{\mathbf{s}}_{LL} = \boldsymbol{\Gamma}(\boldsymbol{\Gamma} + N_0\mathbf{I})^{-1} \mathbf{r}_{LL}. \quad (\text{A.20})$$

Note que  $\tilde{\mathbf{s}}_{LL}$  é uma estimativa largamente linear em  $\mathbf{r}$ , já que depende de  $\mathbf{r}$  e  $\mathbf{r}^*$ . Esta dependência pode ser explicitada pela aplicação do lema de inversão de matrizes a  $(\boldsymbol{\Gamma} + N_0\mathbf{I})^{-1}$ . O MSE correspondente é uma função de  $N_0$  e autovalores de  $\boldsymbol{\Gamma}$  (isto é,  $\lambda_n$ ) e é dado por

$$MSE = \frac{N_0}{2} \sum_{n=1}^{2N} \frac{\lambda_n}{\lambda_n + N_0}. \quad (\text{A.21})$$

Este MSE é uma função Schur côncava do conjunto  $\lambda_n$  com  $N_0$  como parâmetro, o teorema (ver seção anterior) é aplicável e o ganho de desempenho máximo do processamento LL sobre o linear é dado por:

$$\begin{aligned} & \max_{N_0} \left[ MSE([\boldsymbol{\lambda}, \boldsymbol{\lambda}], N_0) / MSE([\mathbf{2}\boldsymbol{\lambda}, \mathbf{0}_N], N_0) \right] = \\ & \max_{N_0} \left[ \sum_{n=1}^N \frac{2\lambda_n}{\lambda_n + N_0} / \sum_{n=1}^N \frac{2\lambda_n}{2\lambda_n + N_0} \right] \rightarrow 2, \quad N_0 \rightarrow 0. \end{aligned} \quad (\text{A.22})$$

Portanto, quando  $N_0 \rightarrow 0$ , o processamento LL pode conseguir a metade do MSE do processamento linear. Nesta aplicação, a vantagem da filtragem LL diminui quando o ruído aumenta e desaparece completamente quando  $N_0 \rightarrow \infty$ .

### A.4.3 Ganho na detecção

Analisaremos, agora, o problema de detecção. O problema de detecção que iremos resolver é um simples teste de hipótese:

$$\begin{cases} H_0 : \mathbf{r} = \mathbf{n} \\ H_1 : \mathbf{r} = \mathbf{s} + \mathbf{n}. \end{cases} \quad (\text{A.23})$$

Novamente, o ruído  $\mathbf{n}$  é AWGN complexo com matriz de covariância aumentada  $N_0\mathbf{I}$  e  $\mathbf{s}$  é um vetor randômico de média nula com matriz de covariância aumentada  $\mathbf{\Gamma}$ . O detector ótimo realiza uma comparação de limiar baseada na razão  $L = \frac{1}{2} \mathbf{r}_{LL}^H \left( N_0^{-1} \mathbf{I} - (\mathbf{R} + N_0 \mathbf{I})^{-1} \right) \mathbf{r}_{LL}$ .

Para avaliar o desempenho deste detector, usamos o critério de deflexão, definido por:

$$D = \frac{(E_1(L) - E_2(L))^2}{V_0(L)}. \quad (\text{A.24})$$

Aqui  $E_i(\cdot)$  denota a esperança sob a hipótese  $i$ , e  $V_0(L)$  denota variância sob a hipótese zero. A deflexão pode ser considerada como uma relação sinal-ruído de saída. Isso é mais fácil calcular que a curva característica completa do receptor. Depois de alguma álgebra, a deflexão pode ser calculada por:

$$D = \frac{1}{2N_0^2} \frac{\left( \sum_{n=1}^{2N} \frac{\lambda_n^2}{\lambda_n + N_0} \right)^2}{\sum_{n=1}^{2N} \left( \frac{\lambda_n}{\lambda_n + N_0} \right)^2}. \quad (\text{A. 25})$$

É possível mostrar que  $D$  é uma função Schur convexa dos autovalores  $\lambda_n$ , com o parâmetro  $N_0$ . Usando novamente o teorema, o ganho máximo de desempenho do processamento LL sobre o processamento linear, para esta aplicação, é dado por:

$$\begin{aligned} & \max_{N_0} \left[ D([\lambda, \lambda], N_0) / D([2\lambda, \mathbf{0}_N], N_0) \right] = \\ & \max_{N_0} \left[ 2 \frac{\left( \sum_{n=1}^N \frac{\lambda_n^2}{2\lambda_n + N_0} \right)^2 \sum_{n=1}^N \left( \frac{\lambda_n}{\lambda_n + N_0} \right)^2}{\sum_{n=1}^N \left( \frac{\lambda_n}{\lambda_n + N_0} \right)^2 \left( \sum_{n=1}^N \frac{\lambda_n^2}{\lambda_n + N_0} \right)^2} \right] \rightarrow 2. \end{aligned} \quad (\text{A. 26})$$

O ganho de desempenho máximo é 2 e este valor é alcançado de forma assintótica quando  $N_0 \rightarrow 0$  ou  $N_0 \rightarrow \infty$ .

A Figura A.1 mostra o ganho do processamento LL sobre o linear na estimação, segundo a expressão (A.22). A Figura A.2 apresenta o ganho para o caso de detecção, segundo a expressão (A.26). Nestes exemplos consideramos  $\lambda = [1,0; 0,75; 0,63; 0,50; 0,35; 0,25; 0,10; 0,10; 0,08; 0,06; 0,05; 0,04]$ .

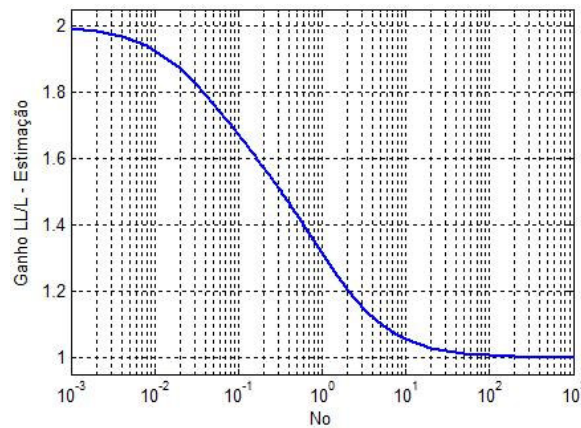


Figura A.1: Ganho do processamento LL sobre o linear para o caso de estimação

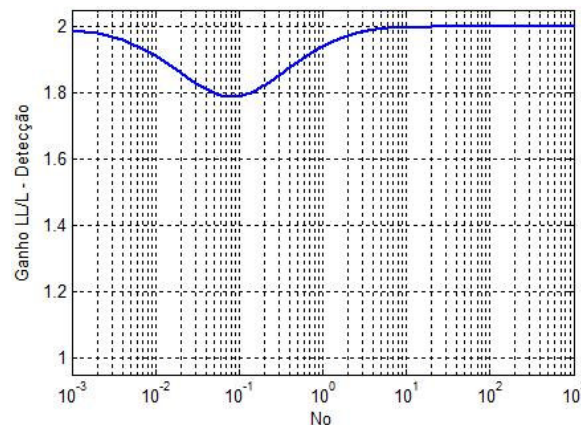


Figura A.2: Ganho do processamento LL sobre o linear para o caso de detecção

## Apêndice B – Complementos do Capítulo 3

Neste apêndice apresentamos alguns complementos ao Capítulo 3. Na primeira seção deduzimos o sistema de equações (3.6) e comentamos sobre o atraso ótimo para a equalização de um canal. Na seção seguinte, comparamos as estimativas de variância de um sinal aleatório usando as equações (3.53) e (3.58) e apresentamos também algumas formas de se obter estimativas da matriz de correlação e de sua inversa. Na última seção deste apêndice, apresentamos o impacto do aumento do número de coeficientes no desempenho de um equalizador LL.

### B.1 Sistema de equações (3.6) e atraso ótimo

Dividimos essa seção em duas partes. Na primeira sub-seção deduzimos o sistema de equações (3.6) e na segunda sub-seção analisamos o atraso ótimo para os equalizadores ótimos segundo os critérios ZF e MSE.

#### B.1.1 Dedução do sistema de equações (3.6)

O erro na resposta do equalizador largamente linear é dado por

$$e[k] = a[k - k_0] - (\mathbf{f}^H \mathbf{r}[k] + \mathbf{g}^H \mathbf{r}^*[k]), \quad (\text{B.1})$$

onde  $\mathbf{r}[k]$  pode ser descrito como

$$\mathbf{r}[k] = \mathbf{H}\mathbf{a}[k] + \boldsymbol{\eta}[k], \quad (\text{B.2})$$

onde a matriz  $\mathbf{H}$  do canal apresenta uma dimensão  $(P+1) \times (P+L+1)$  e pode ser descrita por

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \dots & h_L & 0 & \dots & 0 \\ 0 & h_0 & h_1 & \dots & h_L & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_0 & h_1 & \dots & h_L \end{bmatrix}, \quad (\text{B.3})$$

$\mathbf{a}[k] = [a[k] \ a[k-1] \ \dots \ a[k-P-L]]^T$  é um vetor  $(P+L+1) \times 1$  contendo os símbolos transmitidos e vetor de ruído  $\boldsymbol{\eta}[k] = [\eta[k] \ \eta[k-1] \ \dots \ \eta[k-P-L]]^T$  tem dimensão igual a

$(P+1) \times 1$ . A minimização de  $J = E\{|e[k]|^2\}$  segue as regras padrões de derivação complexa aplicadas a vetores:

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{f}^*} &= E \left\{ \frac{\partial e}{\partial \mathbf{f}^*} e^* + e \frac{\partial e^*}{\partial \mathbf{f}^*} \right\} = 0 \\ &= E \{ \mathbf{r} e^* + 0 \} = 0 \\ &= E \left\{ (\mathbf{H}\mathbf{a} + \boldsymbol{\eta}) (a[k - k_0] - (\mathbf{r}^H \mathbf{f} + \mathbf{r}^T \mathbf{g})) \right\} = 0 \end{aligned} \quad (\text{B.4})$$

Como  $\mathbf{r}^H = \mathbf{a}^H \mathbf{H}^H + \boldsymbol{\eta}^H$ , temos:

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{f}^*} &= E \left\{ -\mathbf{H}\mathbf{a}\mathbf{a}^H \mathbf{H}^H \mathbf{f} - \boldsymbol{\eta} (\mathbf{a}^H \mathbf{H}^H + \boldsymbol{\eta}^H) \mathbf{f} \right\} + \\ &+ E \left\{ -\mathbf{H}\mathbf{a}\mathbf{a}^T \mathbf{H}^T \mathbf{g} - \boldsymbol{\eta} (\mathbf{a}^T \mathbf{H}^T + \boldsymbol{\eta}^T) \mathbf{g} \right\} \\ &+ E \left\{ \mathbf{H}\mathbf{a} a[k - k_0] + \boldsymbol{\eta} a[k - k_0] \right\} = 0. \end{aligned} \quad (\text{B.5})$$

Assumindo ruído branco, os símbolos i.i.d e pertencentes a uma constelação real, podemos simplificar (B.5) para:

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{f}^*} &= E \left\{ -\sigma_a^2 \mathbf{H}\mathbf{H}^H \mathbf{f} - \sigma_\eta^2 \mathbf{f} \right\} + \\ &+ E \left\{ -\sigma_a^2 \mathbf{H}\mathbf{H}^T \mathbf{g} \right\} \\ &+ E \left\{ -\sigma_a^2 \mathbf{h}_{k_0} \right\} = 0, \end{aligned} \quad (\text{B.6})$$

onde  $\mathbf{h}_{k_0} = [h_{k_0} \ h_{k_0-1} \ \dots \ h_{k_0-P}]^T$ . Logo, a partir de (B.6), podemos escrever a equação:

$$(\mathbf{H}\mathbf{H}^H + \xi \mathbf{I}_P) \mathbf{f} + \mathbf{H}\mathbf{H}^T \mathbf{g} = \mathbf{h}_{k_0}, \quad (\text{B.7})$$

onde  $\mathbf{I}_P$  é uma matriz identidade de dimensão  $P+1$  e  $\xi = \sigma_n^2 / \sigma_a^2$  é a relação entre a variância do ruído e do sinal. De forma semelhante, derivando  $J$  em relação ao vetor  $\mathbf{g}$ , obtemos:

$$\mathbf{H}^* \mathbf{H}^H \mathbf{f} + (\mathbf{H}^* \mathbf{H}^T + \xi \mathbf{I}_P) \mathbf{g} = \mathbf{h}_{k_0}^*, \quad (\text{B.8})$$

(B.7) e (B.8) formam o sistema de equações descrito em (3.6).

### B.1.2 Atraso ótimo

O atraso ótimo para um equalizador LL vai depender do critério de otimização utilizado (MSE ou ZF), do número de coeficientes do equalizador e do nível de ruído (para um equalizador MSE). Em geral, o atraso ótimo para um equalizador LL MSE será o mesmo que para um equalizador LL ZF se ambos tiverem o mesmo número de coeficientes, a não ser que o nível de ruído seja muito alto (baixa relação sinal-ruído).

Para um equalizador LL ótimo  $\mathbf{w}_{opt}$ , segundo o critério MSE, com um dado atraso  $\Delta$  de equalização, o erro na saída do equalizador, considerando que o ruído é AWGN e a fonte de sinal é i.i.d., pode ser expresso por:

$$\begin{aligned}
 e[k] &= a[k - \Delta] - \mathbf{w}_{opt}^H \mathbf{r}[k] \\
 e[k] &= a[k - \Delta] - \mathbf{w}_{opt}^H \mathbf{x}[k] + \mathbf{w}_{opt}^H \boldsymbol{\eta}[k] \\
 e[k] &= \underbrace{a[k - \Delta] - \mathbf{w}_{opt}^H \mathbf{H}_c \mathbf{a}[k]}_{\text{IES residual}} + \underbrace{\mathbf{w}_{opt}^H \boldsymbol{\eta}[k]}_{\text{ruído filtrado}},
 \end{aligned} \tag{B.9}$$

sendo  $\mathbf{H}_c = [\mathbf{H}^T \ \mathbf{H}^H]$  e  $\mathbf{H}$  definido em (B.3). Notamos que a IES residual em (B.9) pode ser escrita como

$$IES[k] = \mathbf{a}^T[k] (\mathbf{h}_\Delta - \mathbf{H}_c \mathbf{w}_{opt}^*), \tag{B.10}$$

sendo  $\mathbf{h}_\Delta$  um vetor coluna com um único elemento unitário na posição  $\Delta$  e zero nas demais. Considerando o ruído filtrado e a IES residual independentes, podemos escrever o MSE resultante como

$$\begin{aligned}
 J(\Delta) &= \sigma_a^2 (\mathbf{h}_\Delta - \mathbf{H}_c \mathbf{w}_{opt}^*)^H (\mathbf{h}_\Delta - \mathbf{H}_c \mathbf{w}_{opt}^*) + \sigma_\eta^2 \mathbf{w}_{opt}^T \mathbf{w}_{opt}^* \\
 &= \sigma_a^2 (\mathbf{h}_\Delta^H \mathbf{h}_\Delta - \mathbf{h}_\Delta^H \mathbf{H}_c \mathbf{w}_{opt}^* - \mathbf{w}_{opt}^T \mathbf{H}_c^H \mathbf{h}_\Delta + \mathbf{w}_{opt}^T \mathbf{H}_c^H \mathbf{H}_c \mathbf{w}_{opt}^*) + \sigma_\eta^2 \mathbf{w}_{opt}^T \mathbf{w}_{opt}^* \\
 &= \sigma_a^2 (\mathbf{h}_\Delta^H \mathbf{h}_\Delta - \mathbf{h}_\Delta^H \mathbf{H}_c \mathbf{w}_{opt}^* - \mathbf{w}_{opt}^T \mathbf{H}_c^H \mathbf{h}_\Delta + \mathbf{w}_{opt}^T (\mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I}) \mathbf{w}_{opt}^*),
 \end{aligned} \tag{B.11}$$

Lembrando que  $(\mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I}) \mathbf{w}_{opt} = \mathbf{H}_c \mathbf{h}_\Delta$ , podemos simplificar (B.11):

$$\begin{aligned}
J(\Delta) &= \sigma_a^2 \left( \mathbf{h}_\Delta^H \mathbf{h}_\Delta - \mathbf{h}_\Delta^H \mathbf{H}_c \mathbf{w}_{opt}^* - \mathbf{w}_{opt}^T \mathbf{H}_c^H \mathbf{h}_\Delta + \mathbf{w}_{opt}^T \mathbf{H}_c^H \mathbf{h}_\Delta \right) \\
&= \sigma_a^2 \left( \mathbf{h}_\Delta^H \mathbf{h}_\Delta - \mathbf{h}_\Delta^H \mathbf{H}_c \left( \mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I} \right)^{-1} \mathbf{H}_c^H \mathbf{h}_\Delta \right) \\
&= \sigma_a^2 \mathbf{h}_\Delta^H \left( \mathbf{I} - \mathbf{H}_c \left( \mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I} \right)^{-1} \mathbf{H}_c^H \right) \mathbf{h}_\Delta.
\end{aligned} \tag{B.12}$$

Como  $\mathbf{h}_\Delta$  é um atraso puro, o produto  $\mathbf{h}_\Delta^H \mathbf{M} \mathbf{h}_\Delta$ , sendo  $\mathbf{M}$  uma matriz quadrada, resulta no elemento da diagonal principal correspondente à coluna e linha de índices  $\Delta$ . Logo, para a minimização de  $J(\Delta)$ , segundo o critério MSE, o atraso ótimo  $\Delta$  deve ser escolhido de tal forma que selecione o menor elemento da diagonal principal de  $\mathbf{I} - \mathbf{H}_c \left( \mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I} \right)^{-1} \mathbf{H}_c^H$ . Em termos matemáticos:

$$\Delta_{opt-MSE} = \arg \min_{\Delta} \left\{ \mathbf{h}_\Delta^H \left( \mathbf{I} - \mathbf{H}_c \left( \mathbf{H}_c^H \mathbf{H}_c + \xi \mathbf{I} \right)^{-1} \mathbf{H}_c^H \right) \mathbf{h}_\Delta \right\}. \tag{B.13}$$

A expressão para cálculo do atraso ótimo para um equalizador LL ZF é bem mais simples (ver 3.17):

$$\Delta_{opt-ZF} = \arg \min_{\Delta} \left\| \mathbf{H}^+ \mathbf{h}_{zf-\Delta} \right\|_2. \tag{B.14}$$

No exemplo a seguir, comparamos o atraso ótimo de um equalizador LL MSE e outro LL ZF em função da relação sinal-ruído. Consideramos o canal definido por  $\mathbf{h} = [0,5312; -0,1328 + 0,1062j; 0,2443 + 0,6719j; -0,3333 - 0,2583j]$ , uma constelação 4-PAM e equalizadores LL com 10 ou 12 coeficientes (5 ou 6 no filtro  $\mathbf{f}$  e 5 ou 6 no filtro  $\mathbf{g}$ ). Um número maior de coeficientes traz um maior grau de liberdade e pode resultar em um menor erro residual (IES residual + amplificação de ruído) na saída do equalizador.

Na Figura B.1, vemos a comparação entre os equalizadores LL ZF e LL MSE, ambos com 10 coeficientes. Percebemos que o desempenho dos dois equalizadores é muito similar quando a relação sinal-ruído é maior ou igual a 30 dB. Para essa situação, o atraso ótimo dos equalizadores é igual a dois. Já para uma relação sinal-ruído de apenas 10 dB, o atraso ótimo do equalizador LL MSE é três.

Na Figura B.2, comparamos os equalizadores MSE e ZF com 12 coeficientes cada. Podemos observar que ocorre uma ligeira melhora no desempenho geral dos equalizadores. O atraso ótimo, nessa nova configuração, é três quando a relação sinal-ruído é alta. Quando a

relação sinal-ruído é baixa (10 dB), o atraso ótimo muda para quatro.

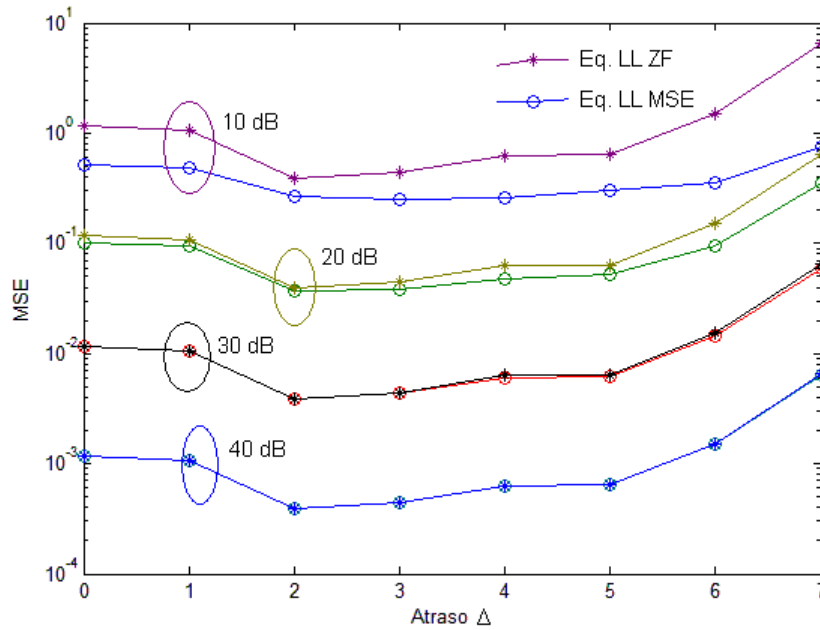


Figura B.1: Atraso ótimo para equalizadores LL (ZF e MSE) com 10 coeficientes em função da relação sinal-ruído para o canal  $\mathbf{h}$ .

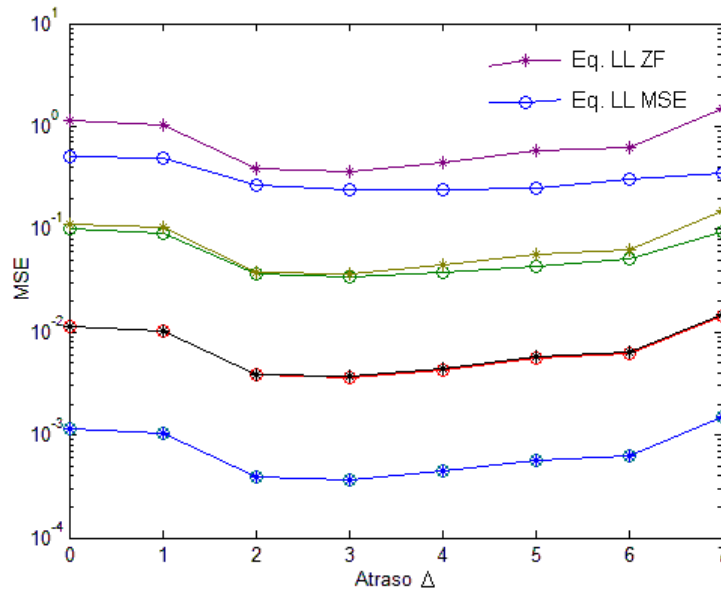


Figura B.2: Atraso ótimo para equalizadores LL (ZF e MSE) com 12 coeficientes em função da relação sinal-ruído para o canal  $\mathbf{h}$ .



## B.2 Comparação das estimativas de variância de um sinal aleatório usando as equações (3.53) e (3.58) e outras estimativas

Podemos verificar facilmente, por meio de simulações, que as estratégias para cálculo da variância de um sinal randômico dadas pelas equações (3.53) e (3.58) levam ao mesmo resultado quando o número  $k$  de iterações cresce. As maiores discrepâncias ocorrem apenas nas primeiras iterações, se o valor de  $s_i[k]$  (em 3.58) for inicializado com zero, como mostra a Figura B.3. Na Figura B.4 temos a mesma comparação, mas com (3.58) sendo inicializada com um.

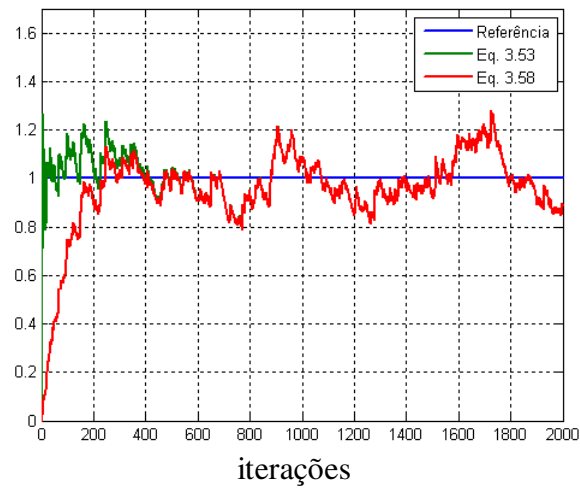


Figura B.3. Comparação entre as equações (3.53) e (3.58), com  $s_i[k]$  inicializado com zero e  $\gamma = 0,99$ . Sinal aleatório de média nula e variância unitária.

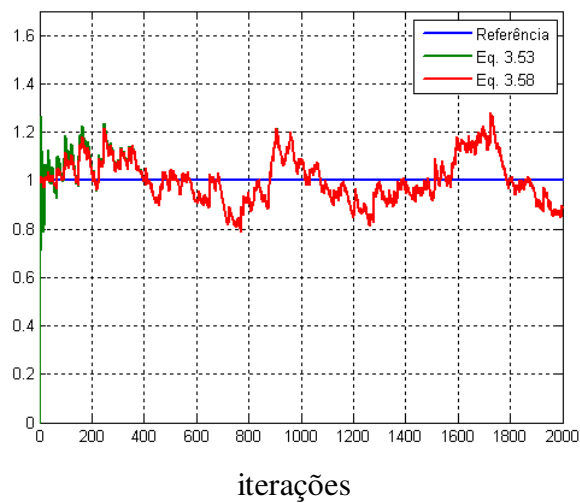


Figura B.4. Comparação entre as equações (3.53) e (3.58), com  $s_i[k]$  inicializado com um e  $\gamma = 0,99$ . Sinal aleatório de média nula e variância unitária.

De uma forma mais geral, podemos estimar, além da variância, a matriz de correlação pela mediação de  $N$  amostras de vetores de um sinal por (Godara, 2004):

$$\tilde{\mathbf{R}}(N) \approx \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{x}(n) \mathbf{x}^H(n). \quad (\text{B.15})$$

Esta estimativa (não polarizada) também pode ser obtida iterativamente por (Godara, 2004):

$$\tilde{\mathbf{R}}(n+1) \approx \frac{n\tilde{\mathbf{R}}(n) + \mathbf{x}(n) \mathbf{x}^H(n)}{n+1}. \quad (\text{B.16})$$

Podemos definir o *erro de estimação* entre a matriz de correlação estimada e a matriz de correlação exata como sendo a norma de Frobenius<sup>16</sup> da matriz diferença:  $\varepsilon_{\mathbf{R}}(N) = \|\tilde{\mathbf{R}}(N) - \mathbf{R}_{\mathbf{x}}\|_F$ . Na Figura B.5, temos um exemplo que mostra o valor médio do erro de estimação para 10 (dez) realizações. Neste exemplo, consideramos que o vetor de dados  $\mathbf{x}$  é gerado a partir da convolução de uma seqüência BPSK de variância unitária pelo canal real  $\mathbf{h} = [0,882; -0,378; 0,252; -0,126]$  e uma relação sinal-ruído de 30 dB. Consideramos ainda, para este exemplo, que o vetor de dados tem dimensão  $6 \times 1$  e que a matriz  $\mathbf{R}_{\mathbf{x}}$  tem dimensão  $6 \times 6$ .

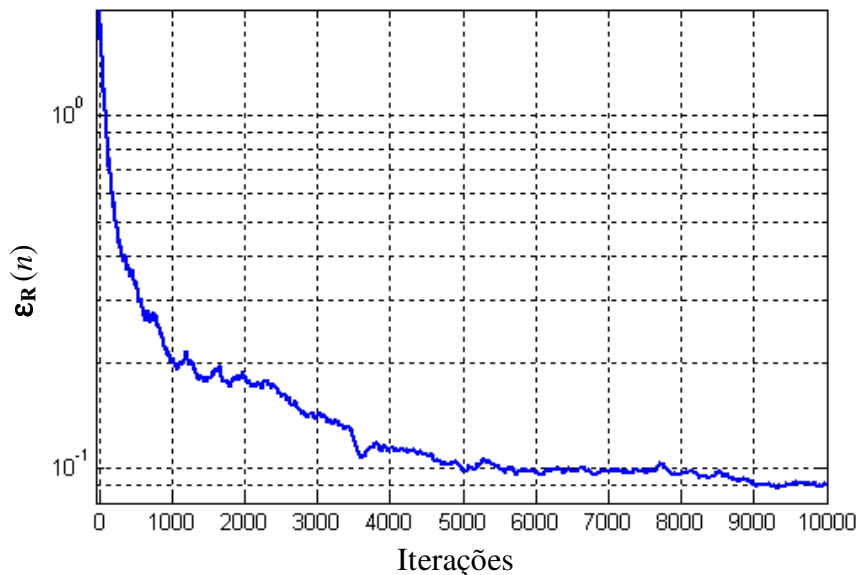


Figura B.5: Evolução do erro de estimação  $\varepsilon_{\mathbf{R}}$  da matriz de correlação

<sup>16</sup> A norma de Frobenius de uma matriz é definida como  $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2}$ , sendo  $a_{ij}$  um elemento da matriz  $\mathbf{A}$  (G. W. Stewart, *Matrix Algorithms*, Vol. 1, SIAM, Philadelphia, 1998).

Considerando um ambiente estacionário, podemos calcular a matriz inversa de  $\tilde{\mathbf{R}}$  por:

$$\tilde{\mathbf{R}}^{-1}(n+1) \approx \tilde{\mathbf{R}}^{-1}(n) - \frac{\tilde{\mathbf{R}}^{-1}(n)\mathbf{x}(n)\mathbf{x}^H(n)\tilde{\mathbf{R}}^{-1}(n)}{1 + \mathbf{x}^H(n)\tilde{\mathbf{R}}^{-1}(n)\mathbf{x}(n)}, \quad (\text{B.17})$$

com  $\tilde{\mathbf{R}}(0) = \frac{1}{\varepsilon_0} \mathbf{I}$ ,  $\varepsilon_0 > 0$ .

Consideraremos agora o aumento no fator de diagonalização da matriz de correlação largamente linear  $\mathbf{R}_{LL}$  devido ao procedimento *multi-split*, ver (3.67), através de um exemplo numérico. Para isso, usaremos um conjunto de mil canais complexos aleatórios cuja resposta ao impulso  $\mathbf{h}$  consiste de cinco ou oito coeficientes. O número de coeficientes do equalizador LL-MS será de 4, 8, 16 ou 32 em cada filtro do equalizador. Na Tabela B.1 temos um resumo dos resultados obtidos, explicitando os ganhos médio, máximo e mínimo em função da ordem da transformação MS. Constatamos que o aumento médio no fator de diagonalização só é significativo quando  $N \leq 16$ .

TABELA B.1: FATOR DE DIAGONALIZAÇÃO DO EQUALIZADOR LL-MS EM FUNÇÃO DE  $N$

Canais aleatórios complexos de ordem 4					
$N$	$\gamma(R_{\perp})$	$\gamma(R_{LL})$	$\gamma(R_{\perp})/\gamma(R_{LL})_{\text{médio}}$	$\gamma(R_{\perp})/\gamma(R_{LL})_{\text{min}}$	$\gamma(R_{\perp})/\gamma(R_{LL})_{\text{max}}$
4	0,6004	0,3666	1,6378	1,1912	3,1656
8	0,4082	0,2761	1,4788	1,1599	2,4627
16	0,3102	0,2461	1,2602	0,9716	2,0144
32	0,2504	0,2336	1,0723	0,8273	1,6724
Canais aleatórios complexos de ordem 7					
4	0,6010	0,3536	1,6997	1,1007	3,7341
8	0,3889	0,2382	1,6323	1,2590	3,2386
16	0,2873	0,2024	1,4192	1,0361	2,7437
32	0,2610	0,2433	1,0727	0,8234	1,6799

### B.3 Número de coeficientes *versus* $J_{min}$ para o equalizador LL

Mostraremos nesta seção, por meio de simulações, o efeito do aumento do número de coeficientes no desempenho do equalizador LL. Para isto, iremos considerar três canais do

tipo FIR distintos: o primeiro de fase mínima, o segundo de fase mista e o terceiro de fase máxima. Todos estes canais têm seis zeros. Consideraremos apenas um nível de relação sinal-ruído: 30 dB. Lembramos que o MMSE do equalizador LL é dado por (ver equação 3.23):

$$J_{min} = \sigma_a^2 - \mathbf{p}^H \mathbf{w}_{opt} = \sigma_a^2 - \mathbf{p}^H \mathbf{R}_{LL}^{-1} \mathbf{p}, \quad (\text{B.18})$$

e que o valor do  $J_{min}$  também depende do atraso  $k_0$  adotado. Os canais considerados são:

- $\mathbf{h}_{misto} = [0,3612 + 0,1084j; 0,0144 + 0,0831j; 0,5013 - 0,0701j; 0,4784 + 0,3947j; 0,2532 + 0,2736j; -0,0640 + 0,2453j; -0,0813 - 0,0038j];$
- $\mathbf{h}_{min} = [0,6857 + 0,2057j; -0,1406 + 0,5931j; -0,1812 - 0,0936j; 0,0101 + 0,0255j; 0,1826 - 0,1097j; 0,0773 + 0,1410j; -0,0468 + 0,0158j];$
- $\mathbf{h}_{max} = [0,1398 + 0,0419j; -0,0587 + 0,1195j; -0,3046 - 0,1402j; -0,0156 - 0,0614j; 0,3331 + 0,2501j; -0,2000 + 0,3119j; -0,5233 - 0,5098j].$

Na Figura B.6 vemos os zeros destes canais e suas respectivas respostas em frequência. Para estes canais, calcularemos de forma numérica o *atraso ótimo* e o MMSE obtido para o equalizador LL para um dado comprimento do equalizador e uma dada relação sinal-ruído. A título de comparação, fazemos o mesmo considerando um equalizador linear. Estamos considerando que o sinal a ser equalizado pertence a uma constelação 4-PAM com  $\sigma_a^2 = 5$  e levaremos em conta apenas a parte real do erro. Apresentamos os resultados em forma de gráficos nas figuras B.6 a B.9.

A partir dos gráficos apresentados (obtemos resultados aproximadamente semelhantes para diferentes SNR's), podemos tirar algumas conclusões gerais em relação aos canais avaliados:

- o atraso ótimo para o equalizador LL é menor que para o equalizador linear, o que pode ser vantajoso em sistemas de comunicação práticos;
- para um mesmo número de coeficientes, o desempenho do equalizador LL é sempre melhor, o que pode levar a um equalizador mais econômico em termos da utilização de recursos (área em hardware, consumo de energia);
- o desempenho do equalizador LL é menos sensível ao aumento do número de coeficientes (diferença entre o MMSE com poucos e muitos coeficientes).

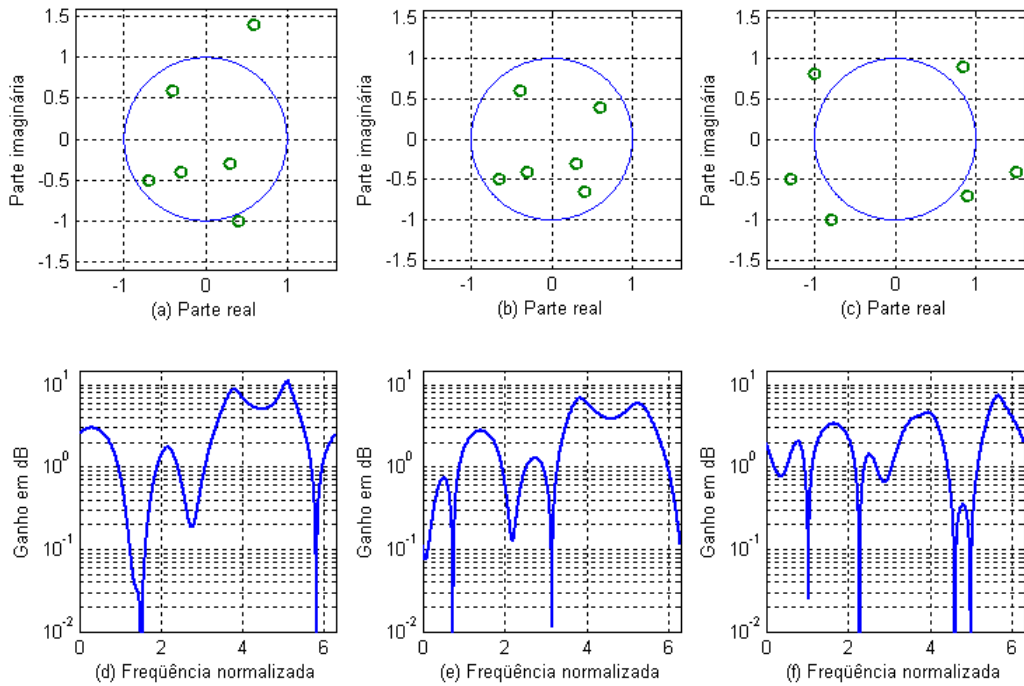


Figura B.6: Características dos canais. (a) Zeros do canal de fase mista, (b) zeros do canal de fase mínima, (c) zeros do canal de fase máxima, (d) resposta em freqüência do canal de fase mista, (e) resposta em freqüência do canal de fase mínima, (f) resposta em freqüência do canal de fase máxima.

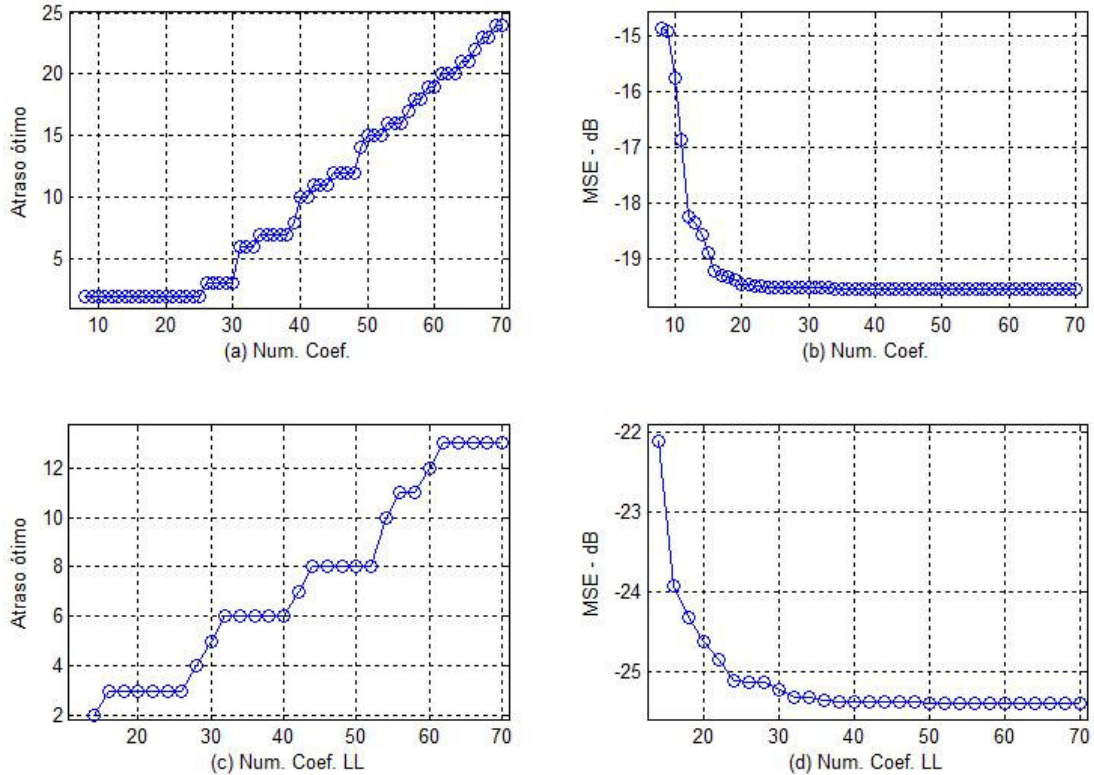


Figura B.7: Atraso ótimo e MMSE para o canal de fase mínima, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE em função do número de coeficientes para o equalizador LL.

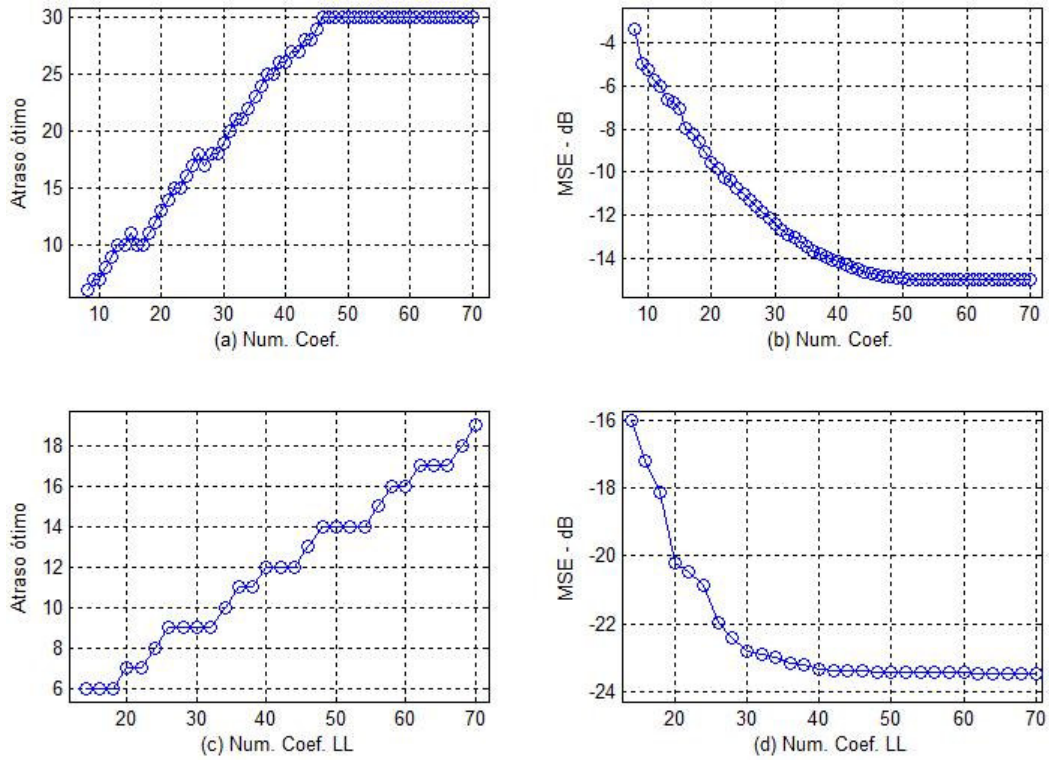


Figura B.8: Atraso ótimo e MMSE para o canal de fase mista, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE em função do número de coeficientes para o equalizador LL.

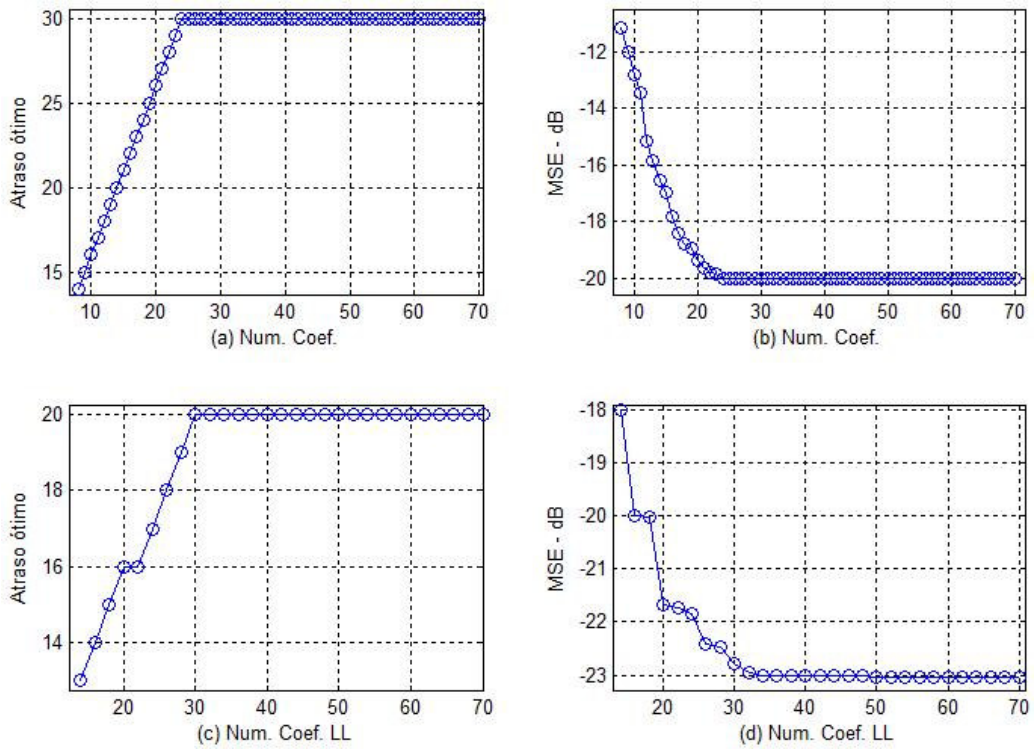


Figura B.9: Atraso ótimo e MMSE para o canal de fase máxima, (a) atraso ótimo e (b) MMSE em função do número de coeficientes para o equalizador linear, (c) atraso ótimo e (d) MMSE em função do número de coeficientes para o equalizador LL.

Para os canais considerados e uma SNR alta (acima de 30 dB), as vantagens do equalizador LL se ampliariam em relação ao equalizador linear, já que ele pode conseguir uma inversão quase ideal do canal (gerando uma baixa IES) sem amplificar significativamente o ruído. Se nos canais considerados, um (ou mais) zero(s) fosse(m) deslocado(s) para muito próximo ou sobre o CRU, o desempenho do equalizador linear seria seriamente prejudicado enquanto o desempenho do equalizador LL não sofreria alteração significativa.

## B.4 Efeito da latência: algoritmo DLMS

Quando implementado no modo totalmente paralelo, o algoritmo LMS se torna um algoritmo LMS atrasado (DLMS) devido à latência das multiplicações. Nesta seção, apresentamos um exemplo simples de um equalizador real de apenas um coeficiente para facilitar a visualização do efeito da latência na estabilidade do algoritmo DLMS. Para que os efeitos de precisão finita não tenham influência significativa, a palavra digital nesse equalizador têm 16 bits, sendo 12 para a parte fracionária. Com isso, evitamos os erros de saturação (“*overflow*”) e os erros de arredondamento e truncamento são desprezíveis.

Na Figura B.10, vemos um equalizador linear de apenas um coeficiente. É fácil percebermos que, devido à existência de uma latência nos blocos multiplicadores (Mult1 e Mult2), para ocorrer o sincronismo nas operações do algoritmo LMS é necessário a inclusão de alguns atrasos (Delay1 e Delay2), transformando esse algoritmo em sua versão atrasada (DLMS).

Podemos escrever o algoritmo DLMS real com apenas um coeficiente como:

$$\begin{aligned} y[k] &= w[k-D]x[k-D], \\ e[k] &= d[k-D] - y[k], \\ w[k] &= w[k-1] + \mu e[k]x[k-D], \end{aligned} \tag{B.19}$$

onde  $D$  é a latência e  $\mu$  é o passo de adaptação. Na simulação a seguir, consideramos um sistema sem ruído, canal ideal, com os dados pertencentes a uma constelação BPSK e  $w[0] = 0$ . Para evidenciarmos o efeito da latência, variamos  $D$  dentro do intervalo  $D = 1, 2, 3$  e  $4$ , e o passo  $\mu \in \{\mu_{\max}/5, \mu_{\max}/4, \mu_{\max}/3, \mu_{\max}/2, \mu_{\max}\}$ , sendo  $\mu_{\max}$  calculado simplesmente por:

$$\mu_{\max} = \frac{1}{D+1}, \tag{B.20}$$

note que esta é uma simplificação da expressão mais geral indicada em (3.39). O valor  $\mu_{\max}$  é um limitante superior para o maior valor que o passo de adaptação pode assumir e que ainda garante a convergência do algoritmo DLMS.

Em uma implementação prática em FPGA usando o sistema da Virtex IV, o valor da latência  $D$  é, em geral, igual a 3 ou 4.

Na Figura B.11, vemos os resultados das simulações. Verificamos que quanto maior a latência das multiplicações, mais próximo à instabilidade fica o algoritmo para  $\mu = \mu_{\max}$ . Quando a latência é de apenas uma amostra, o algoritmo DLMS se comporta de forma muito semelhante à sua versão não atrasada.

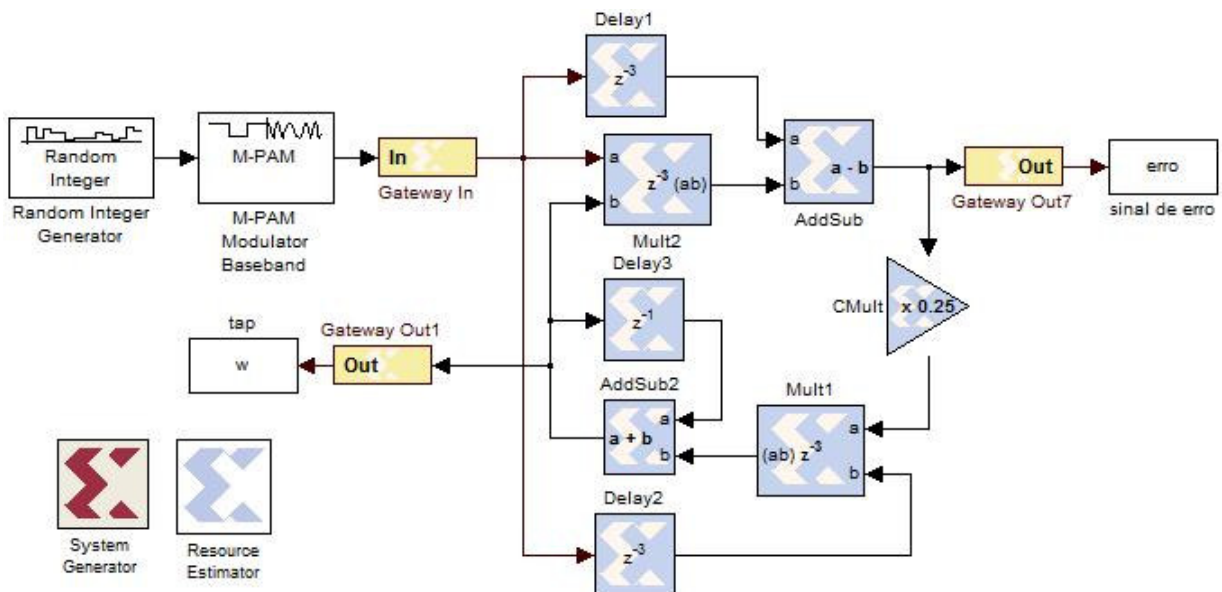


Figura B.10: Equalizador FPGA de apenas um coeficiente, todos os multiplicadores com latência  $D = 3$ .



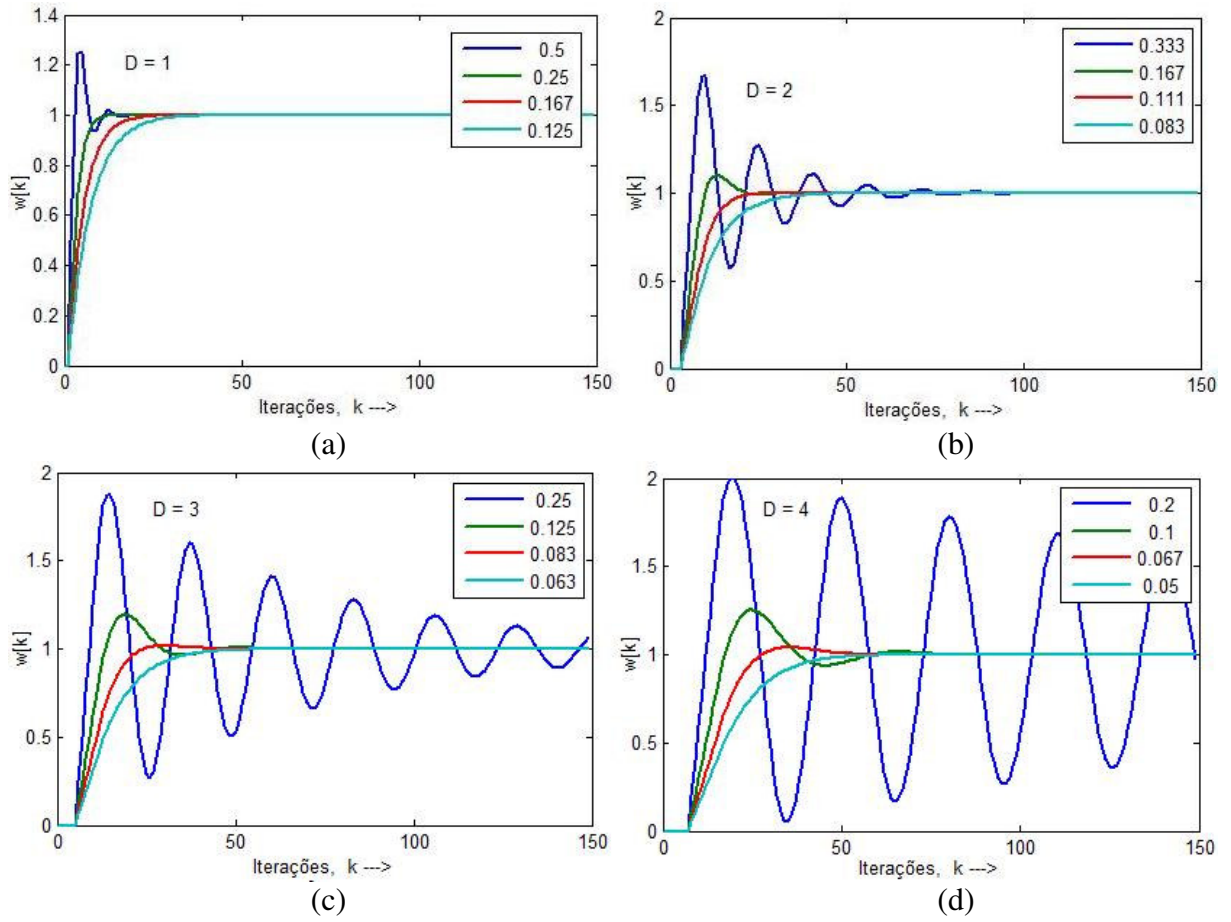


Figura B.11: Efeito da latência e do passo no comportamento do algoritmo DLMS, (a) latência de apenas uma amostra, (b) latência de duas amostras, (c) latência de três amostras e (d) latência de quatro amostras.

# Apêndice C – Complementos do Capítulo 4

Neste apêndice apresentamos alguns algoritmos cegos já bastante conhecidos na literatura e alguns complementos ao Capítulo 4. Na Seção C.1, discutimos sobre os algoritmos cegos mais referenciados na literatura técnica. Essa seção é baseada, em parte, no Capítulo 6 da tese de Mathis (2001). Na Seção C.2 comparamos as soluções ZF e MSE para o FEPLL em função da relação sinal-ruído. Na Seção C.3, apresentamos alguns detalhes da implementação em FPGA do FEPLL.

## C.1 Algoritmos cegos

Desde o trabalho pioneiro de Sato (1975) sobre equalização cega para sinais de comunicação, vários outros autores analisaram e/ou desenvolveram diversos algoritmos com o mesmo objetivo. Na Figura C.1, vemos a estrutura do equalizador linear cego que estamos considerando nesta tese. Para esta estrutura, o filtro  $w[k]$  é inicialmente nulo, com exceção do coeficiente central que é feito igual à unidade (inicialização “center spike”). Observamos que na Figura C.1 não está explícito nenhum dispositivo para o controle automático de ganho (CAG) para a correção de magnitude e de fase do sinal após o equalizador ( $\tilde{a}[k]$ ).

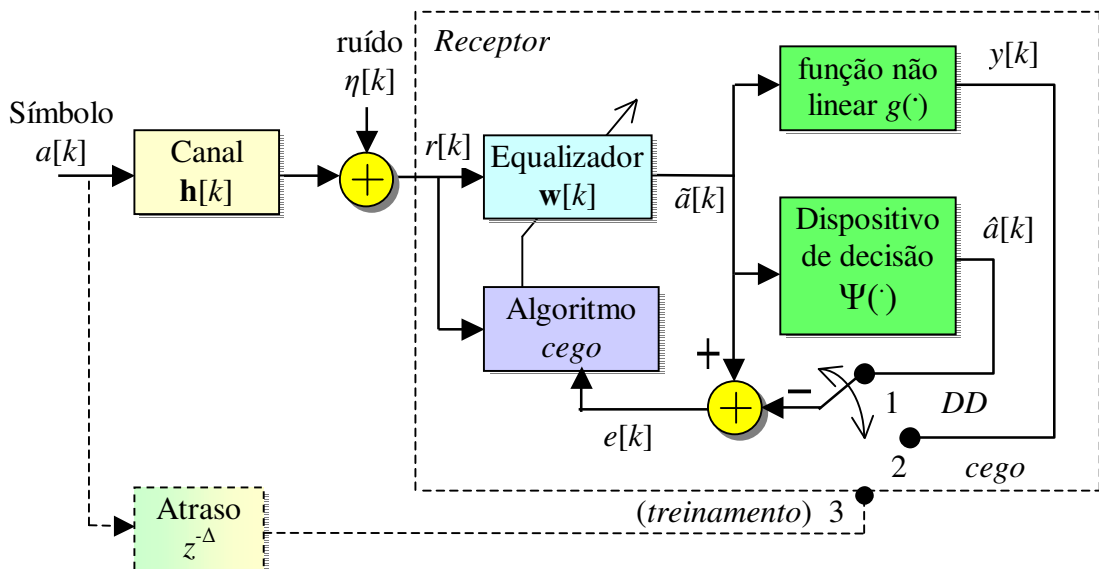


Figura C.1 Equalizador cego e seus modos de operação.

### C.1.1 Algoritmo de Sato

Considerando um sistema de comunicação usando uma constelação real (PAM) e um canal também real, Sato (1975) propôs a seguinte equação recursiva para a adaptação (no modo cego) para o equalizador linear autodidata:

$$\mathbf{w}[k+1] = \mathbf{w}[k] - \mu (\gamma \text{sign}(\tilde{a}[k]) - \tilde{a}[k]) \mathbf{r}[k], \quad (\text{C.1})$$

onde função não linear  $g(\tilde{a}[k]) = \gamma \text{sign}(\tilde{a}[k])$  e o sinal de erro  $e_s[k] = \gamma \text{sign}(\tilde{a}[k]) - \tilde{a}[k]$ , sendo  $\mu$  o passo de adaptação e parâmetro  $\gamma$  (*constante de dispersão*) é definido por:

$$\gamma = \frac{E\{|a[k]|^2\}}{E\{a[k]\}}. \quad (\text{C.2})$$

É fácil verificar que para o caso de uma modulação BPSK, o algoritmo de Sato se reduz ao algoritmo de decisão direta. A função custo a ser minimizada é:

$$J_s(\mathbf{w}) = E\{|e_s[k]|^2\} = E\{(|\tilde{a}[k]| - \gamma)^2\}. \quad (\text{C.3})$$

O algoritmo de Sato pode ser estendido para o caso de constelações complexas pela separação do sinal em suas partes real e imaginária (Benvineste e Goursat, 1984):

$$e_{sc}[k] = \gamma_R \text{sign}(\tilde{a}_R[k]) - \tilde{a}_R[k] + j(\gamma_I \text{sign}(\tilde{a}_I[k]) - \tilde{a}_I[k]), \quad (\text{C.4})$$

onde  $\tilde{a}_R[k]$  e  $\tilde{a}_I[k]$  são as partes real e imaginária de  $\tilde{a}[k]$  e

$$\gamma_R = \frac{E\{|a_R[k]|^2\}}{E\{a_R[k]\}}, \quad \gamma_I = \frac{E\{|a_I[k]|^2\}}{E\{a_I[k]\}}. \quad (\text{C.5})$$

### C.1.2 Algoritmos de Godard e CMA

Talvez o algoritmo cego mais popular seja o CMA (*constant-modulus algorithm*), um dos membros da família de algoritmos desenvolvida por Godard (1980), devido a sua baixa complexidade computacional. Godard considerou uma função custo da forma

$$J_G(\mathbf{w}) = E \left\{ \left( |\tilde{a}[k]|^p - R_p \right)^2 \right\}, \quad (\text{C.6})$$

sendo

$$R_p = \frac{E \left\{ |a[k]|^{2p} \right\}}{E \left\{ |a[k]|^p \right\}}, \quad (\text{C.7})$$

uma constante de dispersão de ordem  $p$ . A atualização dos coeficientes pode ser obtida a partir do gradiente de (C.6):

$$\begin{aligned} \mathbf{w}[k+1] &= \mathbf{w}[k] - \mu e_G^*[k] \mathbf{r}[k], \\ e_G[k] &= \tilde{a}[k] \left( |\tilde{a}[k]|^p - R_p \right) |\tilde{a}[k]|^{p-2}. \end{aligned} \quad (\text{C.8})$$

Para o caso de  $p=2$  temos o algoritmo CMA:

$$\begin{aligned} R_2 &= E \left\{ |a[k]|^4 \right\} / E \left\{ |a[k]|^2 \right\}, \\ e_{CMA}[k] &= \tilde{a}[k] \left( |\tilde{a}[k]|^2 - R_2 \right), \\ \mathbf{w}[k+1] &= \mathbf{w}[k] - \mu e_{CMA}^*[k] \mathbf{r}[k]. \end{aligned} \quad (\text{C.9})$$

### C.1.3 Algoritmo BGR (Benveniste-Goursat-Ruget)

Benveniste *et al.* (1980) apresentaram alguns resultados fundamentais válidos para muitos algoritmos cegos e apresentaram uma generalização do algoritmo de Sato, resultando na seguinte equação para adaptação dos coeficientes:

$$\mathbf{w}[k+1] = \mathbf{w}[k] - \mu \left( \gamma_{BGR} \text{sign}(\tilde{a}[k]) - f(\tilde{a}[k]) \right) \mathbf{r}[k], \quad (\text{C.10})$$

onde

$$\gamma_{BGR} = \frac{E\{f(a[k])a[k]\}}{E\{a[k]\}}. \quad (C.11)$$

### C.1.4 Algoritmo Benveniste-Goursat

O algoritmo Benveniste-Goursat (1984) é uma combinação suave (soma ponderada) dos erros de Sato e de decisão direta, sendo a função erro dada por:

$$e_{BG}[k] = \alpha_1 e_{DD}[k] + \alpha_2 |e_{DD}[k]| e_S[k], \quad (C.12)$$

onde  $\alpha_1$  e  $\alpha_2$  são constantes,  $e_S[k]$  é o erro de Sato e  $e_{DD}[k]$  o erro de decisão direta ( $e_{DD}[k] = \psi(\tilde{a}[k]) - \tilde{a}[k]$ ). Devido  $e_{BG}$  ser uma combinação de erros, não é possível se estabelecer uma função custo simples para este algoritmo. Na fase inicial de adaptação, quando a condição é de olho fechado, o segundo termo de (C.12) é dominante. Se houver convergência para a condição de olho aberto, o primeiro termo é o dominante. Deshpande (1997) percebeu que a troca do erro de Sato pelo erro CMA em (C.12) pode levar a um melhor desempenho do algoritmo BG.

### C.1.5 Algoritmo “Stop-and-Go”

Este algoritmo foi originalmente proposto por Picchi e Prati (1987). Da mesma forma que o algoritmo BG, o algoritmo Stop-and-Go é uma combinação dos erros de Sato e de decisão direta, sendo a função erro dada por:

$$e_{SG}[k] = \frac{1}{2} e_{DD}[k] + \frac{1}{2} |e_{DD}[k]| \text{sign}(e_S[k]). \quad (C.13)$$

A principal diferença entre (C.10) e (C.11) é que os termos de (C.11) se cancelam se os sinais de  $e_S$  e  $e_{DD}$  forem diferentes em um dado instante. Neste caso, o algoritmo não faz a adaptação dos coeficientes do equalizador, justificando o nome do algoritmo. Este algoritmo tende a ser mais robusto que o algoritmo BG.

O algoritmo *Stop-and-Go* recebeu a atenção de vários pesquisadores e diversas modificações foram propostas, ver, por exemplo, Hatzinakos (1992); Weerackody e Kassam (1994) e Kim *et al.* (1996).

### C.1.6 Outros algoritmos cegos

Além desses algoritmos “clássicos” e das suas diversas versões, vários outros algoritmos para equalização cega foram propostos nos últimos anos, indicando que esta área desperta grande interesse dos pesquisadores. Entre estas novas técnicas podemos citar: algoritmo de módulo constante generalizado (GCMA - *Generalized Constant Modulus Algorithm*, Cavalcanti *et al.*, 1998); algoritmos do tipo super-exponencial (Shalvi e Weinstein, 1993; Regalia e Mboup, 1999); equalização concorrente (De Casto *et al.*, 2001; Chen, 2003). Ainda mais recentemente, Chung e You (2007) propuseram um algoritmo chamado de “Run-and-Go” que apresenta uma taxa de convergência muito superior ao do algoritmo “Stop-and-Go”. Podemos, é claro, incluir nesta lista (obviamente incompleta) a nossa própria proposta de equalização cega usando um filtro de erro de predição largamente linear (Aquino *et al.*, 2007a).

### C.1.7 Modo de decisão direta (DD)

Após a convergência inicial (se houver), quando o “olho” já está aberto e a probabilidade de acerto do símbolo decidido é alta, os algoritmos cegos podem entrar no modo de decisão direta. A passagem para o modo DD pode ser realizada tendo por base, por exemplo, o valor médio do módulo do erro DD nas últimas iterações:

$$V_{DDm}[k] = \alpha V_{DDm}[k-1] + (1-\alpha) |\varepsilon_{DD}[k]|, \quad (\text{C.14})$$

com  $0 \ll \alpha < 1$ ,  $V_{DDm}[0] = \max\{a[k]\}$ . Quando  $V_{DDm}[k]$  for menor que um certo valor *limiar*, por exemplo,  $1/3 \min\{a[k]\}$ . Esta passagem também pode ser *suave*, isto é, o erro pode ser uma soma ponderada do erro DD e do erro cego, isto é,  $\varepsilon_{comb}[k] = \alpha_1 \varepsilon_{DD}[k] + (1-\alpha_1) \varepsilon_{cego}[k]$ . À medida que vai ocorrendo a convergência, o erro DD vai tendo maior peso ( $\alpha_1 \rightarrow 1$ ).

A título ilustrativo, refizemos a última simulação da Seção 4.8.3 incluindo (C.14), com  $\alpha = 0,985$ , usando *limiar* = 0,4 e mantendo os demais parâmetros, com exceção do passo de adaptação que é aumentado em 10 vezes após o algoritmo entrar no modo DD. A Figura C.2 mostra os diagramas de olho (uma realização) para os algoritmos, enquanto a Figura C.3 mostra a média de 300 realizações das curvas de convergência. É fácil perceber que o erro residual é bem inferior aos resultados obtidos no Capítulo 4.

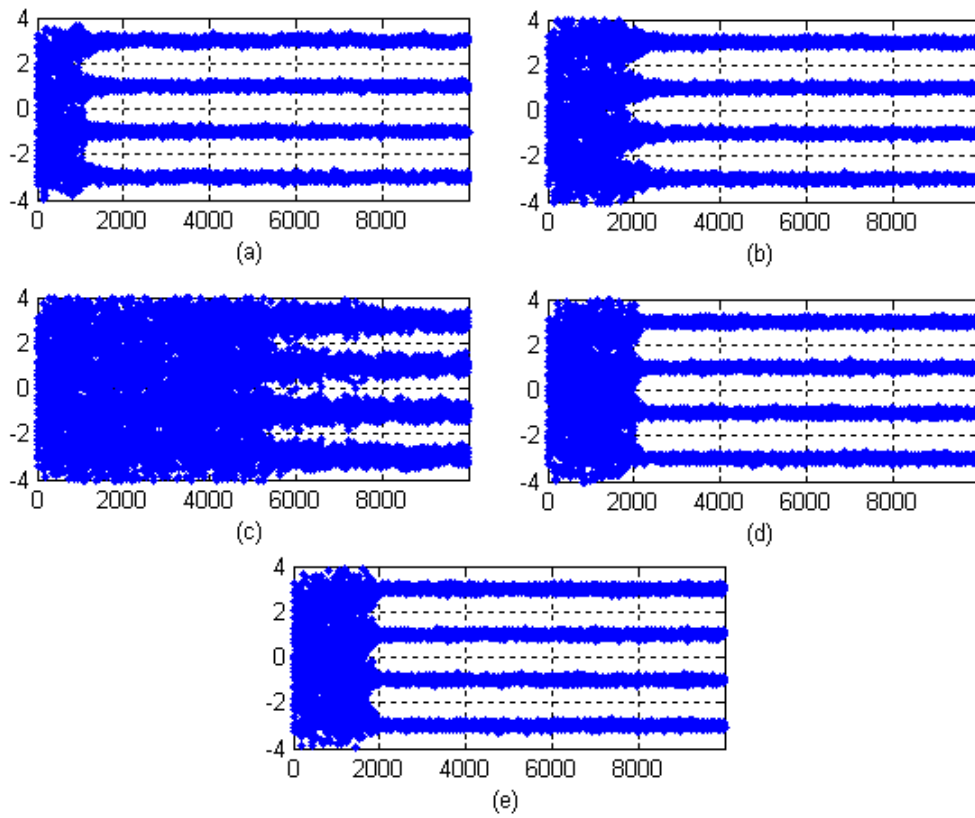


Figura C.2. Diagrama de olho de uma realização para as diversas técnicas cegas + DD para o canal  $\mathbf{h}_m$  e uma relação sinal-ruído de 30 dB: (a) FEPLL, (b) CMA, (c) FEPL, (d) Sato, (e) Benveniste-Goursat.

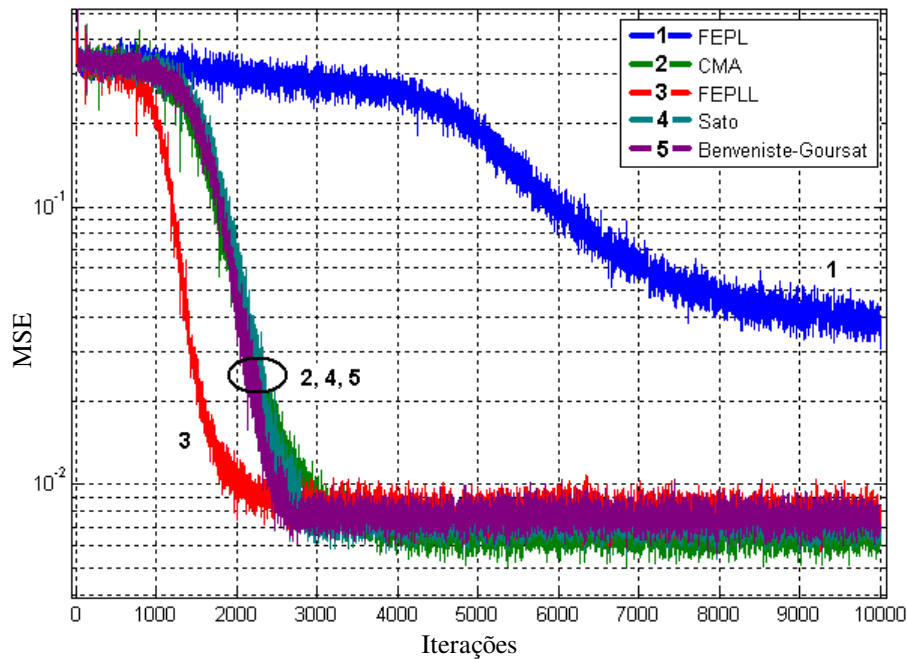


Figura C.3. Comparação dos algoritmos cegos: MSE do erro de decisão direta.

## C.2 Comparação das soluções ZF e MSE para o FEPLL

Nesta seção apresentamos um exemplo numérico com o objetivo de ilustrar os resultados teóricos da Seção 4.3. Para isto, consideramos o canal de fase não-mínima cuja resposta ao impulso é dada por  $\mathbf{h} = [0,5972 + 0,1493j; 0,2687 - 0,4404j; 0,0155 - 0,5545j; -0,1047 - 0,1903j]$ . Este canal pode ser equalizado, a menos de um ganho complexo, por um FEPLL com apenas 6 coeficientes (3 no filtro  $\mathbf{f}$  e outros 3 coeficientes no filtro  $\mathbf{g}$ ).

A solução ZF (canal sem ruído), usando a equação (4.19), resulta nos seguintes filtros  $\mathbf{f}$  e  $\mathbf{g}$ :

$$\mathbf{f}_{\text{ZF}} = \begin{bmatrix} 0,000 - 0,6077j \\ 0,349 - 0,2763j \\ 0,0666 - 0,1789j \end{bmatrix}, \quad \mathbf{g}_{\text{ZF}} = \begin{bmatrix} 0,3111 - 0,0521j \\ 0,2319 - 0,5606j \\ -0,1155 - 0,1520j \end{bmatrix}, \quad (\text{C.15})$$

enquanto que a solução MSE usando as equações (4.15) a (4.17), para uma relação sinal ruído de 30 dB, é dada por:

$$\mathbf{f}_{\text{MSE}} = \begin{bmatrix} 0,1935 - 0,5454j \\ 0,0938 - 0,1438j \\ -0,0246 - 0,0492j \end{bmatrix}, \quad \mathbf{g}_{\text{MSE}} = \begin{bmatrix} 0,1655 - 0,1958j \\ 0,0948 - 0,4231j \\ -0,1288 - 0,0030j \end{bmatrix}, \quad (\text{C.16})$$

que é um resultado claramente diferente da solução ZF. Entretanto, quando a relação sinal-ruído tende ao infinito, verificamos que  $\mathbf{f}_{\text{MSE}} \rightarrow \mathbf{f}_{\text{ZF}}$ . Por exemplo, para uma relação sinal-ruído de 60 dB, encontramos:

$$\mathbf{f}_{\text{MSE}} = \begin{bmatrix} 0,0007 - 0,6075j \\ 0,1348 - 0,2757j \\ 0,0661 - 0,1784j \end{bmatrix}, \quad \mathbf{g}_{\text{MSE}} = \begin{bmatrix} 0,3105 - 0,0526j \\ 0,2314 - 0,5602j \\ -0,1156 - 0,1513j \end{bmatrix}. \quad (\text{C.17})$$

Neste exemplo, comparando (C.15), (C.16) e (C.17), é fácil percebermos a “evolução” da solução MSE em direção à solução ZF. As figuras C.4 e C.5 mostram essa evolução de forma mais detalhada, pois fazemos a relação sinal-ruído variar de 0 (canal muito ruidoso) a 80 dB (canal quase sem ruído) com um passo de apenas 0,5 dB. Na Figura C.4 temos os valores assumidos pelo filtro  $\mathbf{f}_{\text{MSE}}$ , enquanto que na Figura C.5 temos os valores que os coeficientes do filtro  $\mathbf{g}_{\text{MSE}}$  atingem. Para uma relação sinal-ruído de 80 dB, as soluções ZF e MSE praticamente se confundem.



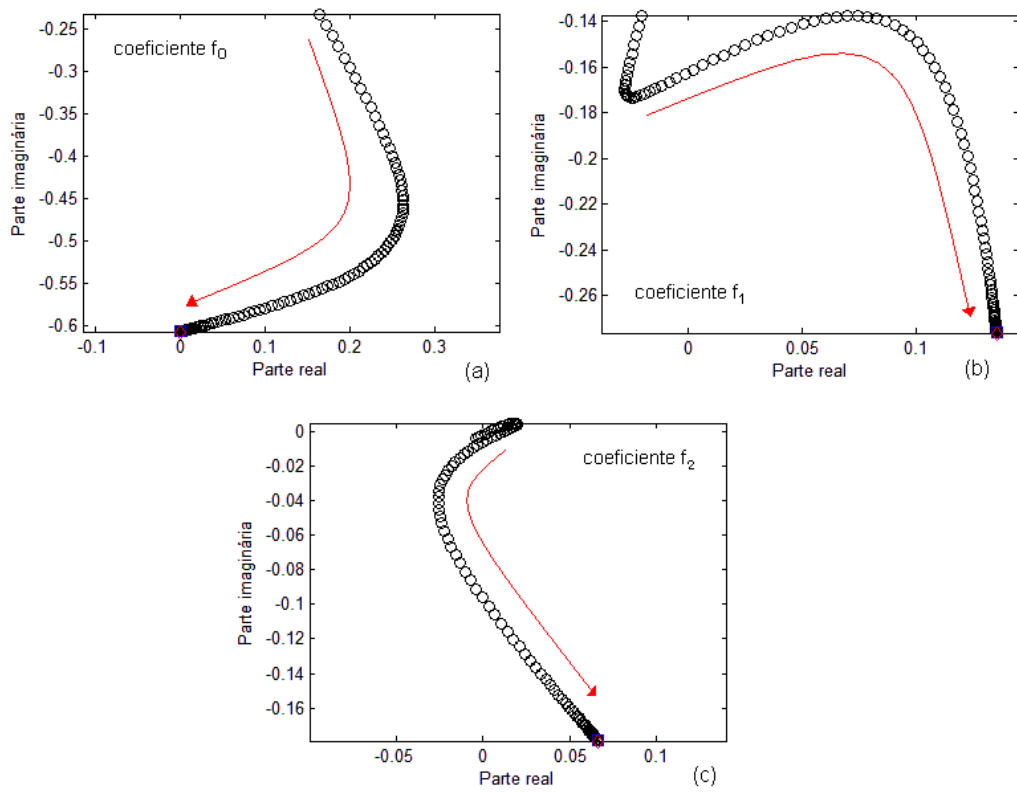


Figura C.4: Evolução dos coeficientes do filtro  $f_{MSE}$  em função da relação sinal-ruído (de 0 a 80 dB), (a) coeficiente  $f_0$ , (b) coeficiente  $f_1$  e (c) coeficiente  $f_2$ .

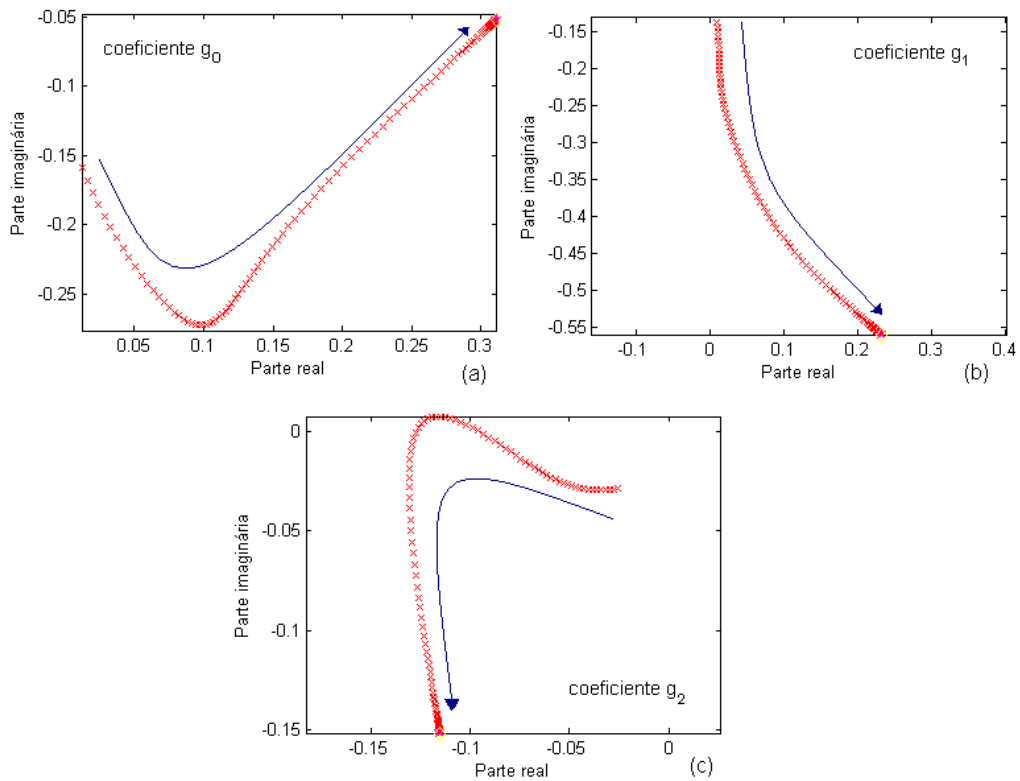


Figura C.4: Evolução dos coeficientes do filtro  $g_{MSE}$  em função da relação sinal-ruído (de 0 a 80 dB), (a) coeficiente  $g_0$ , (b) coeficiente  $g_1$  e (c) coeficiente  $g_2$ .

### C.3 Detalhamento da implementação do FEPLL com passo fixo

Nesta seção apresentamos alguns detalhes da implementação do FEPLL já descrita na Seção 4.7. Lembramos que os sinais complexos são separados em suas partes em fase e quadratura antes de iniciar o processamento. Desta forma, uma multiplicação complexa é efetuada com o uso de quatro multiplicações reais e mais duas somas reais.

A Figura C.6 mostra o detalhamento do circuito para a realização de cada um dos coeficientes do FEPLL. Observar que o sinal é separado nas suas parte real e imaginária (“xI” e “xQ”) e que a multiplicação é realizada após um atraso de 3 amostras (“delay1” e “delay4”) para haver sincronização entre o sinal de entrada e o sinal de erro. Este atraso é necessário devido à latência dos blocos multiplicadores. Por isso, o algoritmo implementado é do tipo LMS com atraso (DLMS – Delayed LMS). É fácil perceber que este bloco FPGA tem quatro entradas (“xI”, “xQ”, “ $\mu$ .erroI”, “ $\mu$ .erroQ”) e quatro saídas (“xI” e “xQ” atrasados, “wI.xI” e “wQ.xQ”). Na Figura C.7 vemos o conjunto de cinco destes blocos (“tap”) que correspondem ao cálculo da parte real do filtro W. Na Figura C.8 temos os circuitos para cálculo da correção de fase e para decisão do símbolo após a correção de fase e de ganho. A Figura C.9 mostra a estimativa de recursos desta implementação.

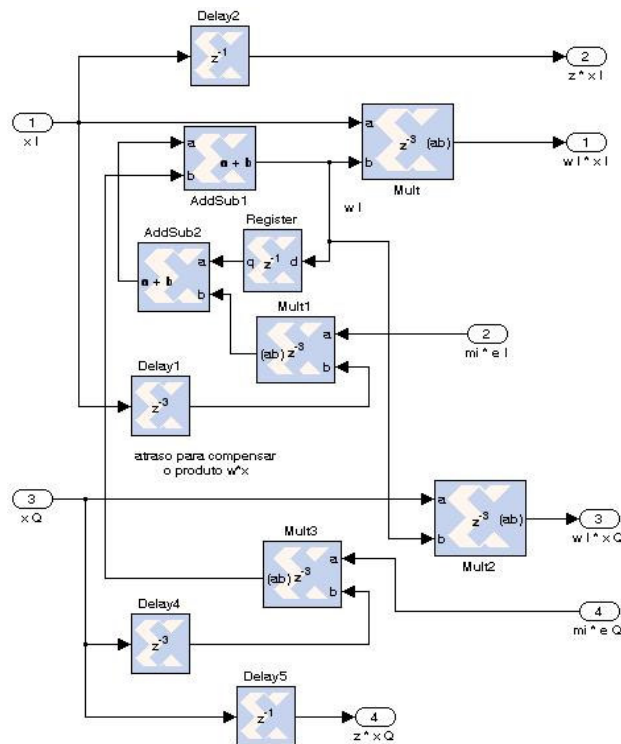


Figura C.6: Detalhamento do cálculo de um coeficiente em FPGA do FEPLL (“tap”).

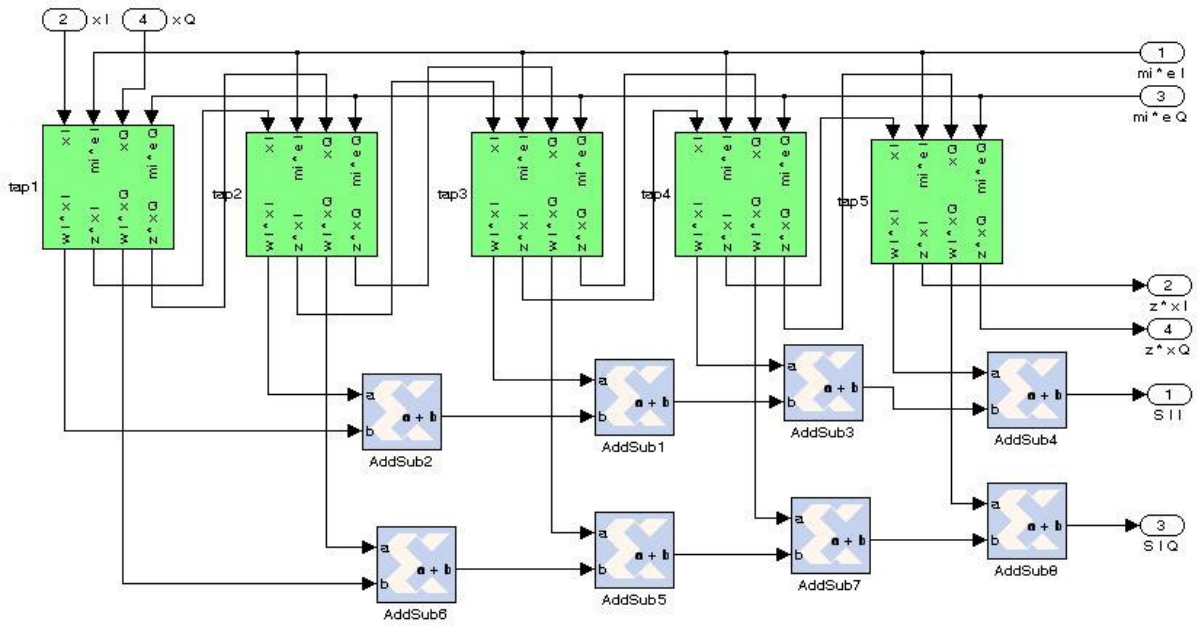


Figura C.7: Conjunto de “taps” e somadores para cálculo parcial da saída do FEPLL.

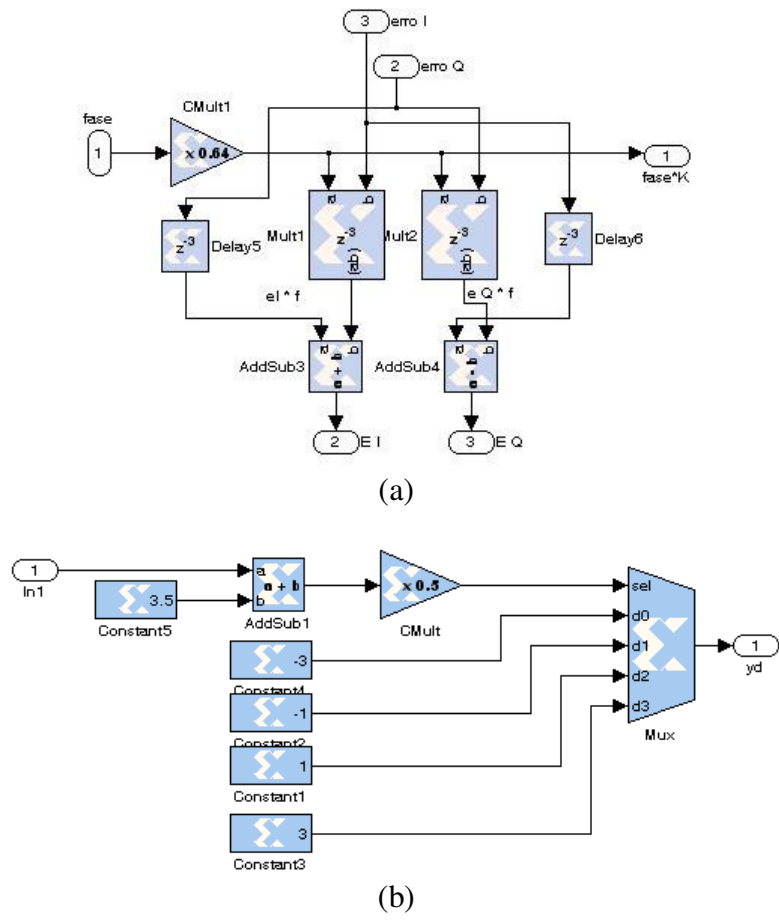


Figura C.8: (a) correção de fase e (b) cálculo do símbolo decidido em FPGA.

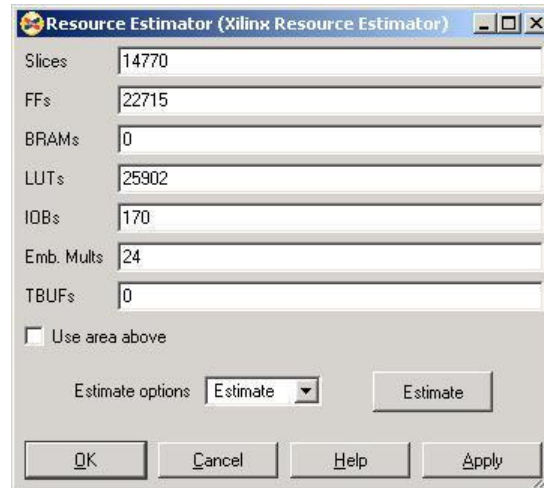


Figura C.9: Estimação de área do FEPLL.

Finalmente, na Figura C.10 temos um exemplo de equalização usando o FEPLL implementado em FPGA. Podemos observar (ver Figura C.10-b) que na correção de ganho ocorre uma saturação do fator de correção do ganho devido à implementação escolhida (somador sem sinal e com apenas um bit após o ponto decimal). Nesta simulação consideramos uma relação sinal ruído de 30 dB, o canal de fase não-mínima e com um zero próximo ao CRU definido por  $\mathbf{h} = [0,5432 + 0,1207j; 0,0805 - 0,2012j; -0,2314 - 0,6840j; -0,2615 - 0,2314j]$  e um sinal pertencente a uma constelação 4-PAM. O passo de adaptação  $\mu = 2^{-13} + 2^{-16}$ , o total de bits usados nos somadores e multiplicadores foi de 22 bits, sendo 16 bits para a parte fracionária.

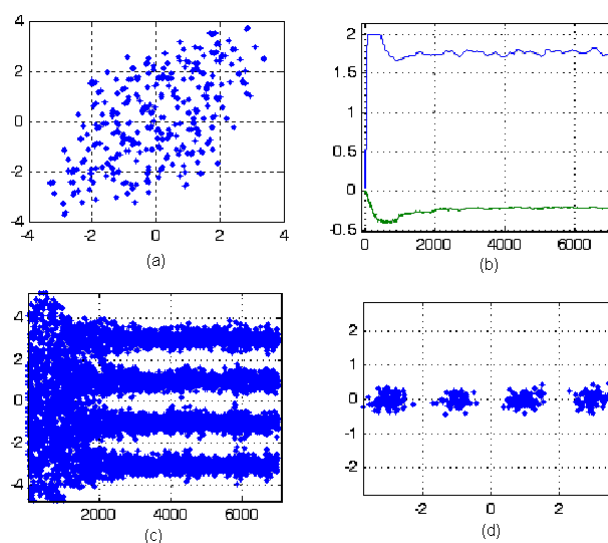


Figura C.10: Equalização com o FEPLL implementado em FPGA. (a) Sinal após o canal, (b) ajuste de ganho e de fase, (c) evolução da constelação, (d) diagrama de olho após a convergência. Relação sinal-ruído de 30 dB.

## Referências Bibliográficas

- Abed-Meraim, K., Moulines, E., e Loubaton, P., “Prediction Error Method for Second-Order Blind Identification”, *IEEE Trans. Signal Processing*, vol. 45, no. 3, pp. 694-705, Mar. 1997.
- Alamouti, S. M., “A simple transmit diversity technique for wireless communications”, *IEEE J. Select. Areas Commun.*, pp. 1451–1458, Oct. 1998.
- Al-Dhahir N., e Cioffi, J. M., “MMSE decision-feedback equalizers: finite length results”, *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 961–975, Jul. 1995.
- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Accelerating the Convergence of the Widely Linear LMS Algorithm for Channel Equalization,” *VI International Telecommunications Symposium (ITS2006)*, Fortaleza-CE, Brazil, vol. 1, pp. 57-61, setembro de 2006a.
- Aquino, Francisco J.A, Vargas, Cesar H.V. e Rocha, Carlos A.F., “A FPGA Implementation of the Widely Linear LMS Adaptive Filter for Channel Equalization”, *Proceedings of the 5th International Information and Telecommunication Technologies Symposium (I2TS2006)*, Cuiabá-MT, Brazil, vol. 1, dezembro de 2006b.
- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Widely Linear Prediction for Blind Equalization”, *IEEE International Conference on Communications - ICC - Glasgow, Scotland*, vol. 1, pp. 1-6, junho de 2007a.
- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Equalização de Canal Usando um Algoritmo LMS Largamente Linear Multi Split”, *XXV Simpósio Brasileiro de Telecomunicações (SBrT’07)*, Recife, Pernambuco, pp. 1-6, setembro de 2007b.
- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Equalização Cega via um Banco de Filtros de Erro de Predição Largamente Lineares com Atraso de Equalização”, *XXV Simpósio Brasileiro de Telecomunicações (SBrT’07)*, Recife, Pernambuco, pp. 1-6, setembro de 2007c.
- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Multi-Split Widely Linear LMS Algorithm for Channel Equalization”, *XXI Colloque GRETSI*, Troyes – France, setembro de 2007d.

- Aquino, Francisco J. A., Rocha, C. A. F., e Resende, Leonardo S., “Equalização Cega Usando Filtro de Erro de Predição Largamente Linear de Passo Variável”, *Proceedings of the 6th International Information and Telecommunication Technologies Symposium (I2TS2007)*, Brasília – Distrito Federal, dezembro de 2007e.
- Aquino, F. J. A. de, Rocha, C. A. F. da, Resende, Leonardo S., “Análise Estatística do Algoritmo LMS Largamente Linear e Cálculo do Atraso Ótimo para Equalização Largamente Linear”. Artigo aceito para publicação no *XXV Simpósio Brasileiro de Telecomunicações (SBrT’08)*, Rio de Janeiro – RJ, setembro de 2008.
- Attux, R. R. de Faissol, “Novos Paradigmas Para Equalização e Identificação de Canais Baseados em Estruturas Não-Lineares e Algoritmos Evolutivos”, Campinas, 2005. Tese de Doutorado, Universidade Estadual de Campinas – Unicamp – Faculdade de Engenharia Elétrica e de Computação, Departamento de Comunicações.
- Barbosa, L.M.J., Mota, J.C.M., Cavalcanti, F.R.P., “A Family of Predictive Constant Modulus Algorithms for Blind Equalization”, *IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium, AS-SPCC*, pp: 260–265, 2000.
- Belline, S., “Busgang Techniques for Blind Deconvolution and Equalization”, em *Blind Deconvolution* (S. Haykin, editor), pp.: 8-59, Prentice Hall, Englewood Cliffs, New Jersey, 1994.
- Benedetto, S., Biglieri, E., Castellani, V., “Digital Transmission Theory”, Prentice-Hall International, Inc., New Jersey, 1987.
- Benveniste, A., e Goursat, M., “Blind equalizers”, *IEEE Transactions on Communications*, vol. 32, no. 8, pp. 871–883, Agosto 1984.
- Benveniste, A., Goursat, M., e Ruget, G., “Robust identification of a nonminimum phase system: Blind adjustment of a linear equalizer in data communications”, *IEEE Transactions on Automatic Control*, vol. 25, no. 3, pp. 385–399, Junho 1980.
- Blackard, K.L.; Rappaport, T.S.; Bostian, C.W., “Measurements and models of radio frequency impulsive noise for indoor wireless communications”, *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 7, pp.: 991 - 1001, setembro 1993.
- Bordin Jr., C.J., “Métodos Estatísticos para a Equalização de Canais de Comunicação”, São Paulo, 2006. Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Telecomunicações e Controle.

- Brown, W. M. e Crane, R. B., “Conjugate linear filtering,” *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 462-465, 1969.
- Carlson, A. Bruce, “Communication Systems – An Introduction to Signals and Noise in Electrical Communication”, McGraw-Hill International Editions, 3a. Edição, 1986.
- Chang, Bruno S., “Implementação em FPGA de Técnicas de Equalização Adaptativa Utilizando o Algoritmo CORDIC”, dissertação de mestrado, Departamento de Engenharia Elétrica, UFSC, 2008.
- Chen, S., “Low complexity concurrent constant modulus algorithm and soft decision directed scheme for blind equalization”, *IEE Proceedings - Vision, Image and Signal Processing*, vol. 150, no. 5, pp:312-320, Oct. 2003.
- Chevalier, P. e Blin, A., “Widely Linear MVDR Beamformers for the Reception of an Unknown Signal Corrupted by Noncircular Interferences”, *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5323 – 5336, Nov. 2007.
- Chevalier, P. e Pison, F., “New insights into optimal widely linear array receivers for demodulation of BPSK,MSK, and GMSK corrupted by noncircular interferers - application to SAIC”, *IEEE Trans. Signal Processing*, vol. 54, pp. 870–883, Mar. 2006.
- Chung, W. e You, C., “Fast Recovery Blind Equalization for Time-Varying Channels Using ‘Run-and-Go’ Approach”, *IEEE Transactions on Broadcasting*, vol. 53, no. 3, pp. 693-696, Set. 2007.
- Costa, M.H., e Bermudez, J.C.M., “An Improved Model for the Normalized LMS Algorithm with Gaussian Inputs and Large Number of Coefficients”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 1385-1388, maio 2002.
- Cover, T.M., Thomas, J.A., “Elements of Information Theory”, John Wiley and Sons, Inc., USA, New York, 1991.
- De Castro, F.C.C., De Castro, M.C.F. e Arantes, D.S., “Concurrent Blind Deconvolution for Channel Equalization”, *IEEE International Conference on Communications - ICC 2001*, Helsinque, vol. 2, pp: 366-371, junho 2001.
- Delsarte, P., e Genin, Y.V., “The Split Levinson Algorithm”. *IEEE Trans. on Acoust., Speech and Signal Processing*, vol. ASSP-34, no. 3, pp. 470-478, junho 1986.

- Deshpande, N., “Fast recovery equalization techniques for DTV signals”, *IEEE Transactions on Broadcasting*, vol. 43, no. 4, pp. 370–377, dezembro 1997.
- Dietl, G., Mensing, C., e Utschick, W., “Iterative Detection Based on Widely Linear Processing and Real-Valued Symbol Alphabets”. In *Proceedings of the Eleventh European Wireless Conference 2005*, vol. 1, pp. 74-78, abril 2005.
- Ding, Z., Li, Ye (Geoffrey), “Blind Equalization and identification”, Marcel Dekker, Inc., New York, 2001.
- Endres, T.J., Halford, S.D., Johnson, C.R., Giannakis, G.B., “Blind Adaptive Channel Equalization Using Fractionally-Spaced Receivers: A Comparison Study”, *Proc. Conference on Information Sciences and Systems*, Princeton, N.J., março 1996.
- Gelli, G., Paura, L. e Ragozini, A. R. P., “Blind widely linear multiuser detection”, *IEEE Commun. Lett.*, vol. 4, no. 6, pp. 187–189, Jun. 2000.
- Gerstacker, W. H., Obernosterer, F., Schober, R., Lehmann, A., Lampe, A., e Gunreben, P., “Equalization Concepts for Alamouti’s Space–Time Block Code”, in *IEEE Trans. on Communication*, vol. 52, no. 7, pp. 1178–1190, julho 2004.
- Gerstacker, W. H., Schober, R., e Lampe, A., “Receivers with widely linear processing for frequency-selective channels”, *IEEE Trans. on Communication*, vol. 51, no. 9, pp. 1512–1523, setembro 2003.
- Gesbert, D., Shafi, M., Shiu, D., Smith, P., e Naguib, A., “From Theory to Practice: An Overview of MIMO Space-Time Coded Wireless Systems”, *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 3, pp. 281-302, abril 2003.
- Giannakis, G.B., Halford, S.D., “Blind fractionally spaced equalization of noisy FIR channels: direct and adaptive solutions”, *IEEE Trans. on Signal Processing*, vol. 45, pp. 2277 - 2292, setembro 1997.
- Godara, L. C., “Smart Antennas” , CRC Press LLC, Boca Raton, Florida, 2004.
- Godard, Dominique N., “Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems”, *IEEE Trans. on Communication*, vol. 28, pp. 1867 - 1875, novembro 1980.
- Goldsmith, A., “Wireless Communications”, Cambridge University Press, 2005.
- Goupil, A., Palicot, J., “Quasi-Newton formulation and analysis of split LMS”, *14th International Conference on Digital Signal Processing - DSP 2002*, Vol. 2, pp. 753-756, julho 2002.



- Guidoux, L., e Macchi, O., “Un nouvel égaliseur — L'égaliseur à double échantillonnage”, *Ann. Telecommun.*, vol. 30, nos. 9/10, pp. 331–338, 1975.
- Haimi-Cohen, R., Herzberg, H. e Be'ery, Y., “Delayed Adaptive LMS Filtering: Current Results”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1273 - 1276, abril 1990.
- Harikumar, G. e Bresler, Y. “FIR perfect signal reconstruction from multiple convolutions: Minimum deconvolver orders”, *IEEE Transactions Signal Processing*, vol. 46, pp. 215–218, janeiro 1998.
- Hatzinakos, D., “Blind equalization using stop-and-go adaptation rules”, *Optical Engineering*, vol. 31, no. 6, pp. 1181–1188, junho 1992.
- Hauck, S., “The Roles of FPGAs in Reconfigurable Systems,” *Proceedings of the IEEE*, vol. 86, no. 4, pp. 615-638, Abril 1998.
- Haykin, S., “Adaptive Filter Theory”, 4<sup>th</sup>. ed., Prentice Hall, New Jersey, 2002.
- Haykin, S., “Modern Filters”, Macmillan Publishing Company, New York, 1990.
- Hwang, J., Milne, B., Shirazi, N. e Stroemer, J., “System Level Tools for DSP in FPGAs”, Xilinx Inc. 2001.
- Jeon, J.-J., Andrews, Jeffrey G. e Sung, Koeng-Mo, “The Blind Widely Linear Minimum Output Energy Algorithm for DS-SS Systems”, *IEEE Transactions on Signal Processing*, vol. 54, no. 5, pp. 1926 - 1931, maio 2006.
- Kim, Y., Kim, S., e Kim, M., “The derivation of a new blind equalization algorithm”, *ETRI Journal*, vol. 18, no. 2, pp. 53–60, julho 1996.
- Kuchi, K., Mattellini, G.P., Prabhu, V.K., “Generalization of Widely Linear Filtering Concepts for Equalization and Interference Suppression in PAM/QAM Systems”, *Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, pp. 1287 – 1291, 28 outubro – 1 novembro, 2005.
- Kuchi, K.K., “Generalization of Widely Linear Filtering Concepts for Equalization and Interference Suppression in PAM/QAM Systems”, Arlington, 2006. Tese de Doutorado em Engenharia Elétrica, Faculty of the Graduate School of The University of Texas, USA.
- Kuruoglu, E.E., Fitzgerald, W.J. e Rayner, P.J.W., “Near optimal detection of signals in impulsive noise modeled with a symmetric  $\alpha$ -stable distribution”, *IEEE Communications Letters*, vol. 2, no. 10, pp. 282 – 284, outubro 1998.

- Kwong, R.H. e Johnston, E.W., “A Variable Step-Size LMS Algorithm”, *IEEE Trans. on Signal Processing*, vol. 40, no. 7, pp. 1633-1642, julho 1992.
- Lim, J., Han, K.-Y. e Jeon, J., “Adaptive step-size widely linear linearly constrained constant modulus algorithm for DS-CDMA receivers in nonstationary interference environments”, in *Signal Processing*, vol. 87, pp. 1523-1527, Jun. 2007. Disponível no endereço eletrônico: <http://dx.doi.org/10.1016/j.sigpro.2006.12.003>
- Lin, J.-C., “Blind equalisation technique based on an improved constant modulus adaptive algorithm”, *IEE Proc. Commun.*, vol. 149, no. 1, pp. 45-50, fevereiro 2002.
- Long, G., Ling F., e Proakis, J.G., “The LMS Algorithm with Delayed Coefficient Adaptation”, in *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, no. 9, pp. 1397-1405, setembro 1989.
- Lucky, R.W., “Automatic Equalization for Digital Communication”, *Bell System Technical Journal*, vol. 44, pp. 547-588, abril, 1965.
- Lysaght, P e Dunlop, J., “Dynamic Reconfiguration of Field Programmable Gate Arrays”. *Proceedings of the 3rd International Workshop on Field Programmable Logic and Applications (FPL'93)*, Oxford, UK Abingdon EE&CS Books, p. 82-94, 1993.
- Manolakis, D. G., Ingle, V. L. e Kogon, S. M., “Statistical and Adaptive Signal Processing”, McGraw Hill, 2000.
- Mathis, H., “Nonlinear Functions for Blind Separation and Equalization”, Zurique, 2001. Tese de Doutorado em Ciências Técnicas, Instituto Federal de Tecnologia da Suíça.
- Mattera, D., Paura, L. e Sterle, F., "Widely Linear Decision-Feedback Equalizer for Time-Dispersive Linear MIMO Channels", *IEEE Trans. Signal Processing*, vol 53, no. 7, pp. 2525-2536, julho 2005.
- Mattera, D., Paura, L. e Sterle, F., “Widely linear MMSE equaliser for MIMO linear time-dispersive channel,” *Electron. Lett.*, vol. 39, pp. 1481–1482, outubro 2003.
- McWhorter, T., e Schreier, P.; “Widely-Linear Beamforming”, *IEEE Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 753-759, novembro 2003.
- Mosen, P., “Feedback equalization for fading dispersive channels”, *IEEE Transactions on Information Theory*, vol. 17, no. 1, pp. 56–64, janeiro 1971.

- Nagumo, J.I. e Noda, A., “A learning method for system identification”, *IEEE Trans. Automat. Control*, vol AC-12, pp.: 282–287, junho 1967.
- Neeser, F. D. e Massey, J. L. “Proper complex random processes with applications to information theory,” *IEEE Trans. Inform. Theory*, vol. 39, pp. 1293–1302, julho 1993.
- Neves, A.O., Attux, R. R.F., Suyama, R., Miranda, M.D. e Romano, J.M.T., “Sobre Critérios Para Equalização Não-Supervisionada”, *Revista Controle & Automação*, vol. 17, no. 3, pp. 278-299, julho, agosto e setembro de 2006.
- Okwit, S., “An Historical View of the Evolution of Low-Noise Concepts and Techniques”, *IEEE Transactions On Microwave Theory And Techniques*, vol. Mtt-32, no. 9, setembro 1984.
- Papadias, C.B. e Slock, D.T.M., “Fractionally spaced equalization of linear polyphase channels and related blind techniques based on multichannel linear prediction”, *IEEE Trans. on Signal Processing*, vol. 47, pp. 641-654, março 1999.
- Picinbono, B. e Bondon, P., “Second-order statistics of complex signals”, *IEEE Trans. Signal Processing*, vol. 45, pp. 411-420, fevereiro 1997.
- Picinbono, B. e Chevalier, P., “Widely linear estimation with complex data”, *IEEE Trans. Signal Processing*, vol. 43, pp. 2030–2033, agosto 1995.
- Picinbono, B., “On circularity”, *IEEE Trans. Signal Processing*, vol. 42, pp. 3473-3482, dezembro 1994.
- Picinbono, B., “Wide-sense linear mean square estimation and prediction,” *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-95*, vol. 3, pp. 2032-2035, maio 1995.
- Porat, B. e Friedlander, B., “Blind Equalization of Digital Communication Channels Using High-Order Moments”, *IEEE Trans. on Signal Processing*, vol. 39, no. 2, pp. 522-526, fevereiro 1991.
- Proakis, J. G., “Digital Communications”, McGraw-Hill, 3a. Edição, 1995.
- Qureshi, S.U.H., “Adaptative Equalization”, *Proc. IEEE*, vol. 53, pp. 1349-1387, Set., 1985.
- Rappaport, T.S.; “Wireless Communication: Principles and Practice”, Prentice Hall, Upper Saddle River, NJ, 1996.
- Reed, I. S., “On a moment theorem for complex Gaussian processes”, *IRE Trans. Inf. Theory*, IT-8, pp. 194–195, Abril, 1962.

- Reed, J. H., “Software Radio: A Modern Approach to Radio Engineering”, Prentice Hall, Upper Saddle River, NJ, 2002.
- Resende, L. S., Romano, J. M. T., e Bellanger, M. G., “Split Wiener filtering with application in adaptive systems”. *IEEE Transactions on Signal Processing*; vol. 52; no. 3; pp. 636-644; Março 2004.
- Resende, L. S., Romano, J. M. T. e Bellanger, M., “Multi-Split Adaptive Filtering”, *IEEE Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2001. ICASSP 2001, vol. 6, pp. 3729-3732, maio 2001.
- Rocha, C. A. F., “Técnicas Preditivas para Equalização Autodidata”, Campinas, 1996. Tese de Doutorado, Universidade Estadual de Campinas – Unicamp – Faculdade de Engenharia Elétrica e de Computação, Departamento de Comunicações.
- Sayed, A.H., “Fundamentals of Adaptive Filtering”, John Wiley & Sons, NY, 2003.
- Schober, R., Gerstacker, W.H., Lampe, L. H.-J., “Data-aided and blind stochastic gradient algorithms for widely linear MMSE MAI suppression for DS-CDMA”, *IEEE Transactions on Signal Processing*, vol. 52, pp. 746-756, março 2004.
- Schreier, P.J. e Scharf, L.L., “Second-Order Analysis of Improper Complex Random Vectors and Processes”, *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 714-725, março 2003.
- Schreier, P.J., Scharf, L.L. e Mullis, C.T., “A Unified Approach to Performance Comparisons Between Linear and Widely Linear Processing”, *IEEE Workshop on Statistical Signal Processing*, pp.: 114 – 117, 28 setembro - 1 outubro 2003.
- Shalvi, O., e Weinstein, E., “New criteria for blind deconvolution of nonminimum phase systems (channels)”, *IEEE Transactions on Information Theory*, vol. 36, pp. 312-321, março 1990.
- Silva, M. T. M., “Um Estudo Sobre Técnicas de Equalização Autodidata”, São Paulo, 2005. Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Telecomunicações e Controle.
- Sklar, B., “A structured overview of digital communications — A tutorial review, Part I”, *IEEE Commun. Mag.*, no. 21, pp. 4–17, agosto, 1983.
- Slock, D.T.M., “Blind fractionally spaced equalization based on cyclostationarity and second-order statistics”, in *Proc. ATHOS (ESPRIT Basic Research Working Group*

- 6620) *Workshop Syst. Ident. High Order Stat.*, Sophia Antipolis, France, setembro 1993.
- Slock, D.T.M., “Blind fractionally-spaced equalization, perfect-reconstruction filterbanks and multichannel linear prediction,” *Proc. ICASSP*, vol. 4, pp. 585–588, 1994.
  - Souza, R. D., Resende, L. S., Rocha, C. A. F. da, e Bellanger, M. G., “On Split FIR Filtering in Blind Equalization”, *IEEE International Conference on Communications, ICC 2002*, vol. 3, pp. 1918-1922, maio 2002.
  - Souza, R.D., Resende, L.S. e Bellanger, M.G., “Efficient implementation of multi-split LMS filtering with complex parameters”, *IEEE Proc. of the 4th International Symposium on Image and Signal Processing and Analysis, ISPA 2005*, pp. 419-423, setembro 2005.
  - Sterle, F., “The Exploitation of Widely Linear Filtering in MIMO Communication Systems”, Nápolis, 2005. Tese de doutorado, Università degli Studi di Napoli “Federico II”, Departamento de Engenharia Eletrônica e de Telecomunicações.
  - Stockham Jr., T., Cannon, T.M. e Ingebretsen, R.B., “Blind deconvolution through digital signal processing”, *Proceedings of the IEEE*, vol. 63, no. 4, pp. 678–692, abril 1975.
  - Tarokh, V., Alamouti, S.M. e Poon, P., “New Detection Schemes for Transmit Diversity with no Channel Estimation”, *IEEE International Conference on Universal Personal Communications, ICUPC’98*, vol. 2, pp. 917-920, outubro 1998.
  - Therrien, C. W., “Discrete Random Signals and Statistical Signal Processing”, Prentice-Hall International Editions, New Jersey, 1992.
  - Tidestav, C., Sternad, M. e Ahlén, A. “Reuse within a cell: Interference rejection or multiuser detection?”, *IEEE Trans. Commun.*, vol. 47, pp. 1511–1522, outubro 1999.
  - Tobias, O.J., Bermudez, J.C.M. e Bershada, N.J., “Stochastic analysis of the delayed LMS algorithm for a new model”, *IEEE Proc. of the International Conference on Acoustics, Speech, and Signal Processing, ICASSP’00*, vol. 1, pp.:404-407, Jun. 2000.
  - Tong, L., Xu, G. e Kailath, T., “Fast blind equalization via antenna arrays”, *IEEE Proc. of the International Conference on Acoustics, Speech, and Signal Processing, ICASSP-1993*, vol. 4, pp: 272-275, abril 1993.
  - Tu, J. C., “Optimum MMSE Equalization for Staggered Modulation”, *IEEE Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1401-1406, novembro 1993.

- Tugnait, J. K. e Huang, B., “Multi-Step Linear Predictors-Based Blind Equalization of Multiple-Input Multiple-Output Channels”, *IEEE Proc. of the International Conference on Acoustics, Speech, and Signal Processing, ICASSP’99*, vol. 5, pp. 2949-2952, março 1999.
- Ungerboeck, G. “Fractional tap-spacing equalizer and consequences for clock recovery in data modems”, *IEEE Trans. Commun.*, vol. 24, pp. 856–864, agosto 1976.
- Varanasi, M. K., “Decision feedback multiuser detection: a systematic approach”, *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 219–240, janeiro 1999.
- Vargas, Cesar H. V., “Técnicas de Equalização Adaptativa Aplicadas aos Modelos de Canal Sui”, dissertação de mestrado, Departamento de Engenharia Elétrica, Universidade Federal de Santa Catarina, 2007.
- Vargas, C. H. V., Aquino, F. J. A. e Rocha, C. A. F., “Implementação e Comparação das Estruturas Paralela, Semi-Paralela e Seqüencial de um Equalizador Largamente Linear em FPGA”, *XXV Simpósio Brasileiro de Telecomunicações – SBrT’07*, pp. 1-6, setembro, 2007.
- Voois, P. A., Lee, I. e Cioffi, J. M., “The effect of decision delay in finite-length decision feedback equalization”, *IEEE Trans. Inform. Theory*, vol. 42, pp. 618-621, março 1996.
- Wan, K. F., e Ching, P.C., “Multilevel split-path adaptive filtering and its unification with discrete walsh transform adaptation,” *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 147–151, fevereiro 1992.
- Weerackody, V. e Kassam, S.A., “Dual-mode type algorithms for blind equalization,” *IEEE Transactions on Communications*, vol. 42, no. 1, pp. 22–28, janeiro 1994.
- XILINX Inc. Achieving Breakthrough Performance in Virtex-4 FPGAs. White Paper: Virtex-4 FPGAs. Maio, 2006.
- XILINX Inc. System Generator for DSP version 8.1. ed., San Jose California: XILINX Inc., 2005.
- XILINX Inc. DSP: Designing for Optimal Results ed. 1.0: XILINX Inc., Mar. 2005b.
- Xiong, Z., Linsheng, Dongfeng, Z. e Zengshou, D., “A New Adaptive Step-Size Blind Equalization Algorithm Based on Autocorrelation Error Signal”, *IEEE Proceedings of the 7th International Conference on Signal Processing, ICSP’04*, vol 2, pp. 1719-1722, agosto-setembro 2004.

- 
- Zhidkov, S.V., “Impulsive noise suppression in OFDM-based communication systems”, *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 944 – 948, novembro 2003.
  - Zipf, J.G.F., Tobias, O.J., e Seara, R., “Estudo Comparativo e Classificação de Algoritmos LMS de Passo Variável”. *XXV Simpósio Brasileiro de Telecomunicações (SBrT 2007)*, pp. 1-6, setembro 2007.

## Nota biográfica

**F**rancisco J. A. de Aquino nasceu em Fortaleza, Ceará, em 1970. Concluiu o ensino técnico em Telecomunicações no CEFET-Ce (antiga Escola Técnica Federal do Ceará), em 1988. Obteve a graduação em Engenharia Elétrica, com ênfase em eletrônica e informática, em 1992. Entre 1992 e 1994, trabalhou como engenheiro eletrônico na empresa Fyber Equipamentos Eletrônicos LTDA. Desde 1994 é professor no CEFET-Ce, atuando nos níveis médio e superior. Em dezembro de 1998 obteve o título de Mestre em Eng. Elétrica pela UFSC, na área de concentração de instrumentação eletrônica e processamento de sinais no laboratório LINSE. Também foi professor na UNIFOR (Universidade de Fortaleza) nos cursos de Eng. Eletrônica e Eng. de Telecomunicações no período de 2000 a 2003. De março de 2004 a setembro de 2008 esteve envolvido em atividades acadêmicas como estudante de doutorado na UFSC, junto ao laboratório GpqCom. Seus principais interesses de pesquisa incluem processamento digital de sinais, sistemas de comunicação sem fio e implementação em FPGA de algoritmos adaptativos.