

IDENTIFICAÇÃO DE CANAIS MÓVEIS UTILIZANDO ALGORITMOS
SM-AFFINE PROJECTION

José Ednelson Wesen Moreira

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

Aprovada por:

Prof. Paulo Sérgio Ramirez Diniz, Ph.D.

Prof. Sérgio Lima Netto, Ph.D.

Prof. José Antônio Apolinário Júnior, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

JULHO DE 2005

WESEN, JOSÉ EDNELSON

Identificação de Canais Móveis
Utilizando Algoritmos SM-Affine
Projection [Rio de Janeiro] 2005

VIII, 82 pp 29,7 cm (COPPE/UFRJ,
M.Sc., Engenharia Elétrica, 2005)

Dissertação - Universidade Federal do
Rio de Janeiro, COPPE

1.Filtragem Adaptativa 2.Identificação
de Sistemas 3.Algoritmos SM-Affine
Projection 4.Canal Variante no Tempo

I.COPPE/UFRJ II.Título (série)

Agradecimentos

- A Deus, por tudo;
- Aos professores Paulo Sérgio Ramirez Diniz e Stefan Werner pela orientação, confiança e incentivo;
- Aos meus pais Joaquim Moreira de Almeida e Maria Wesen de Almeida e meus irmãos por estarem sempre ao meu lado em todos os momentos;
- À minha esposa Naath dos Santos Sevalho pelo companheirismo e compreensão;
- Aos meus filhos Fabinho, Bruno e Bianca pelo estímulo de permanecer vivo;
- Aos professores da COPPE que ministraram as disciplinas deste curso;
- Aos colegas de curso pela amizade e incentivo;
- À Amazônia Celular, especialmente ao Mário Jorge Barros Gadelha e equipe pela amizade, incentivo e compreensão ;
- À SUFRAMA, órgão financiador deste curso;
- À UFAM, em especial à professora Marly Guimarães Fernandes Costa, que tornou possível a realização deste curso;
- À UFRJ por ter possibilitado o convênio com a UFAM.

Muito obrigado a todos. Espero poder retribuir a altura o que fizeram por mim.

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

IDENTIFICAÇÃO DE CANAIS MÓVEIS UTILIZANDO ALGORITMOS
SM-AFFINE PROJECTION

José Ednelson Wesen Moreira

Julho/2005

Orientador: Paulo Sérgio Ramirez Diniz

Programa: Engenharia Elétrica

A filtragem adaptativa tem-se constituído numa ferramenta extremamente valiosa para o processamento digital de sinal, sendo empregada nos mais diversos campos da Engenharia e de outras ciências.

Neste trabalho, a técnica da filtragem adaptativa é vista atuando em um importante ambiente da comunicação digital: identificação do canal de sistema móvel celular.

Neste crescente segmento, é fundamental dispor de algoritmos que atendam a exigentes critérios de desempenho, sendo um dos mais críticos a complexidade computacional.

Nesta tese, um novo algoritmo baseado no *Set-Membership Affine Projection* apresenta-se como uma promissora opção para a modelagem do canal de rádio móvel.

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

IDENTIFICATION OF WIRELESS COMMUNICATION CHANNEL USING
SM-AFFINE PROJECTION ALGORITHM

José Ednelson Wesen Moreira

July/2005

Advisor: Paulo Sérgio Ramirez Diniz

Department: Electrical Engineering

Adaptive filters have become a valuable digital signal processing tool which find application in a wide range of fields of engineering and many others.

This work addresses the application of adaptives filters in an important environment of digital communications, namely: mobile wireless channel identification.

In this application, it is crucial to use algorithms meeting several difficult performance requirements, including reduced computational complexity.

This work proposes a variation of the set-membership affine projection algorithm suitable for identification of time-varying wireless channels.

Sumário

1	A Filtragem Adaptativa	1
1.1	Introdução	1
1.1.1	Estrutura básica dos filtros digitais não-recursivos	2
1.1.2	Estrutura básica dos filtros digitais recursivos	3
1.1.3	Filtros convencionais x filtros adaptativos	4
1.2	Fundamentos da Filtragem Adaptativa	5
1.3	Classificação das Aplicações da Filtragem Adaptativa	11
1.3.1	Predição	11
1.3.2	Cancelador de Interferência	12
1.3.3	Identificação (Modelagem de Sistemas)	13
1.3.4	Equalização (Modelagem Inversa)	14
1.4	Organização da Tese	17
1.5	Resumo	18
2	Descrição do Padrão e do Setup Experimental	19
2.1	Introdução	19
2.2	O Canal de Rádio Móvel	19
2.3	Modelagem do Canal de Rádio Móvel	20
2.4	O Padrão 3GPP Release 4 para Modelo de Canal Móvel - COST259 .	20
2.5	Determinação de um Canal de Rádio Móvel de Referência pelo Padrão COST259	22
2.6	Setup Experimental	29
2.6.1	Parâmetros para a Determinação do Modelo de Canal Expe- rimental	29

2.6.2	Análise das Propriedades Estatísticas do Canal Experimental .	30
2.6.3	Valores de Ajuste para Emprego nas Simulações	33
2.7	Resumo	34
3	Avaliação do Desempenho dos Algoritmos Clássicos	35
3.1	Introdução	35
3.2	Algoritmo LMS	35
3.2.1	Propriedades do Algoritmo LMS	38
3.2.2	Identificação de Sistema	40
3.3	Algoritmo LMS Normalizado	42
3.3.1	Identificação de Sistema	43
3.4	Algoritmo RLS Convencional	45
3.4.1	Propriedades do Algoritmo RLS Convencional	48
3.4.2	Identificação de Sistema	49
3.5	Resumo	51
4	Avaliação e Otimização do Algoritmo SM-Affine Projection	52
4.1	Introdução	52
4.2	Filtragem do tipo Set-Membership	52
4.3	O Algoritmo Set-Membership Affine Projection	53
4.4	Identificação de Sistema com o SM-AP	56
4.5	Otimização do Set-Membership Affine Projection	59
4.6	Identificação de Sistema com o Algoritmo SM-Affine Projection com Fator de Reuso Variável	63
4.7	Resumo	65
5	Comparações	66
5.1	Introdução	66
5.2	SM-APVDR <i>versus</i> SM-AP	66
5.3	SM-APVDR <i>versus</i> Algoritmos Clássicos	72
5.4	Resumo	74
6	Conclusões e Trabalhos Futuros	75
6.1	Conclusões	75

6.2	Trabalhos Futuros	76
	Referências Bibliográficas	77
A	Código Matlab para o algoritmo SM-APVDR	80

Capítulo 1

A Filtragem Adaptativa

1.1 Introdução

Os filtros são dispositivos destinados a impedir a propagação ou eliminar informações indesejáveis, sejam impurezas, raios ultra-violetas ou ruídos sonoros[1]. Tradicionalmente, os filtros eletrônicos analógicos impedem que uma determinada faixa de frequência de um sinal aplicado a sua entrada possa atingir a sua saída.

Os filtros digitais são os equivalentes do analógicos no âmbito do processamento digital de sinais. Neste sentido, podemos nos referir aos filtros digitais como sendo qualquer dispositivo que provoca alteração de alguma característica do sinal de entrada, proporcionando um sinal de saída mais próxima do que é desejável, cuja implementação utiliza a tecnologia digital.

A utilização do filtro digital é necessária todas as vezes em que se requer a realização, entre outras, das seguintes tarefas:

- extração da informação desejada no instante k utilizando dados medidos até este instante (Filtragem);
- recuperação do conteúdo espectral de uma informação que foi filtrada por um canal (Equalização);
- previsão das amostras futuras de uma seqüência, através do emprego de informação passada (Predição).

Os filtro digitais lineares podem apresentar-se em uma de duas formas: a não-recursiva ou a recursiva.

1.1.1 Estrutura básica dos filtros digitais não-recursivos

A saída de um filtro não-recursivo, no instante k , está relacionada com a entrada pela equação de diferença:

$$y(k) = \sum_{i=0}^{N-1} w_i x(k-i) \quad (1.1)$$

em que:

w_i são os N coeficientes do filtro

$y(k)$ é a saída no instante k

$x(k)$ é a entrada no instante k

Como a resposta ao impulso dos filtros não-recursivos possui comprimento finito, estes filtros são também referenciados como FIR (*Finite-duration Impulse-Response*) [2].

Os coeficientes w_i correspondem à resposta ao impulso do filtro, pois a função de transferência $H(z)$ é dada por:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{i=0}^{N-1} h_i z^{-i} \quad \text{ou} \quad H(z) = Z\{h(k)\} = \sum_{i=0}^{N-1} w_i z^{-i} \quad (1.2)$$

donde se conclui que $h_i = w_i, i = 0, 1, \dots, N-1$.

A realização da forma direta canônica é obtida a partir da equação (1.2) fornecendo a Figura 1.1.

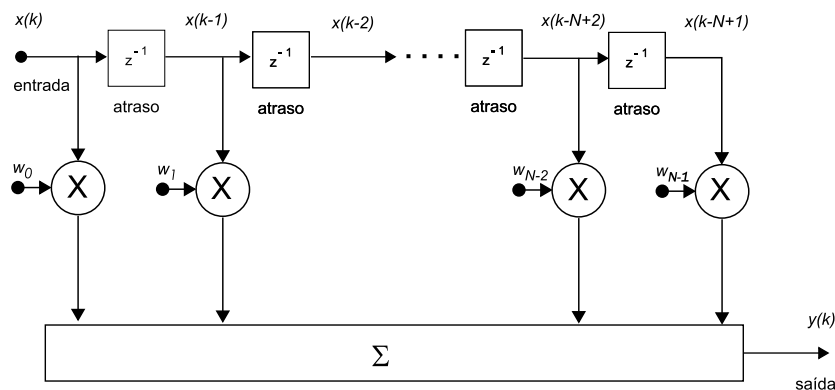


Figura 1.1: Realização direta canônica do tipo1 de um filtro FIR.

1.1.2 Estrutura básica dos filtros digitais recursivos

Os sinais de entrada e saída, no instante k , relacionam-se através da equação de diferença:

$$y(k) = \sum_{i=0}^M b_i x(k-i) - \sum_{i=1}^N a_i y(k-i) \quad (1.3)$$

em que:

a_i e b_i são os coeficientes do filtro

$y(k)$ é a saída no instante k

$x(k)$ é a entrada no instante k

Uma vez que na maioria dos casos a função de transferência dá origem a filtros com resposta ao impulso possuindo duração infinita, os filtros recursivos são também referenciados como IIR (Infinite-duration Impulse-Response).

A Figura 1.2 apresenta um filtro recursivo em uma forma canônica do tipo 1[2].

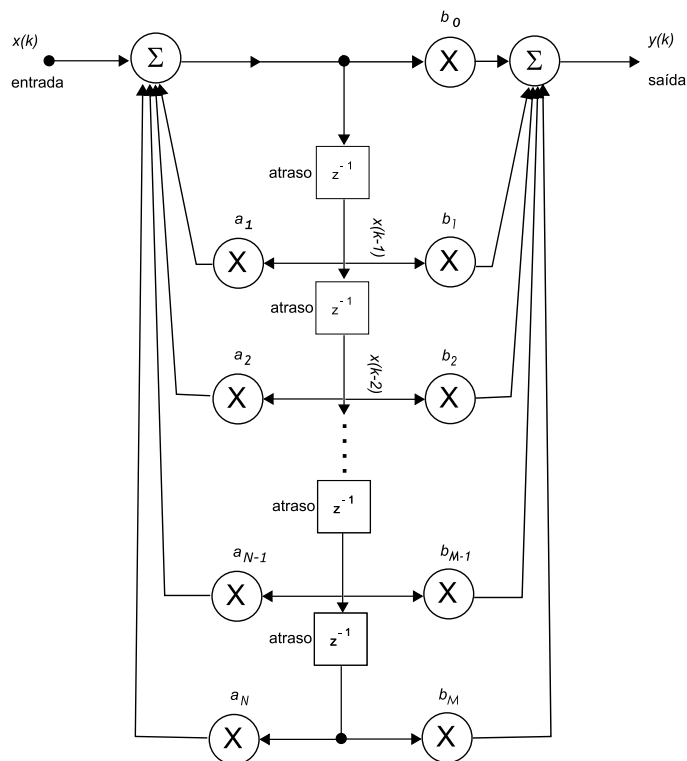


Figura 1.2: Realização direta canônica do tipo 1 de um filtro IIR.

1.1.3 Filtros convencionais x filtros adaptativos

Os filtros digitais convencionais apresentam as seguintes características mais comuns:

- são lineares e invariantes no tempo;
- o processo de projeto utiliza informação sobre a banda de frequência em que o filtro selecionará;
- os coeficientes do filtro são escolhidos durante a fase de projeto e permanecem constantes durante toda a operação normal do filtro.

Há, entretanto, determinadas aplicações que não podem ser solucionadas adequadamente com o simples emprego de filtros com coeficientes fixos. Quer seja porque não se dispõe de informação suficiente para efetuar o projeto do filtro fixo, quer seja porque os critérios do projeto sofrem alterações durante a operação normal da aplicação. Uma grande parcela destas aplicações pode ser resolvida através da utilização de um tipo especial de filtro: os filtros adaptativos. A principal característica deste tipo de filtro é o fato de possuir a habilidade de modificar sua resposta automaticamente objetivando melhorar seu próprio desempenho durante a operação. Assim, um filtro adaptativo é um filtro capaz de realizar a auto-aprendizagem, isto é, à medida que o tempo decorre, o filtro vai ajustando suas características automaticamente de modo a atingir um desempenho requerido. Podemos relacionar como as principais características dos filtros adaptativos os fatos de que:

- são filtros com coeficientes que variam no tempo;
- adaptam-se automaticamente às alterações nos sinais de entrada;
- baseiam-se em algoritmo de minimização do erro.

Os filtros adaptativos, a exemplo dos filtros convencionais, podem ser classificados em lineares e não-lineares. A rigor um filtro adaptativo não poderia ser considerado linear, uma vez que seus parâmetros dependem dos dados de entrada e variam de uma iteração para outra, violando o princípio da superposição. Entretanto, consideramos que o filtro adaptativo é linear quando este utiliza uma estrutura de filtro digital linear (se fixarmos seus coeficientes).

1.2 Fundamentos da Filtragem Adaptativa

Devido à habilidade que os filtros adaptativos possuem de atualizar seus parâmetros usando somente a informação disponível no ambiente, eles são de grande utilidade nas situações em que não estão disponíveis as informações (estatísticas) sobre o ambiente da aplicação ou as condições são variantes no tempo. São também úteis nos casos em que os requisitos para o projeto de filtros com coeficientes fixos não podem ser facilmente especificados [3].

A Figura 1.3 mostra o esquema geral de um filtro adaptativo em que k é o número da iteração, $x(k)$ é o sinal de entrada, $y(k)$ é a saída do filtro adaptativo que na realidade é uma estimativa da resposta desejada $d(k)$ e $e(k)$ é o sinal de erro definido como a diferença entre a resposta desejada e a saída do filtro $e(k) = d(k) - y(k)$.

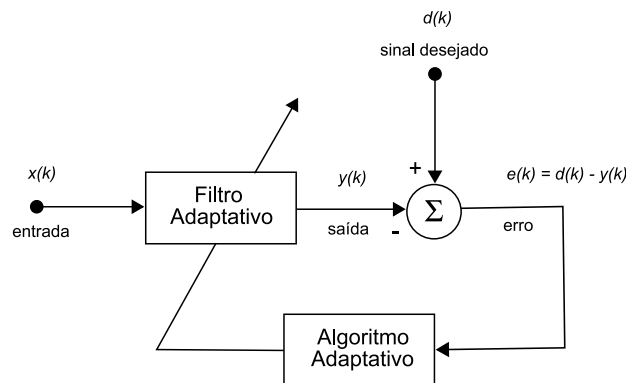


Figura 1.3: Diagrama geral de um filtro adaptativo.

Assim é o funcionamento de um filtro adaptativo: a saída do filtro $y(k)$ é comparada com o sinal desejado $d(k)$. Através da observação do erro de estimação $e(k) = d(k) - y(k)$, um algoritmo de adaptação ajusta os coeficientes do filtro buscando minimizar a função objetivo.

Normalmente o algoritmo de adaptação é implementado através de um dos dois métodos clássicos de minimização: método do gradiente e método dos mínimos quadrados.

No método do gradiente a atualização dos parâmetros de filtro é feita caminhando na direção oposta à indicada pelo vetor gradiente da função objetivo. O

tamanho do passo é determinado por um fator que pode ser constante ou variável ao longo do processo de minimização da função objetivo. A escolha deste fator é um compromisso entre a velocidade de convergência, estabilidade e erro residual do algoritmo. O papel de determinar uma escolha apropriada para este fator é, no entanto, desempenhada por simulações em computador na maioria dos casos [4].

Portanto, a equação de atualização dos coeficientes que resume o algoritmo do gradiente possui a forma:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{grad}[F(e(k))]$$

onde:

$\mathbf{w}(k)$: vetor dos coeficientes do filtro no instante k

μ : passo de adaptação

$F(e(k))$: função objetivo que se deseja minimizar

No método dos mínimos quadrados o objetivo é determinar os coeficientes do filtro que minimizam a soma dos quadrados do erro de estimativa até o presente instante. A janela desta soma pode ser retangular — quando todos os quadrados dos erros têm o mesmo peso — ou pode ser uma janela exponencial decrescente na qual os quadrados dos erros mais recentes têm relevância maior em relação aos anteriores. Neste caso, emprega-se um fator que controlará o tamanho da janela, o chamado fator de esquecimento. Assim, este método soluciona o problema:

$$\min \sum_{i=0}^k \lambda^{k-i} [d(i) - y(i)]^2$$

O escalar λ é o fator de esquecimento, pois determina a memória do algoritmo. Ele deve ser escolhido no intervalo $0 \ll \lambda < 1$. Se $\lambda = 1$ teremos uma janela retangular, pois todos os erros terão a mesma importância. A janela decrescente é convenientemente recomendada para o caso de canais de comunicação ou em aplicações em que os sinais são estatisticamente variantes no tempo.

A solução do problema de otimização apresentado requer o cálculo da inversa da matriz de correlação determinística do sinal de entrada, o que oferece complexidade de $O[N^3]$. Assim, recorre-se normalmente à obtenção desta de forma recursiva.

Os métodos dos mínimos quadrados apresentam rápida convergência e excelente *performance* quando operam em ambientes variantes no tempo; no entanto, possuem uma grande complexidade computacional e alguns problemas de estabilidade. Já os métodos do gradiente são pouco complexos e extremamente robustos; porém, a convergência é mais lenta o que também compromete o funcionamento quando o sinal de entrada não é estacionário ou fortemente correlacionado.

A função objetivo ou função de custo é na realidade o critério de qualidade para avaliar o comportamento do filtro. No processo de adaptação, os parâmetros $\mathbf{w}(k)$ do filtro são ajustados para produzirem uma saída $y(k)$ o mais próximo possível do sinal desejado $d(k)$; esta diferença, o erro de estimação $e(k)$, deve estar relacionado com a função objetivo. Assim sendo, ao minimizar a função objetivo a filtragem torna-se mais eficaz.

A definição desta função objetivo necessita atender a dois critérios básicos[4]:

- Não-negatividade: $F(e(k)) \geq 0$;
- Optimalidade: $F(0) = 0$.

A complexidade computacional do algoritmo de adaptação está intimamente relacionada com a definição da função objetivo. Dentre as funções objetivos mais comumente utilizadas nos algoritmos adaptativos, temos:

- Erro Quadrático Médio (MSE - *Mean-Square Error*): $F(e(k)) = E[|e(k)|^2]$;
- Mínimos Quadrados (LS - *Least Squares*): $F(e(k)) = \frac{1}{k+1} \sum_{i=0}^k |e(k-1)|^2$;
- Mínimos Quadrados Ponderados (WLS - *Weighted Least Squares*):

$$F(e(k)) = \sum_{i=0}^k \lambda^i |e(k-1)|^2, \quad 0 \ll \lambda < 1;$$
- Erro Quadrático Instantâneo (ISV - *Instantaneous Square Value*):

$$F(e(k)) = |e(k)|^2.$$

À medida que o desempenho do filtro melhora através da redução da função objetivo, a auto-aprendizagem está ocorrendo e sua evolução pode ser avaliada por um gráfico que apresenta o comportamento da função objetivo ao longo do processo

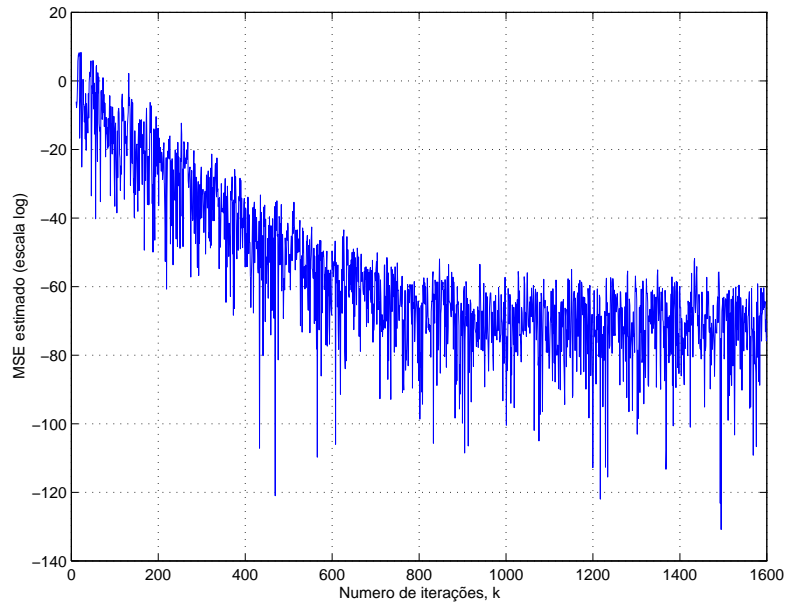


Figura 1.4: Curva de aprendizagem do erro quadrático instantâneo.

adaptativo. Este gráfico recebe o nome de curva de aprendizagem da função de custo. A Figura 1.4 apresenta um exemplo típico desta curva.

De um modo geral, a minimização do erro quadrático médio (MSE) é a figura de mérito mais amplamente utilizada, uma vez que fornece algoritmos adaptativos de fácil implementação.

O erro quadrático médio é uma função de 2^a ordem dos coeficientes do filtro e pode ser visto como uma superfície hiperparabolóide de $N + 1$ dimensões, onde N é a ordem do filtro. No caso particular em que o filtro possui somente dois coeficientes, esta superfície é um parabolóide como ilustrado na Figura 1.5.

O hiperparabolóide possui um único mínimo global e nenhum mínimo local. As coordenadas correspondentes ao valor mínimo da superfície fornecem os coeficientes de um filtro conhecido pelo nome de filtro de Wiener. Deste modo, o filtro de Wiener é ótimo no sentido quadrático médio.

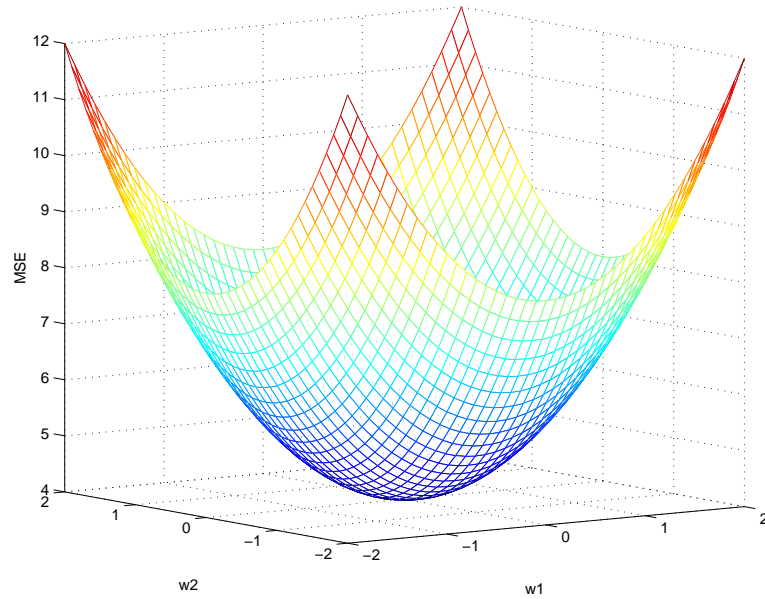


Figura 1.5: Superfície do erro quadrático médio

O filtro de Wiener pode ser obtido através da equação de Wiener-Hopf

$$\mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p}$$

onde:

\mathbf{w}_o : vetor dos coeficientes do filtro de Wiener

$\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$: matriz de autocorrelação do sinal de entrada

$\mathbf{p} = E[d(k)\mathbf{x}(k)]$: vetor de correlação cruzada entre o sinal desejado e a sequência de entrada

O filtro de Wiener nem sempre pode ser obtido na prática, pois requer a matriz de autocorrelação \mathbf{R} e o vetor de correlação cruzada \mathbf{p} que não são conhecidos *a priori*. Além disso, essa solução requer uma inversão matricial que é computacionalmente dispendioso e, se a entrada não for estacionária, \mathbf{R} e \mathbf{p} variam no tempo; conseqüentemente, \mathbf{w}_o precisa ser repetidamente calculado.

De fato, a importância da solução de Wiener reside no fato de servir como parâmetro de referência para avaliação dos diversos algoritmos aplicados à filtragem adaptativa.

O sinal de saída de um filtro adaptativo é produzido a partir de medições do sinal de entrada e de uma estrutura fixa com parâmetros ajustados pelo algoritmo adaptativo. Esta estrutura pode ser implementada basicamente por duas classes de realizações distintas pela forma da resposta ao impulso, são elas a FIR (*Finite-duration Impulse Response*) e IIR (*Infinte-duration Impulse Response*), como visto na Seção 1.1.

Tendo em vista que os filtros não-recursivos são mais amplamente utilizados na filtragem adaptativa devido à sua simplicidade relativa e garantia de estabilidade, daqui em diante será utilizado exclusivamente este tipo de filtro.

Deste modo, os sinais de saída e de erro dos filtros que consideraremos obedecem às seguintes relações:

$$y(k) = \sum_{i=0}^{N-1} w_i x(k-i) = \mathbf{w}^T \mathbf{x}(k) \quad e$$

$$e(k) = d(k) - y(k) = d(k) - \mathbf{w}^T \mathbf{x}(k)$$

As medidas mais importantes do desempenho dos filtros adaptativos são:

- Taxa de convergência: velocidade com que o algoritmo atinge a vizinhança da solução ótima. É normalmente expressa em número de iterações;
- Desajuste: exprime a diferença relativa entre o erro final cometido pelo algoritmo e o erro mínimo proporcionado pela solução ótima;
- Habilidade de *Tracking*: capacidade de acompanhar as variações estatísticas quando opera em um ambiente não-estacionário;
- Complexidade computacional: número de operações necessárias para cada iteração;
- Características numéricas: sensibilidade a erros de quantização e à precisão finita nos cálculos.

1.3 Classificação das Aplicações da Filtragem Adaptativa

Atualmente a filtragem adaptativa é aplicada com sucesso nos mais diversos campos do conhecimento humano como telecomunicações, geofísica, sismologia, engenharia biomédica, no controle de máquinas industriais e no processamento de sinais de áudio, vídeo, radar e sonar. Embora estas várias aplicações sejam de natureza bastante distinta, uma característica comum pode ser observada: um vetor de entrada e uma resposta desejada são usados para calcular o erro de estimação que controla os valores de um conjunto de coeficientes ajustáveis do filtro. As várias aplicações da filtragem adaptativa diferem no modo como este vetor de entrada e a resposta desejada são escolhidos. Neste contexto, é possível distinguir quatro classes básicas de aplicações: predição adaptativa, cancelamento de interferência, identificação de sistema e equalização.

Dependendo da aplicação, a introdução de atrasos pode ser necessária para fazer o sistema funcionar. A razão é simples: o filtro adaptativo é sempre causal, devido ao tempo de propagação através deste. Se o sinal filtrado estiver muito atrasado no ponto de comparação, o sistema possuirá desempenho ruim.

1.3.1 Predição

Nesta aplicação, o filtro adaptativo fornece uma predição do valor presente do sinal de entrada, que poderá ser usado como um modelo deste sinal. O valor presente do sinal é utilizado como a resposta desejada, ao passo que os valores passados do sinal constituem a entrada aplicada ao filtro, conforme a Figura 1.6.

De acordo com a aplicação, os valores de interesse podem ser o sinal de saída do filtro ou o erro de estimação. No primeiro caso, o sistema opera com a função de preditor do sinal de entrada, enquanto que no segundo, este opera como um filtro de erro de predição.

O preditor adaptativo possui um vasto campo de aplicação. Como exemplos temos a codificação linear preditiva, PCM diferencial adaptativo (ADPCM), análise espectral e detecção de sinais.

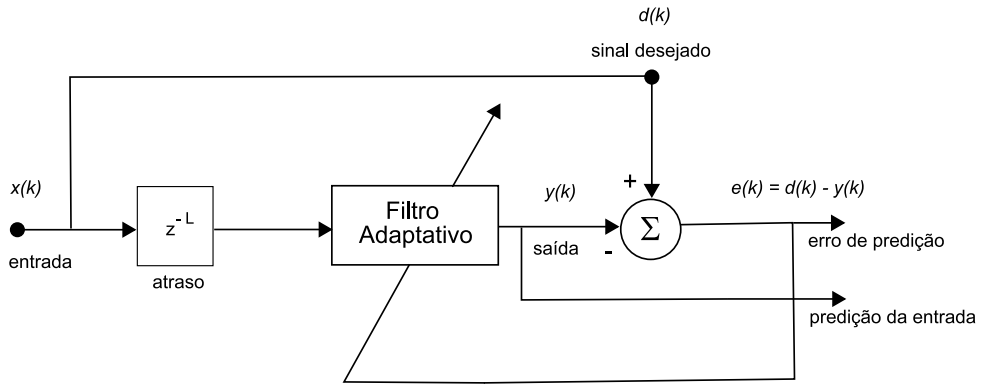


Figura 1.6: Predição de sinal.

1.3.2 Cancelador de Interferência

Nesta classe de aplicação, o filtro adaptativo tem a função de cancelar uma interferência desconhecida presente no sinal de entrada principal. A Figura 1.7 ilustra uma configuração empregada para este propósito.

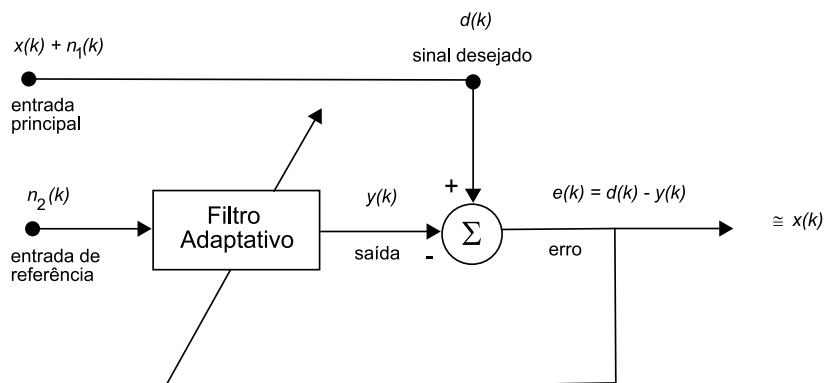


Figura 1.7: Cancelamento de interferências.

O sinal na entrada de referência contém a informação $x(k)$ corrompida pelo ruído $n_1(k)$, este sinal serve como a resposta desejada do filtro. O sinal na entrada de referência contém $n_2(k)$, uma versão distorcida, porém correlacionada com $n_1(k)$ que é obtida a partir de um sensor ou conjunto de sensores localizados, em relação ao sensor que fornece a entrada principal, de tal maneira que a informação $x(k)$ presente, seja essencialmente não-detectável. O filtro aproximará $y(k)$ de $n_1(k)$ de modo que ao final da convergência o erro de estimação será aproximadamente igual a:

$$e(k) = x(k) + n_1(k) - y(k) \approx x(k) + n_1(k) - n_1(k) = x(k)$$

Idealmente, o filtro remove o ruído e fornece o sinal de interesse $x(k)$ intacto; entretanto, na prática, os ruídos do sistema não são completamente removidos mas seu nível é reduzido consideravelmente.

O cancelador de interferências é aplicado no cancelamento de eco em auditórios, cancelamento de interferências em eletromedicina, supressão de ruído em cabines de avião, dentre outras aplicações.

1.3.3 Identificação (Modelagem de Sistemas)

A identificação de sistemas, também conhecida como modelagem de sistema, refere-se à habilidade de um sistema adaptativo encontrar um modelo linear que melhor reproduza a resposta de uma planta desconhecida. Este modelo matemático do sistema pode ser fundamental para algumas ciências como as sociais, econômicas e biológicas, além da engenharia.

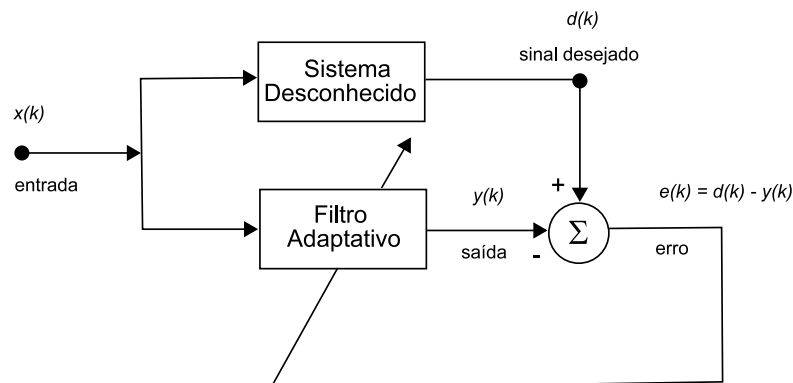


Figura 1.8: Identificação de sistema.

A Figura 1.8 apresenta a configuração clássica para esta aplicação. A planta e o filtro adaptativo são submetidos simultaneamente ao mesmo sinal de entrada que normalmente é um ruído branco a fim de cobrir todas as frequências. A saída da planta fornece a resposta desejada para o filtro. Neste caso, o filtro tentará reproduzir o comportamento do sistema desconhecido, isto é, à medida que o algoritmo converge o filtro emula a planta.

É conveniente mencionar que se a planta for de natureza dinâmica, o modelo será variante no tempo.

Sendo o filtro adaptativo do tipo FIR e o sistema a modelar for do tipo *all-*

pole não será possível reduzir a zero o erro de estimação (não estamos considerando aqui o inevitável ruído de medição), mas certamente após a convergência o filtro fornecerá a melhor aproximação que um FIR pode proporcionar. No mundo real, os sistemas desconhecidos geralmente apresentam pólos e zeros na função de transferência. Novamente, o filtro adaptativo fará o melhor possível para emular este sistema, sendo que seu sucesso dependerá da real resposta em frequência da planta e do número de coeficientes do filtro empregado. Quanto maior for o número de coeficientes melhor será a aproximação.

É a técnica da identificação de sistemas que está por trás de uma função atualmente bastante comum nos sistemas de som de alta-fidelidade: a função que nos permite escolher um ambiente imerso no qual desejamos ouvir a música. Assim, é possível ouvi-la como se esta estivesse tocando em um *show* ao vivo, em uma igreja, em um teatro ou em um estádio. A música passa por um filtro digital que emula a resposta em frequência do cenário em questão.

Existem diversas outras aplicações desta técnica. Para citarmos algumas: modelagem de canais de comunicação com multipercurso, cancelamento de eco causado pelas híbridas em telecomunicações, sistemas de controle e na geofísica.

1.3.4 Equalização (Modelagem Inversa)

A técnica da equalização adaptativa fornece um modelo inverso que melhor se ajusta a um sistema desconhecido. O filtro adaptativo é cascateado com a planta objetivando compensar a distorção linear introduzida pelo canal – distorção de amplitude e de fase.

A Figura 1.9 mostra o esquema usual desta aplicação. Como pode ser visto, uma versão atrasada do sinal entrante na planta desconhecida constitui a resposta desejada do filtro, enquanto o sinal proveniente da planta adicionado ao ruído é a entrada do filtro adaptativo.

No caso de filtros em cascata como este, o conjunto sistema desconhecido-filtro adaptativo provoca um atraso no sinal que chega ao somador da comparação com a resposta desejada. A fim de evitar que o filtro tente se adaptar a um sinal que ainda não foi visto (equivalente a ter que também desempenhar a função de predição), deve-se provocar um atraso no sinal da resposta desejada. É comum não

se saber a ordem do sistema que se está equalizando. Neste caso, é recomendado atrasar a resposta desejada de pelos menos da metade da ordem do filtro adaptativo.

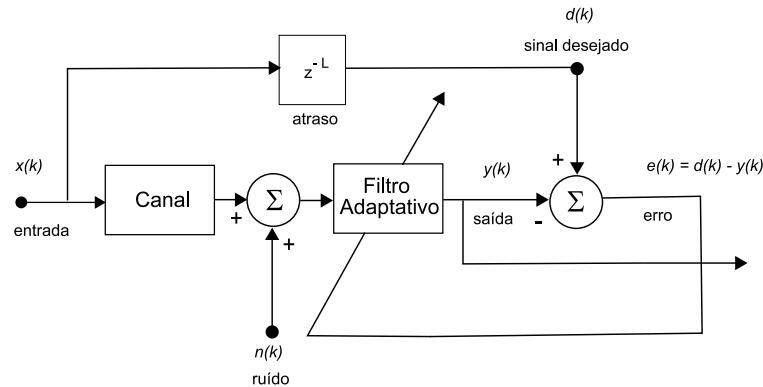


Figura 1.9: Equalização de canal.

Idealmente, o modelo inverso tem uma função de transferência igual ao recíproco da função de transferência da planta. Isto é fundamental para os sistemas de comunicação, pois teoricamente o conjunto canal-equalizador deve apresentar uma função de transferência total de módulo constante a fim de evitar a interferência intersimbólica (*ISI-Intersymbol Interference*) que consiste na influência que os símbolos vizinhos têm uns nos outros e que pode provocar uma decisão incorreta pelo receptor.

A equalização adaptativa normalmente apresenta duas fases: a fase de treinamento e a fase adaptativa com decisão direta. Nos casos em que é viável a execução da fase de treinamento, uma seqüência pseudo-aleatória de comprimento suficiente a permitir que o equalizador compense as distorções do canal, é enviada e comparada com valores previamente armazenados no receptor e devidamente sincronizados com o emissor. Isto permite a atualização dos coeficientes do filtro de forma a minimizar o erro de estimação e conseqüentemente reduzir significativamente a ISI. Após esta fase é que ocorre efetivamente a transmissão da informação e a atualização dos parâmetros do filtro é feita com base na seqüência de saída do próprio decisor presente no receptor.

É necessário mencionar que há casos que impedem a realização da fase de treinamento. Nestas situações, esta fase é substituída por uma fase de equalização às cegas (*blind equalization*) em que são exploradas características conhecidas dos sinais transmitidos.

Existe uma estrutura mais sofisticada, também utilizada para a equalização em sistemas de comunicação digital: o Equalizador de Decisão Realimentada (*DFE-Decision Feedback Equalizer*) em que além do filtro linear adaptativo, já mencionado, é introduzido um outro para realimentar uma combinação linear das decisões já tomadas, auxiliando no cancelamento da ISI (Figura 1.10).

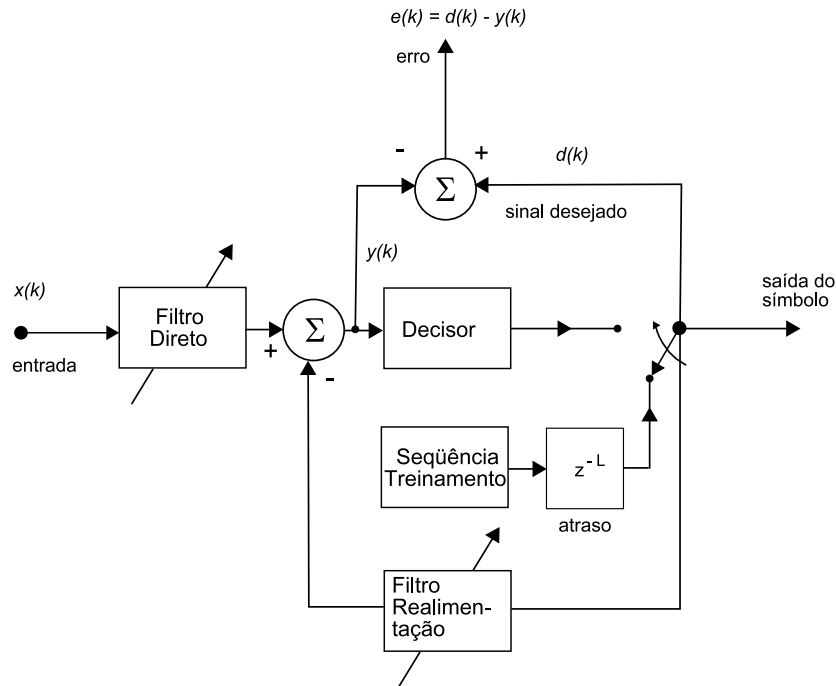


Figura 1.10: Equalizador com realimentação da decisão DFE.

O filtro linear direto responsabiliza-se por eliminar a resposta pré-cursora do canal enquanto o filtro de realimentação encarrega-se da resposta pós-cursora, gerando uma cópia da ISI resultante desta resposta para subtraí-la da saída do filtro direto. Este tipo de sistema apresenta um bom desempenho na presença de ISI severa.

O filtro DFE combate mais eficazmente o ruído aditivo, porém propaga erros de decisão, ao passo que o filtro linear é sensível ao ruído, mas devido à ausência de uma malha de realimentação não propaga decisões incorretas. O equalizador DFE é mais sensível à fase do amostrador que o equalizador síncrono, problema que pode ser contornado empregando-se a técnica da sobreamostragem do sinal para o filtro direto.

1.4 Organização da Tese

O objetivo deste trabalho é o de avaliar e otimizar os parâmetros do algoritmo proposto em [3], o algoritmo *SM-Affine Projection*, para a aplicação de identificação de um sistema constituído por um canal do tipo SISO com multipercurso e variante no tempo no qual serão transmitidos pulsos G-MSK empregados no padrão de comunicação móvel GSM-*Global System for Mobile Communications*.

Para isso, no Capítulo 1, foram abordados os conceitos mais relevantes envolvendo a filtragem adaptativa dando uma especial atenção à aplicação na qual aprofundaremos nosso estudo.

No Capítulo 2 será apresentada uma modelagem para o canal de nosso interesse, levando em consideração o que é recomendado pelo padrão 3GPP *release 4* para modelo de canal móvel - COST259. Nesta altura, optaremos por um *cluster* específico associado a uma velocidade e uma série de parâmetros constituindo, assim, um cenário que procure contemplar as necessidades mais frequentes em uma realização prática.

Serão também definidos os aspectos dos filtros adaptativos a serem utilizados na avaliação da aplicação de interesse.

No Capítulo 3, o canal que previamente foi obtido através do método de Monte Carlo será utilizado para a avaliação do desempenho de diversos algoritmos clássicos fornecendo a criação de uma ampla base a ser empregada para fins de comparação nos capítulos seguintes.

No Capítulo 4, o algoritmo *SM-Affine Projection* entra em ação e será submetido às mesmas condições dos algoritmos clássicos. Será apresentada uma análise dos resultados obtidos e alterações de parâmetros do algoritmo de modo a alcançar sua operação otimizada.

O Capítulo 5 apresenta uma comparação entre as principais medidas de desempenho dos algoritmos através de tabelas e de gráficos convenientes.

O Capítulo 6 exhibe as conclusões decorrentes das análises e sugestões para novos trabalhos.

1.5 Resumo

Este capítulo iniciou apresentando as considerações gerais sobre os filtros digitais convencionais e adaptativos, explorando os aspectos da estrutura básica e suas principais características. Em seguida foi feita uma ampla abordagem sobre as bases fundamentais da filtragem adaptativa, expondo os elementos que constituem esta técnica e os requisitos necessários para empregá-la adequadamente. As principais classes de aplicação foram analisadas e diversos exemplos nos quais a filtragem adaptativa tem obtido sucesso foram mencionados. O capítulo é finalizado pela apresentação da maneira como acontecerá o desenvolvimento deste trabalho.

Capítulo 2

Descrição do Padrão e do Setup Experimental

2.1 Introdução

Este capítulo fornecerá um canal multipercurso lentamente variante no tempo, representado discretamente em forma de um filtro FIR com coeficientes variáveis. Sua obtenção ocorrerá através do padrão 3GPP release 4 para modelo de canal móvel – COST259, parametrizado por fatores que forneçam uma situação o mais próximo possível daquela encontrada na realidade. Este canal, assim determinado, constituirá o cenário a ser empregado ao longo deste trabalho.

Por fim, a aplicação de interesse – Identificação de um canal móvel – terá seus aspectos definidos.

2.2 O Canal de Rádio Móvel

Nas comunicações móveis, as ondas eletromagnéticas enviadas geralmente não chegam diretamente à antena receptora devido aos obstáculos na linha de visada. As ondas recebidas são, na realidade, a superposição de ondas que chegam de todas as direções devido à reflexão, difração e dispersão provocadas por edificações, árvores e outros obstáculos. Este efeito é conhecido por propagação em multipercurso. Por conta do multipercurso, o sinal recebido consiste de um somatório de infinitas réplicas do sinal transmitido que sofreram atenuação, atraso e deslocamento em fase,

umas influenciando as outras. Esta superposição pode ser construtiva ou destrutiva, dependendo da fase de cada onda parcial [5].

Além da propagação em multipercurso, ocorre também o efeito Doppler que tem uma influência negativa nas características do canal de rádio móvel. Devido ao movimento relativo da estação móvel, o efeito Doppler causa o deslocamento da frequência de cada onda parcial provocando uma dispersão na frequência. No domínio do tempo, o efeito Doppler implica que a resposta ao impulso do canal torna-se variante no tempo.

Pode-se demonstrar que o canal de rádio móvel obedece ao princípio da superposição [6, 5] e, portanto, constitui-se num sistema linear. Desta forma, o canal móvel pertence à classe dos sistemas lineares variantes no tempo.

2.3 Modelagem do Canal de Rádio Móvel

Muitos modelos de canais discretos são baseados em uma coleta empírica de dados obtidos a partir de uma ampla gama de ambientes [7, 8, 9]. Tendo em vista a imensa quantidade de modelos, quer matemáticos quer empíricos, propostos recentemente, os projetistas dos sistemas de comunicação móvel deparam-se com uma tremenda dificuldade para escolher um conjunto representativo que modele uma realização prática do mundo real e proporcione as condições para a operação satisfatória do sistema projetado. Felizmente, entidades de padronização internacionais fornecem especificações para um conjunto de canais representativos destinados à análise e simulação do desempenho de diferentes tipos de sistemas de comunicação [10].

A próxima seção apresentará o modelo apresentado pelo comitê COST (Cooperação Européia no Campo da Investigação Científica e Tecnológica).

2.4 O Padrão 3GPP Release 4 para Modelo de Canal Móvel - COST259

Constitui-se num pré-requisito fundamental para o desenvolvimento, simulações e testes de sistemas móveis, a existência de modelos adequados de canal. O

modelo COST207 desempenhou um papel fundamental para o desenvolvimento e avaliação do sistema GSM [11]. O enorme sucesso alcançado por este modelo não é devido apenas a este fato mas também pela sua simplicidade [12]. Entretanto, a crescente demanda de capacidade que conduzirá à implementação de sistemas de terceira e posteriormente de quarta geração requer modelos que incluam informações que possibilitem o emprego de antenas adaptativas e que contemple o uso de micro ¹ e picocélulas ². Uma vez que o modelo COST207 não vislumbrava estas situações, foi necessário desenvolver um modelo que atendesse a estes requisitos. Assim, o COST259 pode ser encarado como uma generalização do bem-sucedido modelo COST207, em três aspectos:

- informações a cerca da direção de chegada do sinal na estação rádio-base e na estação móvel são consideradas;
- não somente macrocélulas, mas também micro e picocélulas são tratadas;
- mudanças em larga escala (*large scale*) são modeladas.

Um outro aspecto a considerar é que o COST207 foi projetado para sistemas não-direcionais e com largura de canal da ordem de kHz, como foi comum nos anos 90s. O padrão COST259 é definido para portadoras de até 20 GHz [13].

A filosofia básica de modelagem seguida pelo COST259 é a abordagem estatística para descrição do canal, com a vantagem de que poucos parâmetros são suficientes para descrever completamente o canal. Por outro lado, foi estabelecida uma classificação para a situação do ambiente. Um conjunto de ambientes de rádio (macro, micro e picocélulas) foi definido, cada qual consistindo de vários cenários de propagação.

Adicionalmente aos cenários para macrocélulas, que são versões generalizadas dos tipos de canais do COST207 (TU/BU/RA/HT), como vistos a seguir, novos cenários são definidos para micro e picocélulas. O modelo COST259 especifica quatro ambientes de rádio para macrocélulas e quatro para microcélulas [14]. Os

¹célula de diâmetro inferior a 700 metros e que não é setorizada. Normalmente esta célula guarda uma certa distância de outras microcélulas.

²microcélula de diâmetro, normalmente, menor do que 200 metros e que é usada, principalmente, em interiores de prédios.

nomes destes ambientes começam com a letra G (para Generalizado) seguido da abreviação do ambiente considerado. Para as macrocélulas, estes ambientes são os mesmos do COST207[15]: TU (*typical urban*), BU (*bad urban*), RA (*rural area*) e HT (*hilly terrain*). Para as microcélulas, SN (*street non-line-of-sight*), SL (*street line-of-sight*), SX (*street crossing*) e OP (*open place*).

Cada cenário é caracterizado por um conjunto de parâmetros (estocásticos) que se distinguem em externo, global e local. Os parâmetros externos são aqueles que permanecem inalterados para as simulações, como a banda de frequência ou a altura média das estações rádio-bases. Os parâmetros locais determinam o espectro de potência do atraso angular dentro um intervalo de algumas dezenas de comprimentos de onda, como o fator de distribuição de Rice, número de *clusters* que espalham os sinais, sobre cerca de dez comprimentos de onda. Os parâmetros globais determinam as funções de densidade de probabilidade dos parâmetros locais.

2.5 Determinação de um Canal de Rádio Móvel de Referência pelo Padrão COST259

Assume-se que o canal de rádio móvel é variante no tempo e apresenta o efeito da propagação em multipercurso, ver Seção 2.2, e que o desempenho de sistemas de rádio-comunicação digital é fortemente afetado pela propagação em multipercurso na forma de dispersão, reflexão e refração. Com isso, o padrão COST259 absorveu do COST207 a idéia de que o canal estocástico pode ser modelado pela classe de processo não-degenerativo mais simples que exhibe dispersão descorrelacionada no tempo de atraso e nos deslocamentos de Doppler conhecido como WSSUS (Wide-Sense Stationary Uncorrelated Scattering). Assim, assume-se para um canal uma superposição linear de ecos descorrelacionados e que seja Estacionário no Sentido Amplo (WSS). A suposição de estacionaridade no sentido amplo (WSS) é vista como válida, quando somente uma área local de recepção é tomada em consideração. A suposição do espalhamento descorrelacionado (US) pode ser entendida como uma conseqüência efetiva da alta frequência da portadora. Dois percursos que chegam à antena receptora com um atraso relativo de poucos centímetros causarão uma completa incerteza nas relações de fase mútua; assim, eles podem ser tratados como sinais

descorrelacionados. O canal assim definido pode ser totalmente determinado pela função de espalhamento (*scattering function*) bidimensional em termos do atraso de eco τ devido ao multipercurso e a frequência Doppler f_D devido ao movimento relativo do veículo.

Seja $f(\tau, t)$ a resposta ao impulso de um canal linear e lentamente variante no tempo, a qual é definida como a resposta no instante t a um impulso aplicado ao canal no instante $t - \tau$. A autocorrelação de $f(\tau, t)$ é dada por [16]

$$E[f(\tau, t)f^*(\tau', t')] = E[\delta(\tau - \tau')f(\tau, t)f^*(\tau', t + \Delta t)] = Q(\tau, \Delta t)\delta(\tau - \tau') \quad (2.1)$$

A transformada de Fourier de $Q(\tau, \Delta t)$ com respeito a Δt fornece a função de espalhamento $P_S(\tau, f_D)$, ou seja:

$$P_S(\tau, f_D) = \mathcal{F}\{Q(\tau, \Delta t)\} \quad (2.2)$$

Devido à equação (2.1), a função de espalhamento representa uma completa descrição do canal

$$P_S(\tau, f_D) = \int_{-\infty}^{+\infty} Q(\tau, \Delta t)e^{-j2\pi f_D \Delta t} d\Delta t \quad (2.3)$$

A equação (2.3) indica a dispersão de energia do canal na direção do tempo (*delay spread*) e da frequência (*Doppler spread*), e pode ser interpretada diretamente como uma função densidade de probabilidade bidimensional [17]. Esta equação pode ser separada em dois fatores, assumindo-se que o efeito Doppler e os atrasos de percursos são processos independentes.

O espectro de potência Doppler do canal ($S_f(f_D)$ *Doppler power spectrum*) e o espectro de potência dos atrasos ($S_\tau(\tau)$ *delay power spectrum*) são obtidos pela integração da função de espalhamento com respeito a f_D e τ respectivamente:

$$S_f(f_D) = \int_0^{+\infty} P_S(\tau, f_D) d\tau \quad (2.4)$$

$$S_\tau(\tau) = \int_{-\infty}^{+\infty} P_S(\tau, f_D) df_D \quad (2.5)$$

Como visto na seção anterior, o padrão COST259 define alguns perfis para canais típicos. O espectro de potência do atraso deve ser modelado com um decaimento exponencial. A função densidade de probabilidade apropriada é do tipo:

$$p_{\tau}(\tau) = \begin{cases} \frac{\beta}{\alpha} e^{-\beta\tau} & : 0 \leq \tau \leq \tau_m \\ 0 & : \text{outros casos} \end{cases}$$

em que:

β : constante de decaimento

τ_m : tempo máximo de atraso

α : fator de ajuste dado por $1 - e^{-\beta\tau_m}$

Pode haver casos em que mais de um *cluster* deve ser considerado. O canal GBU (Generalized Bad Urban) por exemplo, contém tipicamente dois *clusters* para o eco como mostra a Figura 2.1.

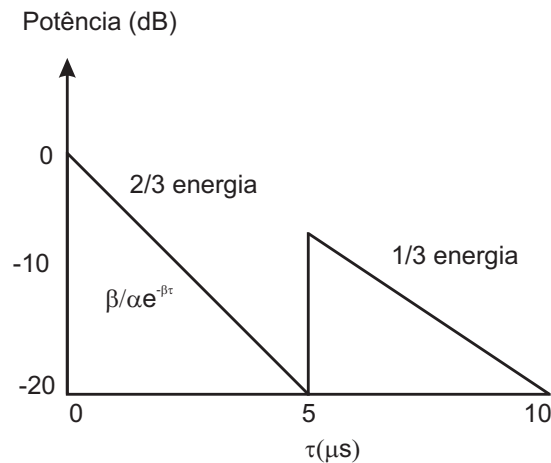


Figura 2.1: Perfil da potência do atraso (Delay power) de um canal típico GBU

O espalhamento da energia no domínio da frequência é causada pelo movimento do automóvel, ou para ser mais preciso, pelo efeito Doppler. A frequência Doppler máxima é determinada pela relação da velocidade do veículo e da frequência da portadora

$$f_{Dmax} = v \frac{f_o}{c_o}$$

em que:

f_{Dmax} : frequência Doppler máxima
 v : velocidade de deslocamento do móvel
 f_o : frequência da portadora
 c_o : velocidade da luz no vácuo

Para um sistema GSM que usa frequência de portadora na faixa de 900 MHz e para uma velocidade de cerca de 60 km/h resulta em uma frequência Doppler máxima $f_{Dmax}=50$ Hz.

Os quatro tipos de densidade espectral de potência Doppler especificados pelo COST259 são apresentados graficamente na Figura 2.2, cujas formulações encontram-se na Tabela 2.1, em que:

$$G(A, f_1, f_2) = Ae^{-\frac{(f-f_1)^2}{2f_2^2}}$$

$$A_1 = \frac{50}{\sqrt{2\pi}3f_{Dmax}}$$

$$A_2 = \frac{10^{1.5}}{\sqrt{2\pi}(\sqrt{10+0.15})f_{Dmax}}$$

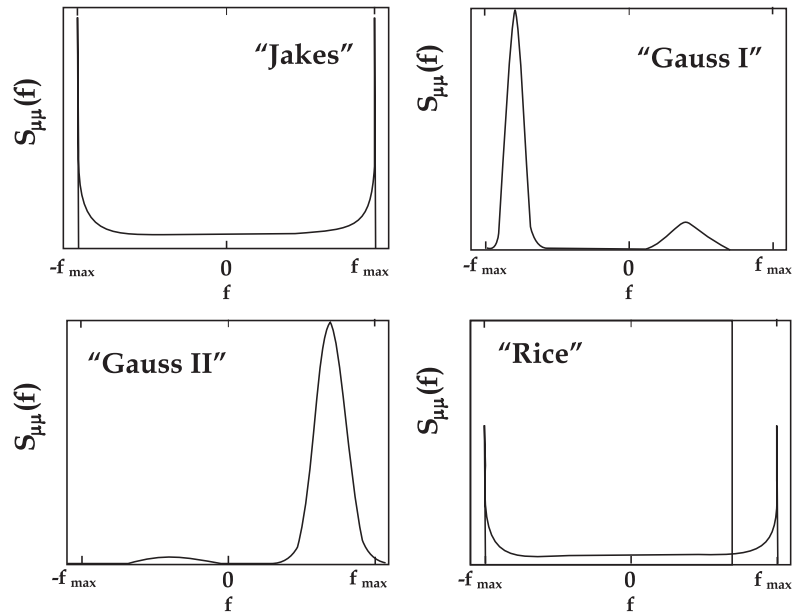


Figura 2.2: Densidades espectrais de potência Doppler $S_{\mu\mu}(f)$ para os modelos de canal de acordo com o COST259.

A densidade espectral de potência de Jakes ou o espectro clássico ocorre apenas nos casos de propagação com atraso muito pequenos ($\tau' \leq 0.5\mu s$). Somente

Tabela 2.1: Especificações das densidades espectrais de potência Doppler $S_{\mu\mu}(f)$.

Tipo	Densidade Espectral de Potência Doppler $S_{\mu\mu}(f)$	Atraso de Propagação τ'	Espalhamento Doppler $B_{\mu\mu}$
Jakes	$\frac{1}{\pi f_{Dmax}} \frac{1}{\sqrt{1 - (\frac{f_D}{f_{Dmax}})^2}}$	$0 \leq \tau' \leq 0.5\mu s$	$\frac{f_{Dmax}}{\sqrt{2}}$
Gauss I	$G(A_1, -0.8f_{Dmax}, 0.05f_{Dmax})$ $+G(\frac{A_1}{10}, 0.4f_{Dmax}, 0.1f_{Dmax})$	$0.5\mu \leq \tau' \leq 2\mu s$	$0.45f_{Dmax}$
Gauss II	$G(A_2, -0.7f_{Dmax}, 0.1f_{Dmax})$ $+G(\frac{A_2}{10^{1.5}}, -0.4f_{Dmax}, 0.15f_{Dmax})$	$\tau' \geq 2\mu s$	$0.25f_{Dmax}$
Rice	$\frac{0.41^2}{\pi f_{Dmax}} \frac{1}{\sqrt{1 - (\frac{f_D}{f_{Dmax}})^2}}$ $+0.91^2\delta(f - 0.7f_{Dmax})$	$\tau' = 0\mu s$	$0.39f_{Dmax}$

neste caso a suposição de que as amplitudes dos componentes são homogêneas e os ângulos de chegada são uniformemente distribuídos de 0 a 2π são justificadas (linhas tracejadas na Figura 2.3). Para os componentes espalhados com atrasos de propagação médio e longo τ' , entretanto, assume-se que as correspondentes frequências Doppler são normalmente distribuídas, resultando uma densidade espectral de potência Doppler com a forma gaussiana (Figura 2.2 “Gauss I” e “Gauss II”).

A Figura 2.4 ilustra uma função de espalhamento típica para o canal GBU modelado pelo COST259.

Um método amplamente empregado para a obtenção dos parâmetros do modelo é o método de Monte Carlo (MMC). Este método foi proposto por Schulze [18] para a modelagem estocástica e para simulação digital de canais de rádio móvel.

Hoeher [17] discutiu uma aproximação pelo método de Monte Carlo para um canal WSSUS gerando a resposta instantânea ao impulso do canal como

$$f(\tau, t) = \lim_{N \rightarrow \infty} \frac{1}{\sqrt{N}} \sum_{n=1}^N e^{j(\theta_n + 2\pi f_{Dn}t)} \delta(\tau - \tau_n) \quad (2.6)$$

com $N = 50$ a 500 na prática. N é o número de caminhos de ecos que compõem o canal e o fator $\frac{1}{\sqrt{N}}$ garante um valor médio unitário para a energia do canal.

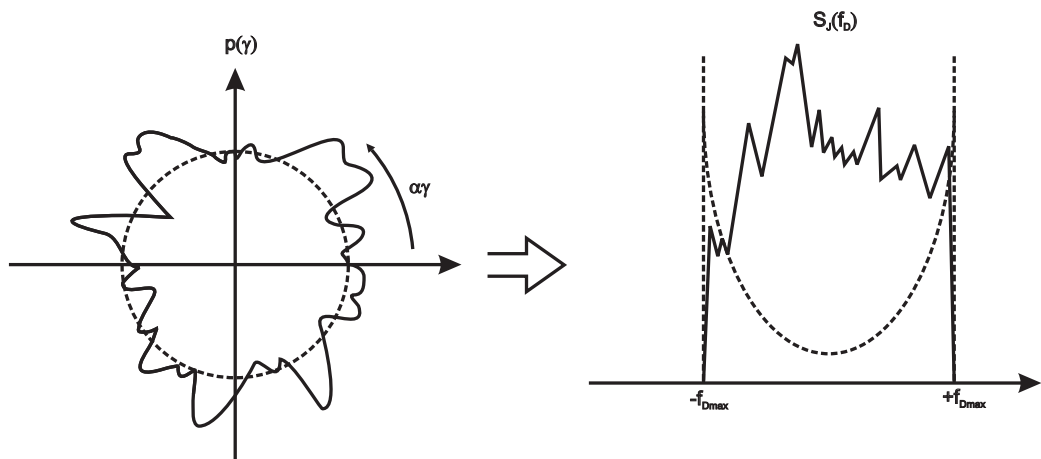


Figura 2.3: Distribuição angular da potência Doppler e densidade espectral.

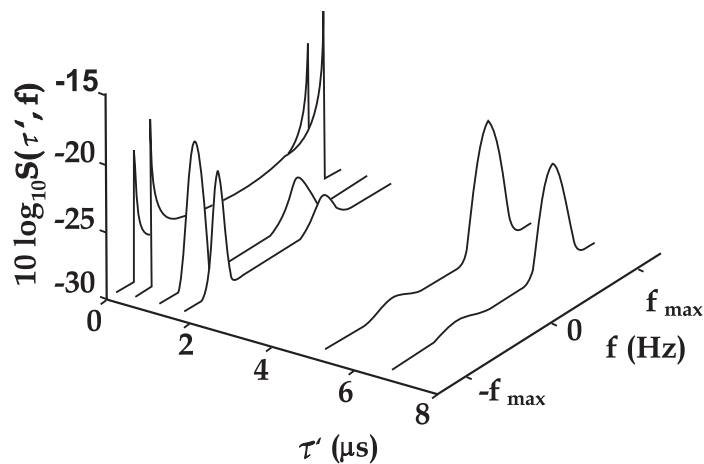


Figura 2.4: Função de espalhamento típica de um canal multipercuro GBU.

O ângulo de fase do canal θ_n é gerado conforme uma distribuição uniforme de 0 a 2π .

A resposta instantânea ao impulso do canal $f(\tau, t)$ é um processo aleatório gaussiano e as amplitudes apresentam, portanto, a distribuição de Rayleigh.

A Figura 2.5 mostra uma aplicação típica do emulador de canal em um sistema de transmissão digital.

É interessante analisar o caso de um filtro com coeficientes fixos na recepção. A razão é que receptores de baixo-custo geralmente não fazem uso de filtro adaptativo na recepção e sim de um filtro casado com o filtro do transmissor. Neste caso, o filtro de recepção da Figura 2.5 deve ser substituído por um filtro passa-baixa ideal. Esta situação pode ser aplicada aos resultados a serem obtidos em seguida

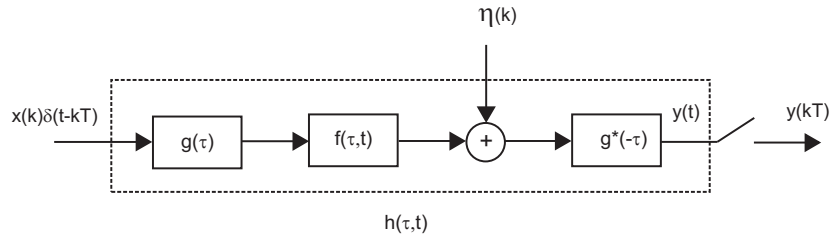


Figura 2.5: Modelo de canal multipercursos com desvanciamento.

com poucas alterações.

Para um filtro com coeficientes fixos na recepção $g^*(-\tau)$, a resposta global ao impulso da composição formada pelo filtro de transmissão $g(\tau)$, canal $f(\tau, t)$ e o filtro de recepção $g^*(-\tau)$ pode ser escrita como

$$h(\tau, t) = f(\tau, t) * g(\tau) * g(-\tau)$$

O filtro resultante da convolução do filtro da transmissão com o da recepção será denotado como $g_{total}(\tau)$.

$$h(\tau, t) = f(\tau, t) * g_{total}(\tau)$$

E a resposta global ao impulso passa a ser da forma

$$h(\tau, t) = \lim_{N \rightarrow \infty} \frac{1}{\sqrt{N}} \sum_{n=1}^N e^{j(\theta_n + 2\pi f_{Dn}t)} g_{total}(\tau - \tau_n) \quad (2.7)$$

Considerando a taxa de amostragem de símbolo $t = kT$, obtém-se um modelo para o canal discreto com coeficientes variantes no tempo conforme é ilustrado na Figura 2.6.

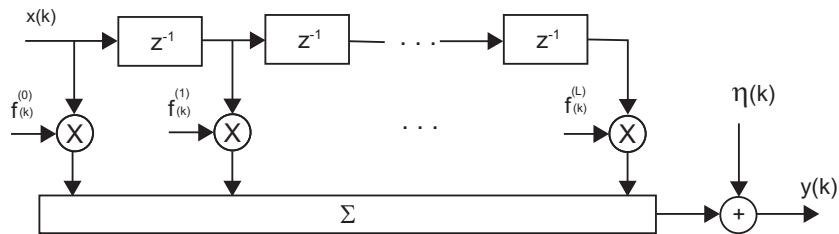


Figura 2.6: Modelo equivalente ao canal discreto no tempo que representa o modelo global da transmissão.

As amostras de saída do canal $y(k)$ são dadas por:

$$\begin{aligned}
Y(kT) = y(k) &= \sum_j x(j)h(kT - jT, kT) + \eta(k) \\
&= \sum_{l=L_-}^{l=L_+} x(k-l)h(lT, kT) + \eta(k) \\
&= \sum_{l=L_-}^{L_+} x(k-l)f^{(l)}(k) + \eta(k)
\end{aligned} \tag{2.8}$$

com

$$f^{(l)}(k) = h(lT, kT) = \lim_{N \rightarrow \infty} \frac{1}{\sqrt{N}} \sum_{n=1}^N e^{j(\theta_n + 2\pi f_{Dn}kT)} g_{total}(lT - \tau_n)$$

onde L_- e L_+ são inteiros positivos de tal modo que $h(\tau, t)$ é zero fora do intervalo $(L_-)T \leq t \leq (L_+)T$. Se $\eta(t)$ for o ruído branco e se $g_{total}(t)$ satisfizer o critério de Nyquist para eliminar a interferência intersimbólica, então os $\eta(k)$ são amostras discretas do ruído branco.

De um modo geral os coeficientes $f_k^{(l)}$ são correlacionados, logo o modelo discreto no tempo não satisfaz a condição de espalhamento não-correlacionado (US) mesmo para ecos descorrelacionados. A razão é que o processo aleatório de tempo contínuo τ é representado por um processo discreto no tempo com espaçamento T .

Os coeficientes $f_k^{(l)}$ são complexos e constituem um processo gaussiano com média nula, isto é, as amplitudes obedecem à distribuição de Rayleigh e as fases são uniformemente distribuídas.

2.6 Setup Experimental

Para os objetivos deste trabalho, um canal de rádio móvel experimental será obtido empregando o método descrito na seção anterior. Este canal servirá como base para todas as simulações a serem efetuadas daqui em diante.

2.6.1 Parâmetros para a Determinação do Modelo de Canal Experimental

Um sistema móvel celular real é constituído, de fato, pela soma dos cenários apresentados pelo modelo COST259 atuando simultaneamente e, caracteriza-se pela operação predominante em regiões urbanizadas. Tendo em vista que o presente

trabalho objetiva otimizar o algoritmo *Set-Membership Affine Projection* para a modelagem do canal de rádio móvel, os parâmetros a serem empregados, para a obtenção deste, poderão gerar a situação mais comumente encontrada na prática, sem perda de generalização. Desta forma, os parâmetros a serem empregados na obtenção experimental do canal de referência, serão:

- Tipo de célula: Macro célula
- Tipo de área: GBU (Generalized Bad Urban) - Área densamente urbana
- Velocidade de deslocamento da estação móvel: 60 km/h
- Frequência da portadora: 900 MHz (GSM)
- Formato do pulso: Gaussiano (GMSK)
- Quantidade de amostras: 2000

2.6.2 Análise das Propriedades Estatísticas do Canal Experimental

As Figuras 2.7 a 2.10 apresentam os resultados obtidos para o canal de referência. Pode-se comprovar, através de uma análise comparativa entre o que foi obtido e aquilo que o modelo COST259 estabelece para o cenário adotado, que estes resultados enquadram-se perfeitamente no que seria o esperado, indicando a validade das considerações feitas anteriormente e creditando a sua utilização para o fim a que se destina neste trabalho.

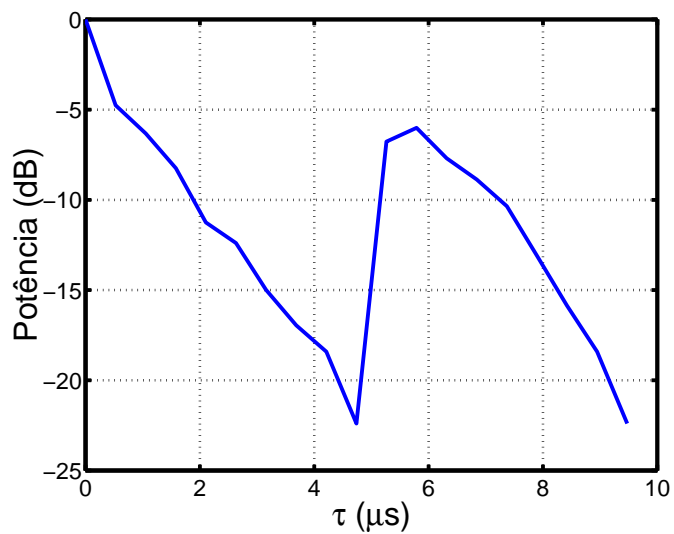


Figura 2.7: Função de densidade de potência dos atrasos na propagação para o canal experimental.

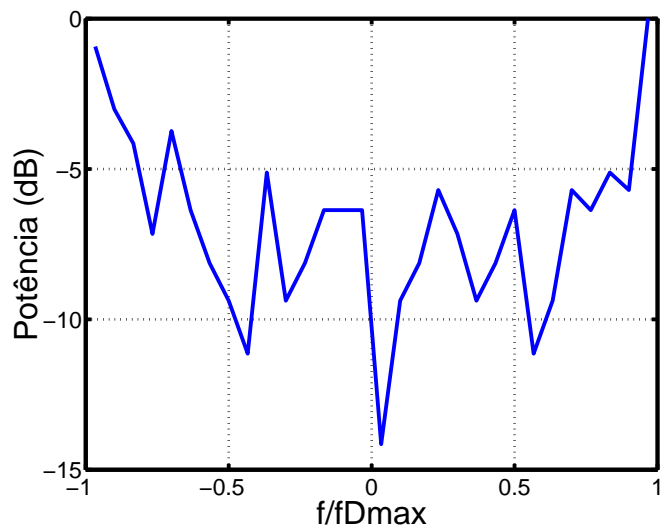


Figura 2.8: Função de densidade de potência Doppler para o canal experimental com $\tau' = 0$.

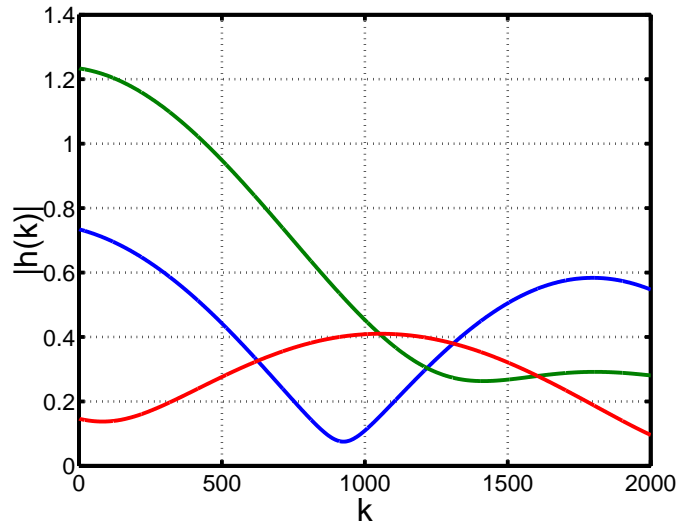


Figura 2.9: Evolução no tempo dos coeficientes 1, 5 e 10 do canal experimental.

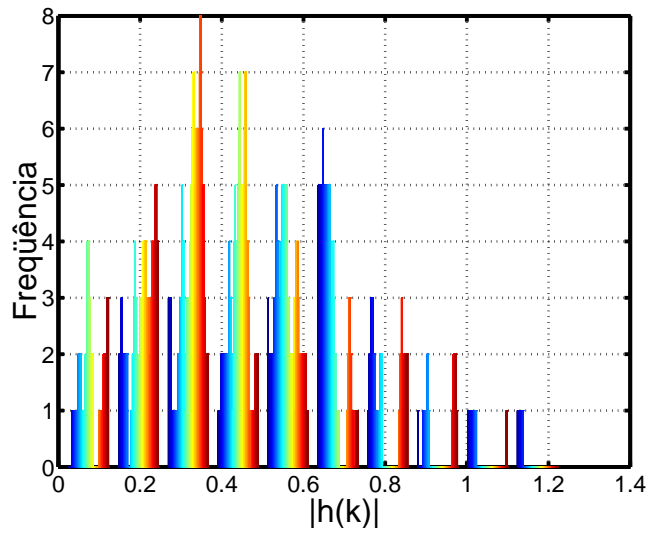


Figura 2.10: Histograma para as amplitudes do coeficiente 1 do canal experimental.

2.6.3 Valores de Ajuste para Emprego nas Simulações

Uma vez obtido o canal de referência, faz-se necessário neste instante estabelecer as características comuns dos sinais e de alguns parâmetros a serem utilizados nas avaliações dos algoritmos adaptativos analisados nos próximos capítulos. Estes ajustes serão efetuados para o tipo de aplicação de interesse: Identificação de Sistema. Para a aplicação, o número de simulações será igual a 500. Portanto, os resultados obtidos refletirão a média destes experimentos.

2.6.3.1 Referências para a Identificação de Sistemas

O diagrama da Figura 2.11 apresenta o modo como será realizada esta aplicação, onde os sinais em questão são modelados da seguinte forma:

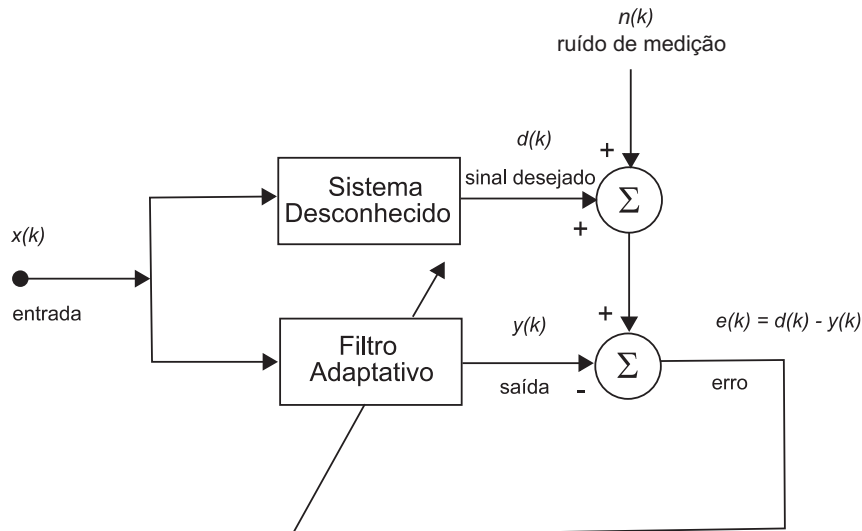


Figura 2.11: Identificação de sistema com o canal experimental.

- $x(k)$: sinal de entrada que consistirá num ruído colorido gerado através da filtragem de um ruído gaussiano pelo IIR da quarta ordem $y(k) = 0.6617y(k - 1) + 0.3402y(k - 2) + 0.5235y(k - 3) - 0.8703y(k - 4)$.
- $n(k)$: ruído de medição, que é inevitável. É do tipo AWGN (Aditive Wideband Gaussian Noise) com média nula e potência ajustada para proporcionar uma relação sinal-ruído (SNR - *Signal-to-Noise Ratio*) $SNR = 30$ dB, 40 dB e 50 dB.

O objetivo de utilizar um ruído colorido como sinal de entrada é o de obter uma matriz de autocorrelação de entrada com um número de condicionamento diferente da unidade.

2.7 Resumo

O capítulo iniciou com uma breve descrição do canal de rádio móvel apresentando suas características principais, tais como os aspectos relacionados ao atraso de propagação do multipercurso e o efeito Doppler provocado pelo movimento da estação móvel em relação à estação rádio-base. Em seguida foi abordada a modelagem deste canal elaborado pelo grupo COST259 e estabelecendo as formulações matemáticas para a obtenção deste. O método de Monte Carlo foi o instrumento utilizado para a obtenção de um canal experimental que visa servir de referência para todas as simulações a serem efetuadas neste trabalho.

Na seqüência foi realizada uma breve análise das principais propriedades estatísticas deste canal e, por fim, foram estabelecidos valores dos parâmetros que servirão de padrão para as avaliações dos algoritmos adaptativos a serem testados.

Capítulo 3

Avaliação do Desempenho dos Algoritmos Clássicos

3.1 Introdução

Neste capítulo será realizada uma série de simulações utilizando os algoritmos clássicos na aplicação de interesse empregando o canal de referência disponível e os valores de ajustes padronizados no Capítulo 2.

Estas simulações fornecerão uma sólida base que será utilizada para efeito de comparação na seqüência do trabalho.

Os algoritmos da filtragem adaptativa que serão objeto deste capítulo são: LMS, LMS normalizado e o RLS convencional.

3.2 Algoritmo LMS

O filtro de Wiener é um método para a obtenção dos parâmetros ótimos para um filtro linear discreto para ambientes estacionários e quando se dispõe das estatísticas dos sinais envolvidos. Este filtro faz uso do critério da minimização do erro quadrático médio, cujo valor é obtido a partir da diferença do sinal desejado e da saída do filtro. No entanto, a solução de Wiener, além de exigir o prévio conhecimento das estatísticas dos sinais envolvidos, requer uma inversão matricial que pode ser muito dispendiosa computacionalmente, especialmente quando o filtro apresenta uma grande quantidade de coeficientes. Assim, surgem métodos iterativos como o

da descida mais íngreme (*steepest-descent algorithm*) que utilizam o vetor gradiente do critério para obter os coeficientes até convergir para a solução de Wiener. O algoritmo adaptativo conhecido por LMS (Least Mean Square) parte do mesmo princípio utilizando uma aproximação estocástica do verdadeiro vetor gradiente. Desta forma, o algoritmo LMS torna-se excepcionalmente simples, do ponto de vista computacional, destacando-se também por sua robustez e comportamento estável quando implementado usando aritmética de precisão finita e sendo normalmente utilizado como referência de desempenho, função que desempenhará neste trabalho.

O algoritmo LMS foi proposto em 1959 por Widrow e Hoff e consiste simplesmente em substituir os valores médios dos sinais envolvidos pelos seus valores instantâneos, uma vez que não é possível calcular valores médios com base em um único instante k sem conhecer a estatística de conjunto.

As principais características do algoritmo LMS são:

- Não exige o conhecimento prévio das características do sinal de entrada;
- Utiliza estimativas instantâneas das estatísticas do sinal de entrada;
- As estimativas dos coeficientes do filtro vão sendo melhoradas à medida que este aprende as características de entrada;
- Os coeficientes nunca atingem o valor ótimo (solução de Wiener) mas oscilam em torno deste.

A idéia por trás do algoritmo de descida mais íngreme é a de que o gradiente da função de custo é um vetor que aponta para a direção de crescimento desta função. Como o objetivo do algoritmo é minimizá-la, deve-se caminhar na direção oposta; logo, a equação genérica que define a atualização dos coeficientes é a seguinte

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \mathbf{grad}[F(e(k))] \quad (3.1)$$

A constante μ é uma constante de proporcionalidade positiva, chamada passo de adaptação ou fator de convergência, de que depende o desempenho do algoritmo.

A função de custo que o algoritmo da descida mais íngreme deve minimizar é o erro quadrático médio (MSE) definida como:

$$F[e(k)] = \xi(k) = E[e^2(k)] = E[d^2(k) - 2d(k)y(k) + y^2(k)]$$

No caso do filtro FIR com coeficientes constantes, a função objetivo pode ser escrita como

$$\begin{aligned}\xi(k) &= E[d^2(k) - 2d(k)\mathbf{w}^T \mathbf{x}(k) + \mathbf{w}^T \mathbf{x}(k)\mathbf{x}(k)^T \mathbf{w}] \\ &= E[d^2(k)] - 2\mathbf{w}^T E[d(k)\mathbf{x}(k)] + \mathbf{w}^T E[\mathbf{x}(k)\mathbf{x}(k)^T] \mathbf{w} \\ &= E[d^2(k)] - 2\mathbf{w}^T \mathbf{p} + \mathbf{w}^T \mathbf{R} \mathbf{w}\end{aligned}$$

onde:

$\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$: matriz de autocorrelação do sinal de entrada

$\mathbf{p} = E[d(k)\mathbf{x}(k)]$: vetor de correlação cruzada entre o sinal desejado e a seqüência de entrada

O gradiente da função de custo com relação aos coeficientes do filtro é dado por

$$\mathbf{grad}[F(e(k))] = \mathbf{grad}[\xi(k)] = \frac{\partial \xi(k)}{\partial \mathbf{w}} = -2\mathbf{p} + 2\mathbf{R}\mathbf{w}$$

Existe também uma outra maneira de expressar o gradiente. Sendo $e(k) = d(k) - y(k) = d(k) - \mathbf{x}(k)^T \mathbf{w}$ e sendo possível trocar os operadores E e ∂ por serem lineares, vem:

$$\mathbf{grad}[\xi(k)] = \frac{\partial E[e^2(k)]}{\partial \mathbf{w}} = E \left[\frac{\partial e^2(k)}{\partial \mathbf{w}} \right] = E \left[2e(k) \frac{\partial e(k)}{\partial \mathbf{w}} \right] = -2E[e(k)\mathbf{x}(k)]$$

Tendo em vista que a simplificação introduzida pelo algoritmo LMS é a substituição dos valores médios pelos instantâneos, pode-se concluir que a estimativa do gradiente $\hat{\mathbf{g}}_{\mathbf{w}}$ será da forma

$$\hat{\mathbf{g}}_{\mathbf{w}} = -2e(k)\mathbf{x}(k)$$

O que equivale ao emprego de valores instantâneos como estimativa para \mathbf{R} e \mathbf{p} do seguinte modo:

$$\hat{\mathbf{R}} = \mathbf{x}(k)\mathbf{x}^T(k)$$

$$\hat{\mathbf{p}} = d(k)\mathbf{x}(k)$$

Tabela 3.1: Código do algoritmo LMS.

Inicialização $\mathbf{x}(0) = \mathbf{w}(0) = [0 \dots 0]^T$
Para cada k { $e(k) = d(k) - \mathbf{x}^H(k)\mathbf{w}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e^*(k)\mathbf{x}(k)$ }

Desta forma, o gradiente estimado $\hat{\mathbf{g}}_{\mathbf{w}}$ é dado por

$$\begin{aligned} \hat{\mathbf{g}}_{\mathbf{w}} &= -2\hat{\mathbf{p}} + 2\hat{\mathbf{R}}\mathbf{w} = -2d(k)\mathbf{x}(k) + 2\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w} \\ &= -2d(k)\mathbf{x}(k) + 2\mathbf{x}(k)y(k) = -2\mathbf{x}(k)[d(k) - y(k)] = -2e(k)\mathbf{x}(k) \end{aligned} \quad (3.2)$$

Assim, a atualização dos coeficientes do filtro adaptado segundo o algoritmo LMS possui a seguinte equação (comparar com a equação (3.1))

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k) \quad (3.3)$$

A Tabela 3.1 apresenta uma descrição detalhada da implementação do algoritmo LMS, na versão complexa, em que $\mathbf{x}^H(k)$ é a transposição e conjugação dos elementos de $\mathbf{x}(k)$ e $e^*(k)$ representa a conjugação de $e(k)$.

Como foi mencionado anteriormente, é extremamente reduzido o esforço computacional exigido na implementação do algoritmo LMS pois, de uma maneira típica, requer $N + 1$ multiplicações para a atualização dos coeficientes do filtro e N multiplicações para a obtenção do cálculo do erro, o que fornece $2N + 1$ operações para cada iteração. Este forte apelo o tornou bastante popular e o preferido entre os demais algoritmos do processo de filtragem adaptativa.

3.2.1 Propriedades do Algoritmo LMS

As principais propriedades relativas ao comportamento do algoritmo LMS são:

- Valores médios da estimativa do gradiente:

$$E[\hat{\mathbf{g}}_{\mathbf{w}}(k)] = \mathbf{g}_{\mathbf{w}}$$

- Valores médios dos coeficientes:

$$\lim_{k \rightarrow \infty} E[\mathbf{w}(k)] = E[\mathbf{w}(\infty)] = \mathbf{R}^{-1}\mathbf{p}$$

- Condição necessária e suficiente para a convergência:

$$0 < \mu < \frac{1}{\lambda_{max}}$$

Como \mathbf{R} é pelo menos semidefinida positiva, seus autovalores são não-negativos, logo $\lambda_{max} \leq \sum_{i=0}^{N-1} \lambda_i = \text{tr}[\mathbf{R}]$, em que $\text{tr}[\cdot]$ significa traço da matriz.

Então, μ poderá ser escolhido com mais segurança se atender ao critério

$$0 < \mu < \frac{1}{\text{tr}[\mathbf{R}]}$$

Nas situações em que são empregados filtros transversais, o traço de \mathbf{R} corresponde a N vezes a potência do sinal de entrada, deste modo é possível dispor de uma relação bastante útil quando os autovalores de \mathbf{R} não são conhecidos:

$$0 < \mu < \frac{1}{NE[\mathbf{x}^2(k)]}$$

- Velocidade de convergência:

Se o passo de adaptação for pequeno, correspondendo a uma adaptação lenta, é possível fazer a aproximação

$$\tau_{wi} \approx \frac{1}{2\mu\lambda_i}$$

onde τ_{wi} é a constante de tempo associada à convergência do coeficiente w_i do filtro relacionado ao autovalor λ_i .

A curva de aprendizagem do algoritmo LMS é uma soma de progressões geométricas cujas razões são o quadrado das que estão associadas à evolução dos coeficientes do filtro. Desta forma, as constantes de tempo de cada série exponencial valem

$$\tau_{ei} \approx \frac{1}{4\mu\lambda_i} \quad i = 0, 1, \dots, N-1 \quad \mu \ll 1$$

- Desajuste:

$$M \approx \frac{\mu \text{tr}[\mathbf{R}]}{1 - \mu \text{tr}[\mathbf{R}]}$$

Tabela 3.2: Parâmetros do algoritmo LMS para a identificação de sistema.

Parâmetro	Valor
Total de simulações	$S = 500$
Total de iterações	$K = 2000$
Passo de adaptação	$\mu = 0,0059$
Ordem do FIR	$N = 17$

Levando-se em consideração as equações das constantes de tempo e a do desajuste, chega-se à conclusão que a escolha do passo de adaptação é um compromisso entre a velocidade de convergência e o desajuste alcançado pelo sistema no regime permanente.

3.2.2 Identificação de Sistema

Nesta seção o algoritmo LMS atuará como a ferramenta destinada a solucionar o problema da identificação de sistema tendo em consideração o canal de rádio móvel obtido no Capítulo 2, nas condições lá estabelecidas e para uma $SNR = 30$ dB.

O passo de adaptação escolhido será $\mu = 0,0059$ que proporcionará um desajuste de cerca de 10%, um valor considerado razoável [19]. A Tabela 3.2 resume os parâmetros empregados.

A Figura 3.1 apresenta o módulo do erro obtido em função das iterações. Como pode ser observado, o algoritmo está acompanhando a variação das características do canal de forma aceitável.

Na Figura 3.2 encontra-se a evolução dos coeficientes ao longo das iterações.

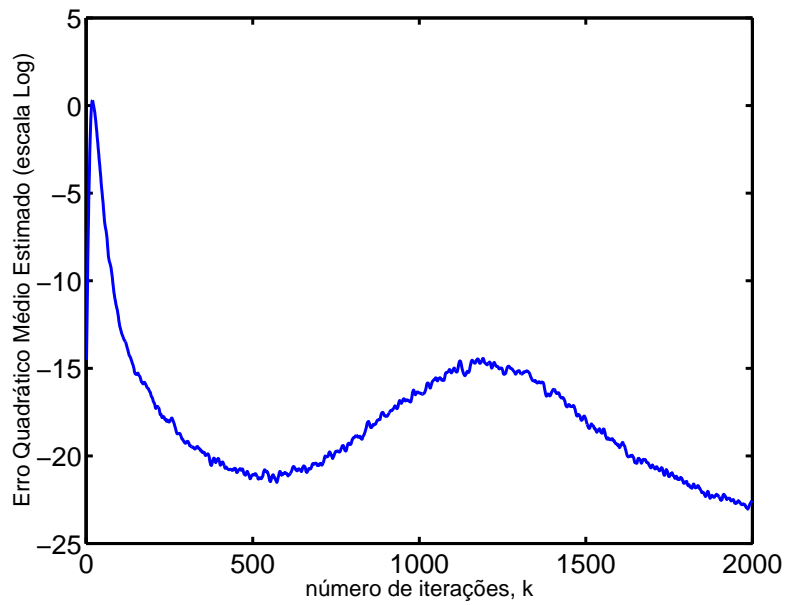


Figura 3.1: Curva de aprendizagem do algoritmo LMS.

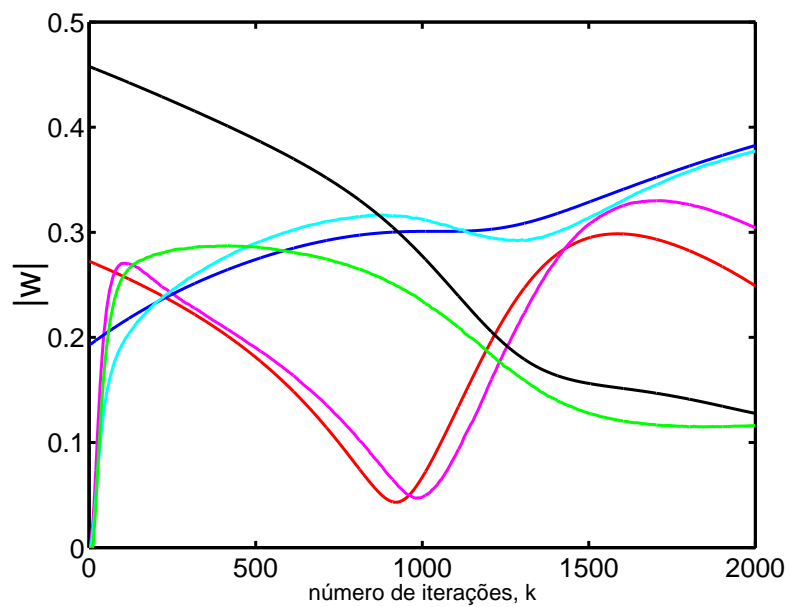


Figura 3.2: Curva de evolução do módulo dos coeficientes 1, 5 e 10 para o LMS.

Tabela 3.3: Tempo de convergência do algoritmo LMS para a identificação de sistema.

Parâmetro	Valor obtido
Tempo de Convergência	$k = 500$

O tempo de convergência obtido a partir da aplicação do algoritmo LMS está apresentado na Tabela 3.3.

3.3 Algoritmo LMS Normalizado

O LMS normalizado é uma variante do algoritmo LMS convencional que utiliza o passo de adaptação variável como uma tentativa de tornar-se mais eficiente no sentido de reduzir o tempo de convergência. O preço a ser pago é normalmente o incremento do desajuste, além de um incremento na complexidade computacional.

Como foi visto na Seção 3.2.1, a escolha do passo de adaptação do algoritmo LMS é uma situação de compromisso entre a velocidade de convergência e o desajuste. Além do mais, um passo definido adequadamente para um certo ambiente pode proporcionar um baixo desempenho do algoritmo ou até mesmo a divergência quando ocorre uma alteração deste ambiente. Isso conduz, na prática, a uma tendência de escolha conservadora pelos usuários do LMS tendo como consequência uma convergência demasiadamente lenta. É esta a motivação para o desenvolvimento de uma técnica em que a escolha do passo de adaptação seja feito de acordo com os dados de entrada [20]. O algoritmo LMS normalizado apresentado por Nagumo e Noda [21] é uma alternativa que atende a este requisito.

No algoritmo LMS normalizado, o passo de adaptação é normalizado relativamente à potência do sinal de entrada em cada iteração. Isto torna o algoritmo menos sensível às variações de potência do sinal de entrada, permitindo seu uso em aplicações nas quais estes valores variam muito como nas aplicações de voz, por exemplo. Em contrapartida, gera o custo de incremento na complexidade computacional especialmente para filtros longos.

A equação que define a atualização dos coeficientes do filtro adaptativo FIR empregando o LMS normalizado é:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu_n}{\delta + \mathbf{x}^T(k)\mathbf{x}(k)} e(k)\mathbf{x}(k)$$

Assim, pode-se encarar o LMS normalizado como um LMS com um passo de adaptação variável, dependente da potência do sinal de entrada e dado por:

$$\mu(k) = \frac{\mu_n}{\delta + \mathbf{x}^T(k)\mathbf{x}(k)}$$

A constante δ foi acrescentada objetivando evitar o surgimento de um passo demasiadamente grande nas situações em que a potência de entrada aproximar-se de zero.

O passo de adaptação fixo μ_n controla o desajuste e pode apresentar valores no intervalo $0 < \mu_n < 2^{-1}$ para garantir a convergência do algoritmo.

Há maneiras distintas de interpretar a normalização:

- tornar o algoritmo independente do nível do sinal de entrada
- ajustar $\mathbf{w}(k+1)$ para proporcionar um erro *a posteriori* nulo para a entrada corrente: $\mathbf{x}^T(k)\mathbf{w}(k+1) = d(k)$

O LMS normalizado converge bem mais rapidamente que o LMS convencional com um custo extra baixo, sendo por isso comumente usado, e tornando-se universal em algumas aplicações. A descrição completa da implementação do algoritmo LMS normalizado, na forma complexa, encontra-se na Tabela 3.4

3.3.1 Identificação de Sistema

Nesta seção, a modelagem do Canal Experimental será realizada pelo algoritmo LMS normalizado para uma $SNR = 30$ dB.

A Tabela 3.5 apresenta os parâmetros empregados nas simulações e as Figuras 3.3 e 3.4, juntamente com a Tabela 3.6 exibem os resultados obtidos.

¹Na prática, utiliza-se $0 < \mu_n < 1$, pois $\mu_n > 1$ gera um desajuste maior sem um correspondente aumento da velocidade de convergência.

Tabela 3.4: Código do algoritmo LMS normalizado.

<p>Inicialização</p> $\mathbf{x}(0) = \mathbf{w}(0) = [0 \dots 0]^T$ <p>escolhe-se μ_n no intervalo $0 < \mu_n \leq 1$</p> <p>escolhe-se $\delta =$ um pequeno valor</p>
<p>Para cada k</p> <p>{</p> $e(k) = d(k) - \mathbf{x}^H(k)\mathbf{w}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu_n}{\delta + \mathbf{x}^H(k)\mathbf{x}(k)} e^*(k)\mathbf{x}(k)$ <p>}</p>

Tabela 3.5: Parâmetros do algoritmo LMS normalizado para a identificação de sistema.

Parâmetro	Valor
Total de simulações	$S = 500$
Total de iterações	$K = 2000$
Passo de adaptação	$\mu_n = 0.25$
Ordem do FIR	$N = 17$

Tabela 3.6: Tempo de convergência do algoritmo LMS normalizado para a identificação de sistema.

Parâmetro	Valor obtido
Tempo de Convergência	$k = 500$

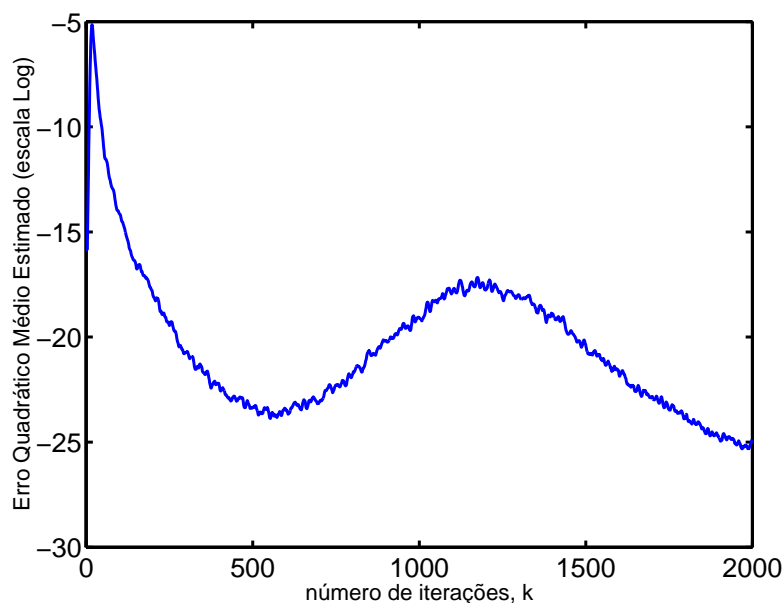


Figura 3.3: Curva de aprendizagem do algoritmo LMS normalizado.

3.4 Algoritmo RLS Convencional

Os algoritmos de adaptação da família RLS (Recursive Least-Squares) apresentam uma elevada velocidade de convergência, independentemente da disparidade dos autovalores da matriz de correlação de entrada. Estes algoritmos proporcionam um bom desempenho quando operam em ambientes variantes no tempo, além de possuírem um desajuste extremamente pequeno ou nulo em regime permanente. Obviamente que por estes aspectos positivos há um preço a ser pago: o incremento na complexidade computacional e alguns problemas de estabilidade.

O algoritmo RLS convencional é implementado através de uma técnica de minimização clássica conhecida como método dos mínimos quadrados, na qual dispensa-se o conhecimento das propriedades estatísticas dos dados. Assim, a função objetivo que deverá ser minimizada é determinística.

O objetivo é determinar os coeficientes do filtro de tal maneira que o somatório dos quadrados da diferença entre os valores do sinal desejado e as estimativas fornecidas pelo filtro sejam o mínimo possível, dentro da janela de observação. A janela de observação equivale ao período de interesse e normalmente assume a forma retangular ou exponencial decrescente. Desta forma, surge um parâmetro responsável por controlar os aspectos relacionados à forma e duração da janela: o

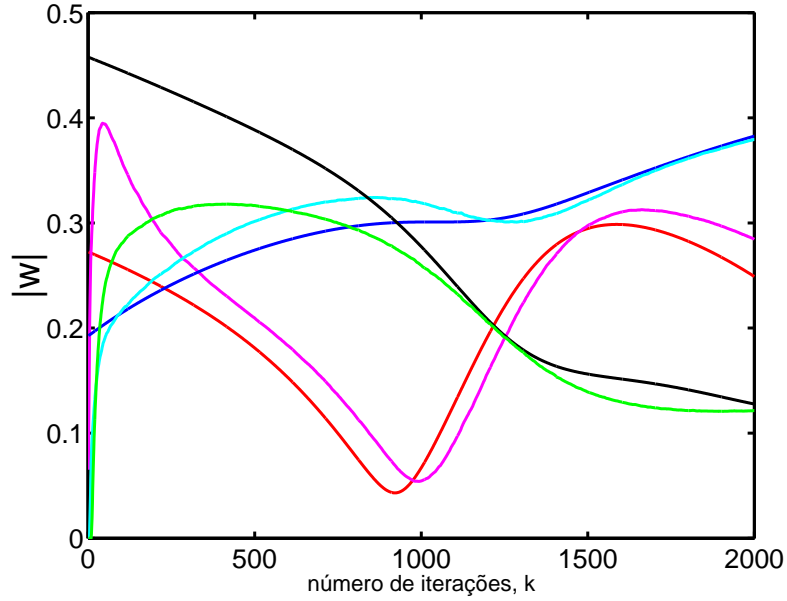


Figura 3.4: Curva de evolução do módulo dos coeficientes 1, 5 e 10 para o LMS normalizado.

chamado fator de esquecimento λ , o qual deve ser ajustado no intervalo $0 \ll \lambda \leq 1$. Quando $\lambda = 1$, a janela é retangular e todos os erros anteriores ponderarão da mesma forma, equivalendo a dizer que o algoritmo tem uma memória infinita. Ao contrário, à medida que λ aproxima-se de zero os erros passados desempenham um papel de baixa influência no resultado final.

Do exposto, a função objetivo do algoritmo apresenta-se da seguinte forma:

$$F(e(k)) = \xi^d(k) = \sum_{i=0}^k \lambda^{k-i} e^2(k) = \sum_{i=0}^k \lambda^{k-i} [d(i) - \mathbf{x}^T(i)\mathbf{w}(k)]^2$$

É possível mostrar que esta função descrita por este parâmetro no espaço multidimensional das variáveis $\mathbf{w}(k)$ é convexa, isto é, a função descrita por $\xi^d(k)$ em termos dos coeficientes do filtro $\mathbf{w}(k)$ apresenta um único mínimo global e nenhum mínimo local. Sendo assim, é possível determinar este ponto simplesmente igualando o gradiente de $\xi^d(k)$ a zero.

$$\frac{\partial \xi^d(k)}{\partial \mathbf{w}(k)} = -2 \sum_{i=0}^k \mathbf{x}(i) [d(i) - \mathbf{x}^T(i)\mathbf{w}(k)] = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Esta relação representa um simples sistema com $k + 1$ equações e com o mesmo número de incógnitas $\mathbf{w}(k)$, cuja solução é

$$\mathbf{w}(k) = \left[\sum_{i=0}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \right]^{-1} \sum_{i=0}^k \lambda^{k-i} \mathbf{x}(i) d(i) \quad (3.4)$$

$$\mathbf{w}(k) = \mathbf{R}_D^{-1}(k) \mathbf{p}_D(k) \quad (3.5)$$

onde:

$\mathbf{R}_D^{-1}(k) = \left[\sum_{i=0}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \right]^{-1}$: matriz determinística de autocorrelação do sinal de entrada

$\mathbf{p}_D(k) = \sum_{i=0}^k \lambda^{k-i} \mathbf{x}(i) d(i)$: vetor determinístico de correlação cruzada entre o sinal desejado e a seqüência de entrada.

Partindo das definições de $\mathbf{R}_D^{-1}(k)$ e de $\mathbf{p}_D(k)$, pode-se facilmente obter as estimativas de forma recursiva para os coeficientes do filtro

$$\mathbf{w}(k) = \mathbf{w}(k+1) + \mathbf{R}_D^{-1}(k) e(k) \mathbf{x}(k)$$

onde:

$e(k)$ é o erro de estimação *a priori*. É o erro obtido quando se estima o sinal desejado com os coeficientes do filtro antes de serem atualizados.

Como a obtenção direta da inversa de $\mathbf{R}_D^{-1}(k)$ demandaria uma grande quantidade de cálculo a cada iteração, da ordem do cubo do número de coeficientes do filtro $O[N^3]$. Esta operação é evitada fazendo uso da recursividade proporcionada pela definição de $\mathbf{R}_D^{-1}(k)$ e pela relação algébrica conhecida como lema da inversão de matrizes

$$(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{B} (\mathbf{D} \mathbf{A}^{-1} \mathbf{B} + \mathbf{C}^{-1})^{-1} \mathbf{D} \mathbf{A}^{-1}$$

Fazendo $\mathbf{A} = \lambda \mathbf{R}_D^{-1}(k-1)$, $\mathbf{B} = \mathbf{x}(k)$, $\mathbf{C} = 1$ e $\mathbf{D} = \mathbf{x}(k)$, obtém-se

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

Tabela 3.7: Código do algoritmo RLS.

<p>Inicialização</p> $\mathbf{S}_D(-1) = \delta \mathbf{I}$ <p>onde δ corresponde ao recíproco da potência estimada do sinal de entrada</p> $\mathbf{x}(-1) = \mathbf{w}(-1) = [0 \dots 0]^T$
<p>Para cada k</p> <p>{</p> $e(k) = d(k) - \mathbf{x}^H(k)\mathbf{w}(k)$ $\Psi(k) = \mathbf{S}_D(k-1)\mathbf{x}(k)$ $\mathbf{S}_D(k) = \frac{1}{\lambda} \left[\mathbf{S}_D(k-1) - \frac{\Psi(k)\Psi^H(k)}{\lambda + \Psi^H(k)\mathbf{x}(k)} \right]$ $\mathbf{w}(k+1) = \mathbf{w}(k) + e^*(k)\mathbf{S}_D(k)\mathbf{x}(k)$ <p>}</p>
<p>Se necessário, calcular</p> $y(k) = \mathbf{w}^H(k+1)\mathbf{x}(k)$ $e'(k) = d(k) - y(k)$

Esta solução reduz a complexidade computacional para $O[N^2]$ em troca de uma possível eclosão de problemas numéricos que podem comprometer a estabilidade do algoritmo.

O algoritmo completo do RLS convencional, na versão complexa, é mostrado na Tabela 3.7.

3.4.1 Propriedades do Algoritmo RLS Convencional

- O algoritmo RLS convencional converge para a solução de Wiener se o fator de esquecimento for unitário e os sinais envolvidos forem ergódicos e estacionários
- Quando um ruído de medição com média nula e descorrelacionado com o sinal desejado está presente, encontram-se as seguintes relações:
 - $E[e(k)] = 0$: média do erro de estimação *a priori*
 - $E[e^2(k)] = \sigma_n^2$: valor quadrático médio do erro de estimação *a priori* em que σ_n^2 é a variância do ruído de medição caso não haja *undermodeling*.

Tabela 3.8: Parâmetros do algoritmo RLS convencional para a identificação de sistema.

Parâmetro	Valor
Total de simulações	$S = 500$
Total de iterações	$K = 2000$
Fator de esquecimento	$\lambda = 0.95$
Ordem do FIR	$N = 17$

$$- E[e^{l^2}(k)] = \left\{ 1 - 2(N + 1) \left[\frac{1-\lambda}{1-\lambda^{k+1}} \right] \right\} \sigma_n^2: \text{ valor quadrático médio do erro de estimação } a \text{ posteriori}$$

- O regime estacionário é atingido em cerca de $2N$ iterações e a velocidade de convergência independe do “número de condicionamento ” da matriz de autocorrelação \mathbf{R} se a relação sinal-ruído for elevada
- No regime estacionário o desajuste apresenta a seguinte fórmula

$$M = \frac{\xi_{exc}}{\xi_{min}} = (N + 1) \frac{1 - \lambda}{1 + \lambda} \left(1 + \frac{1 - \lambda}{1 + \lambda} \kappa \right)$$

onde $\kappa = \frac{\sigma_s^2}{(\sigma_z^2)^2}$ é dependente das estatísticas do sinal de entrada

É fácil perceber que se o fator de esquecimento $\lambda = 1$, o desajuste é nulo; se não, tenderá para um valor pequeno se λ estiver nas proximidades da unidade:

$$M \approx (N + 1)(1 - \lambda), \text{ para } \lambda \text{ próximo de } 1$$

3.4.2 Identificação de Sistema

Nesta seção é a vez do algoritmo RLS convencional ser testado para a aplicação de interesse para uma $SNR = 30$ dB.

Os parâmetros empregados nas simulações encontram-se na Tabela 3.8 e os resultados obtidos são mostrados através dos gráficos 3.5 e 3.6 e da Tabela 3.9.

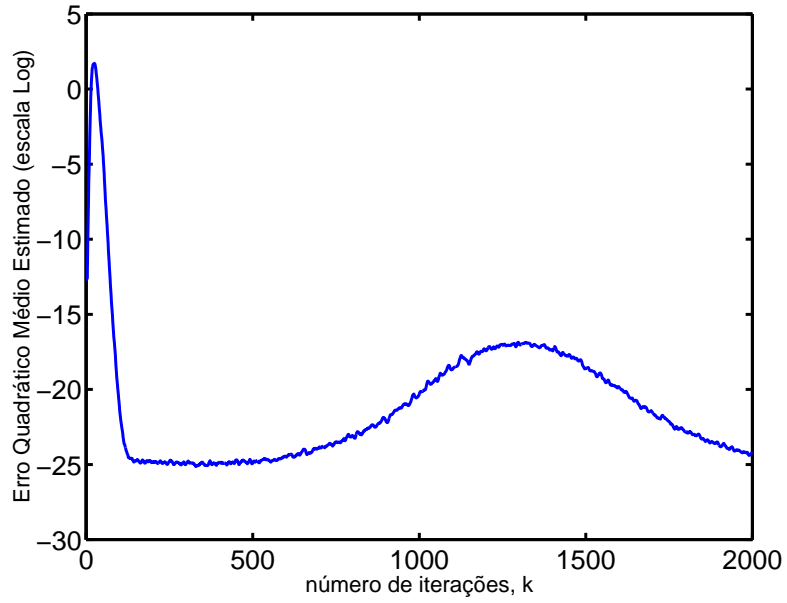


Figura 3.5: Curva de aprendizagem do algoritmo RLS.

Tabela 3.9: Tempo de convergência do algoritmo RLS convencional para a identificação de sistema.

Parâmetro	Valor obtido
Tempo de Convergência	$k = 130$

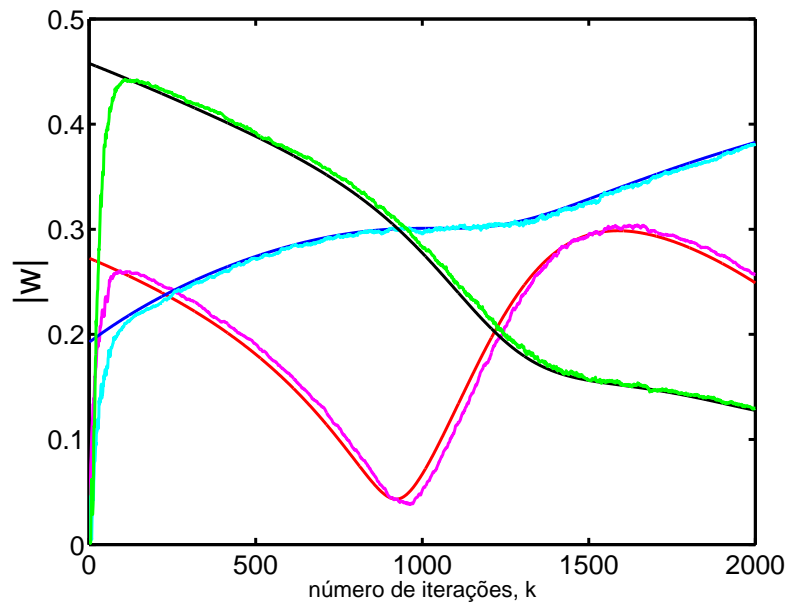


Figura 3.6: Curva de evolução do módulo dos coeficientes 1, 5 e 10 para o RLS.

3.5 Resumo

Neste capítulo foi apresentada uma concisa descrição dos algoritmos clássicos de minimização amplamente empregados no campo da filtragem adaptativa. Dois destes, o LMS e o LMS normalizado, buscam a solução ótima através da minimização do erro quadrático médio utilizando para este fim, estimativas do gradiente da função objetivo.

Enquanto que o outro algoritmo apresentado, o RLS convencional, procura convergir até a solução ótima tentando minimizar a soma dos quadrados dos valores estimados a cada iteração empregando o método dos mínimos quadrados.

As principais propriedades e características dos algoritmos foram sucintamente abordadas, fornecendo sempre uma formulação matemática útil para a obtenção de resultados analíticos a fim de que possam ser confrontados com aqueles obtidos na prática.

Capítulo 4

Avaliação e Otimização do Algoritmo SM-Affine Projection

4.1 Introdução

Neste capítulo o algoritmo SM-AP (Set-Membership of Affine Projection) apresentado por Werner [3] será brevemente descrito e suas principais propriedades serão expostas. Inicialmente, será efetuada uma abordagem a cerca da técnica de filtragem conhecida como set-membership que tem a habilidade de reduzir a complexidade computacional. Em seguida, será mostrado o algoritmo SM-AP em sua forma original, tal como sugerida por Werner e, a partir deste, serão realizadas as simulações de identificação do canal de rádio móvel de referência. Na seqüência, serão efetuadas alterações no algoritmo visando à obtenção de um melhor desempenho. Por fim, novas simulações serão realizadas empregando o algoritmo na versão otimizada para a aplicação de interesse e seus resultados servirão de base para os capítulos seguintes.

4.2 Filtragem do tipo Set-Membership

A principal característica do filtro adaptativo é sua habilidade de auto-aprendizagem. Isto acontece automaticamente como decorrência da minimização da função objetivo, que é uma função relacionada com o erro de estimação. Acontece, porém, que um dos critérios básicos que definem uma função objetivo é a exigência da

não-negatividade, logo seu valor mínimo deverá ser zero. Há aplicações, no entanto, que a melhor solução é quando a função de custo esteja próximo da origem. Esta situação está normalmente relacionada diretamente com o fato de o erro de estimação corrente encontrar-se limitado a um nível determinístico γ . É mostrado em [22] que para a obtenção de uma equalização perfeita basta que este limiar $\gamma = d_{min}/2$, em que d_{min} é a distância mínima entre os sinais da constelação. Assim, este limiar pode ser usado como um parâmetro de projeto e seu valor variar de acordo com a aplicação.

Uma consequência direta do uso de γ como uma limitação do erro é que não apenas um filtro pode atender às especificações mas um conjunto de filtros que pertencem à chamada membresia do conjunto (*set-membership*) exato Ψ_k , definida como:

$$\Psi(k) = \bigcap_{i=1}^k H(i) \quad (4.1)$$

em que:

$$H(k) = \{\mathbf{w}(k) \in R^N : |d(k) - \mathbf{w}^T(k)\mathbf{x}(k)| \leq \gamma\}$$

A outra consequência, de grande importância, é o fato de que se a estimativa dos coeficientes $\mathbf{w}(k)$ e o vetor de entrada $\mathbf{x}(k)$ produzem um erro inferior ao limiar projetado; então, não há a necessidade de se realizar uma atualização dos coeficientes, isto é,

$$\mathbf{w}(k+1) = \mathbf{w}(k), \text{ se } |d(k) - \mathbf{w}^T(k)\mathbf{x}(k)| \leq \gamma$$

Esta estratégia consegue reduzir na média a complexidade computacional do algoritmo que a utiliza.

4.3 O Algoritmo Set-Membership Affine Projection

O aclamado algoritmo LMS com sua robustez e baixa complexidade tem-se tornado enormemente popular. Contudo apresenta o inconveniente de ter sua velocidade de convergência fortemente dependente da disparidade entre os auto-valores da matriz de correlação do sinal de entrada. Já o algoritmo RLS está entre os algoritmos

da filtragem adaptativa mais velozes no que concerne à velocidade de convergência. Porém, problemas de divergência e especialmente sua alta complexidade computacional podem torná-lo desaconselhável em certas aplicações. Esta motivação conduziu à busca por alternativas que resultaram em algoritmos com a complexidade e velocidade de convergência intermediárias ao LMS e o RLS convencional. O Set-Membership Affine Projection é um destes algoritmos.

A técnica de filtragem do tipo set-membership aliada ao conceito de reuso das informações passadas deu origem ao SM-AP, um algoritmo normalizado com seletividade de dados que consiste na generalização dos algoritmos recentemente propostos SM-NLMS (Set-Membership Normalized Least Mean Square) e SM-BNDRLMS (Set-Membership Binormalized Data Reuse Least Mean Square) através da expansão do conjunto de restrições formado a partir de pares de dados (\mathbf{x}, d) passados, possibilitando ser a escolha do número de projeções P um parâmetro de projeto.

Diferentemente da maioria dos algoritmos da filtragem adaptativa, o SM-AP não conflita a velocidade de convergência com o desajuste e a complexidade computacional [3].

O conjunto de *set-membership* exato definido em (4.1) pode ser desmembrado da seguinte forma:

$$\Psi(k) = \bigcap_{i=1}^{k-P} H(i) \bigcap_{j=k-P+1}^k H(j) = \Psi^{k-P}(k) \bigcap \Psi^P(k)$$

em que:

$\Psi^{k-P}(k)$ é a interseção dos primeiros $k - P$ conjuntos de restrição

$\Psi^P(k)$ é a interseção dos últimos P conjuntos de restrição

O objetivo do algoritmo é manter o próximo vetor de coeficientes $\mathbf{w}(k+1)$ o mais perto possível do atual $\mathbf{w}(k)$, enquanto obriga o novo vetor a pertencer aos P últimos conjuntos de restrição, isto é, $\mathbf{w}(k+1) \in \Psi^P(k)$.

A condição necessária para garantir que $\mathbf{w}(k+1) \in \Psi^P(k)$ é

$$d(k-i+1) - \mathbf{w}^T(k+1)\mathbf{x}(k-i+1) = g_i(k), \quad i = 1, 2, \dots, P \text{ com } |g_i(k)| \leq \gamma \quad (4.2)$$

Então o critério de otimização para a atualização do vetor coeficiente quando $\mathbf{w}(k+1) \notin \Psi^P(k)$ é estabelecido como:

$$\min \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \text{ sujeito a}$$

$$\mathbf{d}_{ap}(k) - \mathbf{X}_{ap}^T(k)\mathbf{w}(k+1) = \mathbf{g}(k)$$

onde:

$$\begin{aligned} \mathbf{d}_{ap}(k) &= [d(k) \ d(k-1) \ \dots \ d(k-P+1)]^T \\ \mathbf{X}_{ap}^T &= [\mathbf{x}(k) \ \mathbf{x}(k-1) \ \dots \ \mathbf{x}(k-P+1)] \text{ com} \\ \mathbf{x}(k) &= [x(k) \ x(k-1) \ \dots \ x(k-N)]^T \\ \mathbf{g}(k) &= [g_1(k) \ g_2(k) \ \dots \ g_P(k)]^T \end{aligned} \quad (4.3)$$

Aplicando o método dos multiplicadores de Lagrange, a função a ser minimizada passa a ser:

$$F[\mathbf{w}(k+1)] = \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 + \boldsymbol{\lambda}_{ap}^T(k)[\mathbf{d}_{ap}(k) - \mathbf{X}_{ap}^T(k)\mathbf{w}(k+1) - \mathbf{g}(k)]$$

onde $\boldsymbol{\lambda}_{ap}$ é um vetor dos multiplicadores de Lagrange.

$$\text{grad}\{F[\mathbf{w}(k+1)]\} = 2\mathbf{w}(k+1) - 2\mathbf{w}(k) - \boldsymbol{\lambda}_{ap}^T \mathbf{X}_{ap}^T(k) = 0$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\boldsymbol{\lambda}_{ap}^T(k)}{2} \mathbf{X}_{ap}^T(k)$$

Como $\mathbf{d}_{ap}(k) - \mathbf{X}_{ap}^T(k)\mathbf{w}(k+1) = \mathbf{g}(k)$

$$\mathbf{d}_{ap}(k) - \mathbf{X}_{ap}^T(k) \left[\mathbf{w}(k) + \frac{\boldsymbol{\lambda}_{ap}^T(k) \mathbf{X}_{ap}^T(k)}{2} \right] = \mathbf{g}(k)$$

$$\mathbf{d}_{ap}(k) - \mathbf{X}_{ap}^T(k)\mathbf{w}(k) - \frac{\mathbf{X}_{ap}^T(k) \boldsymbol{\lambda}_{ap}^T(k) \mathbf{X}_{ap}^T(k)}{2} = \mathbf{g}(k)$$

$$\frac{\mathbf{X}_{ap}^T(k) \boldsymbol{\lambda}_{ap}^T(k) \mathbf{X}_{ap}^T(k)}{2} = \mathbf{e}_{ap}(k) - \mathbf{g}(k)$$

$$\frac{\boldsymbol{\lambda}_{ap}^T(k)}{2} = [\mathbf{X}_{ap}^T(k) \mathbf{X}_{ap}(k)]^{-1} [\mathbf{e}_{ap}(k) - \mathbf{g}(k)]$$

Logo,

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mathbf{X}_{ap}(k) [\mathbf{X}_{ap}^T(k) \mathbf{X}_{ap}(k)]^{-1} [\mathbf{e}_{ap}(k) - \mathbf{g}(k)] & \text{se } |e(k)| > \gamma \\ \mathbf{w}(k) & \text{senão} \end{cases}$$

onde $\mathbf{e}_{ap}(k) = [e(k) \ \epsilon(k-1) \ \dots \ \epsilon(k-P+1)]^T$ e $\epsilon(k-i) = d(k-i) - \mathbf{x}^T(k-i)\mathbf{w}(k)$ que por sua vez é denominado erro *a posteriori* na iteração $k-i$.

A definição de valores para os parâmetros $g_i(k)$ desempenha um papel fundamental para o desempenho do algoritmo. Quando o SM-AP foi apresentado, foram escolhidos valores para

$$\begin{aligned} g_i(k) &= \epsilon(k-i+1) \quad \text{se } i \neq 1 \quad \text{e} \\ g_1 &= \gamma \frac{e(k)}{|e(k)|} \end{aligned} \quad (4.4)$$

Desta forma $\mathbf{e}_{ap}(k) - \mathbf{g}(k)$ resulta em:

$$\begin{bmatrix} e(k) \\ \epsilon(k-1) \\ \epsilon(k-2) \\ \vdots \\ \epsilon(k-P+1) \end{bmatrix} - \begin{bmatrix} e(k) \frac{\gamma}{|e(k)|} \\ \epsilon(k-1) \\ \epsilon(k-2) \\ \vdots \\ \epsilon(k-P+1) \end{bmatrix} = \begin{bmatrix} e(k) \left[1 - \frac{\gamma}{|e(k)|} \right] \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = e(k)\alpha(k)\mathbf{u}_1$$

em que $\alpha(k) = 1 - \frac{\gamma}{|e(k)|}$ e $\mathbf{u}_1 = [1 \ 0 \ \dots \ 0]^T$ fornecendo a equação de atualização final:

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) + \mathbf{X}_{ap}(k)[\mathbf{X}_{ap}^T(k)\mathbf{X}_{ap}(k)]^{-1}e(k)\alpha(k)\mathbf{u}_1 \\ \alpha(k) &= \begin{cases} 1 - \frac{\gamma}{|e(k)|}, & \text{se } |e(k)| > \gamma \\ 0, & \text{se } |e(k)| \leq \gamma \end{cases} \end{aligned}$$

O algoritmo SM-AP complexo, encontra-se esquematizado na Tabela 4.1.

Pode ser facilmente percebido que escolhendo o limiar $\gamma = 0$ obtém-se o algoritmo affine projection convencional com o passo de adaptação unitário. Nota-se também que os algoritmos SM-NLMS e SM-BNRDLMS são casos particulares do SM-AP correspondentes às escolhas de $P = 1$ e $P = 2$, respectivamente.

4.4 Identificação de Sistema com o SM-AP

Nesta seção o canal de referência será modelado através do algoritmo SM-AP. As condições são as mesmas que foram submetidos os demais algoritmos para uma $SNR = 30$ dB.

A Tabela 4.2 contém os parâmetros utilizados nas simulações e os resultados encontram-se nas Figuras 4.1 e 4.2 e na Tabela 4.3.

Tabela 4.1: Código do algoritmo set-membership affine projection.

<p>Inicialização</p> $\mathbf{x}(0) = \mathbf{w}(0) = [0 \dots 0]^T$ <p>escolhe-se $\delta =$ um pequeno valor</p>
<p>Para cada k</p> <p>{</p> $e(k) = d(k) - \mathbf{x}^H(k)\mathbf{w}(k)$ <p>se $e(k) > \gamma$</p> <p>{</p> $\alpha(k) = 1 - \frac{\gamma}{ e(k) }$ $\mathbf{t}(k) = [\mathbf{X}_{ap}^H(k)\mathbf{X}_{ap}(k) + \delta\mathbf{I}]^{-1}\alpha(k)e^*(k)\mathbf{u}_1$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}_{ap}(k)\mathbf{t}(k)$ <p>}</p> <p>}</p> <p>senão</p> <p>{</p> $\mathbf{w}(k+1) = \mathbf{w}(k)$ <p>}</p> <p>}</p>

Tabela 4.2: Parâmetros do algoritmo SM-AP para a identificação de sistema.

Parâmetro	Valor
Total de simulações	$S = 500$
Total de iterações	$K = 2000$
Número de projeções	$P = 5$
Limiar para o erro	$\gamma = \sqrt{5\sigma_n^2}$
Ordem do FIR	$N = 17$

Tabela 4.3: Tempo de convergência do algoritmo SM-AP para a identificação de sistema.

Parâmetro	Valor obtido
Tempo de Convergência	$k = 100$

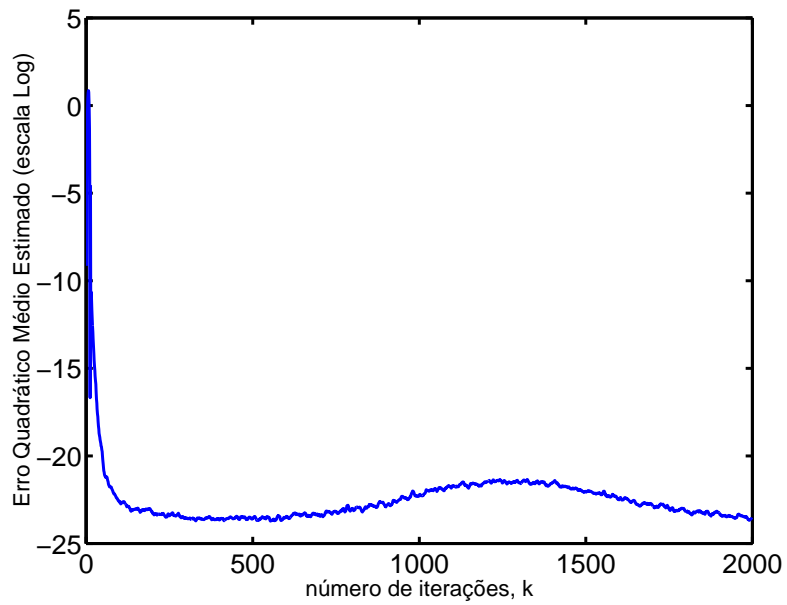


Figura 4.1: Curva de aprendizagem do algoritmo SM-AP.

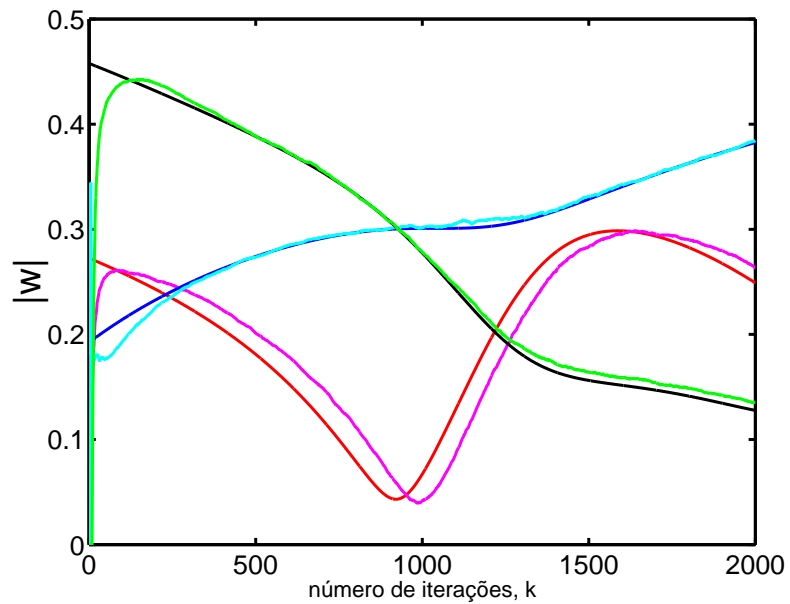


Figura 4.2: Curva de evolução do módulo dos coeficientes 1, 5 e 10 para o SM-AP.

4.5 Otimização do Set-Membership Affine Projection

Como mencionado em [3], o algoritmo SM-Affine Projection não apresenta o compromisso entre a velocidade de convergência e o desajuste, como é comum na maioria dos algoritmos da filtragem adaptativa. Simulações indicam que este algoritmo combina a rápida convergência do algoritmo BNDRLMS com o baixo desajuste propiciado pelo algoritmo SM-NLMS. Além do mais, o emprego da técnica de filtragem do tipo *set-membership* conduz a uma acentuada redução na complexidade computacional do algoritmo através da diminuição da quantidade de atualizações necessárias.

É justamente nesta figura de desempenho — a complexidade computacional — que as alterações ao algoritmo proposto deverão ser realizadas de modo a proporcionar uma redução ainda mais acentuada. Isto é de grande importância, uma vez que a complexidade computacional do algoritmo adaptativo é um dos maiores fatores responsáveis pelo consumo de energia dos filtros de recepção.

O algoritmo SM-Affine Projection, tal como foi proposto, mantém constante o número de projeções ou fator de reuso de dados P durante toda a operação, de modo que toda vez que se requer uma atualização dos coeficientes faz-se necessária uma inversão de uma matriz da ordem P . Se, no entanto, variarmos a quantidade de projeções em função do fator $\alpha(k)$, de forma que à medida que $\alpha(k)$ tenda para zero o número de projeções tenda para 1, o algoritmo SM-Affine Projection tenderá para o SM-NLMS, nesta situação. Por outro lado, à proporção que $\alpha(k)$ aproxime-se da unidade, o número de projeções deverá ser incrementado até um valor máximo P_{max} . Se escolhermos P_{max} como sendo igual ao número de projeções do SM-Affine Projection convencional P , obteremos um algoritmo com complexidade numérica seguramente inferior a este algoritmo, pois a inversão matricial, quando necessária, será realizada para uma matriz da ordem inferior ou igual a P . Em outras palavras, deve-se quantizar a região $[0, 1]$ em P_{vr} regiões que representem um certo número de projeções. O algoritmo, assim resultante, pode ser visto como um SM-Affine Projection com fator de reuso variável, o SM-APVDR (*Set-Membership Affine Projection with Variable Data-Reuse factor*).

Obviamente, existe uma infinidade de maneiras em que pode ser realizada a quantização da região $[0, 1]$. De todas estas maneiras possíveis, a solução ótima será aquela capaz de propiciar uma curva de aprendizagem apresentando velocidade de convergência e desajuste melhores ou iguais aqueles obtidos com a aplicação do algoritmo SM-AP convencional e, ao mesmo tempo, empregue o menor número de projeções, nas vezes em que for necessária a atualização. A equação (4.5) representa o modo como este mapeamento será efetuado, neste trabalho. Sua formulação foi obtida através do método de tentativas e erros.

$$P_{vr}(\alpha(k)) = \left\lceil \frac{P}{1 - e^{-\frac{P}{4}}} (1 - e^{-\frac{P(\alpha(k)-0.2)}{4}}) + 1 \right\rceil, \text{ com } 1 \leq P_{vr} \leq P \quad (4.5)$$

em que $\lceil \cdot \rceil$ significa arredondar para o inteiro sucessor.

A Figura 4.3 mostra a quantização da região $[0, 1]$ para o caso em que $P = 5$.

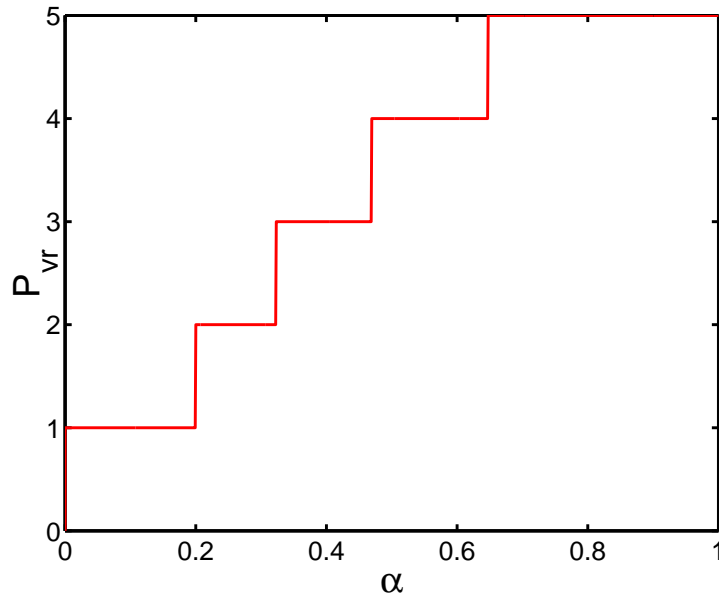


Figura 4.3: Curva para a quantização da região que contém $\alpha(k)$ para o caso em que $P = 5$.

Isto significa que, para esta situação, teremos os 5 níveis da Tabela 4.4. Desta forma, enquanto $\alpha(k) \leq 0.2000$ será utilizada uma projeção para o algoritmo. Por outro lado, se $\alpha(k) > 0.6471$, será empregado o número máximo de projeções, isto é, cinco.

Tabela 4.4: Valores limites para a quantização da região que contém $\alpha(k)$ para o caso em que $P = 5$.

$\alpha(k)$	P_{vr}
0.2000	1
0.3232	2
0.4688	3
0.6471	4
1.0000	5

O uso variável do número de projeções pode conduzir-nos a dois casos distintos:

Caso 1: Se o P_{vr_new} for menor ou igual ao número de projeções utilizadas anteriormente P_{vr_old} , então todos os elementos do vetor $\mathbf{g}(k)$ — definido em (4.3)— exceto o primeiro, podem ser escolhidos como os erros *a posteriori* $\epsilon(k - i)$ como foi realizado anteriormente conforme equação (4.4). Assim, a atualização usual do SM-AP poderá ser empregada, isto é, o vetor unitário $\mathbf{u}_1 = [1 \ 0 \dots 0]^T$ poderá ser utilizado.

Caso 2: Se P_{vr_new} for maior que P_{vr_old} poderá dar origem a situações em que o valor absoluto de algum $\epsilon(k - i)$ seja maior que γ (para as projeções excedentes). Uma vez que os elementos de $\mathbf{g}(k)$ devem atender à restrição (4.2), ou seja, $|g_i(k)| \leq \gamma$, então para estas situações a solução trivial corresponderia a adotar o valor nulo para estes específicos elementos de $\mathbf{g}(k)$. Entretanto, optaremos por fazer um paralelo com a equação (4.4) e adotar para estes elementos os valores dados por $g_i(k) = \gamma \frac{\epsilon(k-i)}{|\epsilon(k-i)|}$.

A Tabela 4.5 apresenta o pseudo-código para a implementação do algoritmo SM-APVDR, na versão complexa.

Tabela 4.5: Código do algoritmo set-membership affine projection com fator de reuso variável (SM-APVDR).

<p>Inicialização</p> $\mathbf{x}(0) = \mathbf{w}(0) = [0 \dots 0]^T$ <p>escolhe-se $\delta =$ um pequeno valor</p>
<p>Para cada k</p> <p>{</p> $e(k) = d(k) - \mathbf{x}^H(k)\mathbf{w}(k)$ <p>se $e(k) > \gamma$</p> <p>{</p> $\alpha(k) = 1 - \frac{\gamma}{ e(k) }$ $P_{vr} = \left\lceil \frac{P}{1-e^{-\frac{P}{4}}} (1 - e^{-\frac{P(\alpha(k)-0.2)}{4}}) + 1 \right\rceil$ <p>se $P_{vr} > P_{vr_old}$</p> <p>{</p> $\epsilon(k-i) = d(k-i) - \mathbf{x}^H(k-i)\mathbf{w}(k)$ <p>se $\epsilon(k-i) > \gamma$ então $u(k-i) = 1 - \frac{\gamma}{ \epsilon(k-i) }$</p> <p>}</p> $\mathbf{X}_{ap}^T = [\mathbf{x}(k) \mathbf{x}(k-1) \dots \mathbf{x}(k-P_{vr}+1)]$ $\mathbf{t}(k) = [\mathbf{X}_{ap}^H(k)\mathbf{X}_{ap}(k) + \delta\mathbf{I}]^{-1}\alpha(k)e^*(k)\mathbf{u}_1$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}_{ap}(k)\mathbf{t}(k)$ $P_{vr_old} = P_{vr}$ <p>}</p> <p>senão</p> <p>{</p> $\mathbf{w}(k+1) = \mathbf{w}(k)$ <p>}</p> <p>}</p>

Tabela 4.6: Parâmetros do algoritmo SM-APVDR para a identificação de sistema.

Parâmetro	Valor
Total de simulações	$S = 500$
Total de iterações	$K = 2000$
Número máximo de projeções	$P = 5$
Limiar para o erro	$\gamma = \sqrt{5\sigma_n^2}$
Ordem do FIR	$N = 17$

Tabela 4.7: Tempo de convergência do algoritmo SM-APVDR para a identificação de sistema.

Parâmetro	Valor obtido
Tempo de Convergência	$k = 100$

4.6 Identificação de Sistema com o Algoritmo SM-Affine Projection com Fator de Reuso Variável

Com o algoritmo agora modificado segundo os critérios estabelecidos na seção anterior, procederemos a identificação do Canal Experimental com os parâmetros constantes na Tabela 4.6.

Os resultados obtidos encontram-se apresentados nas Figuras 4.4 e 4.5 e na Tabela 4.7.

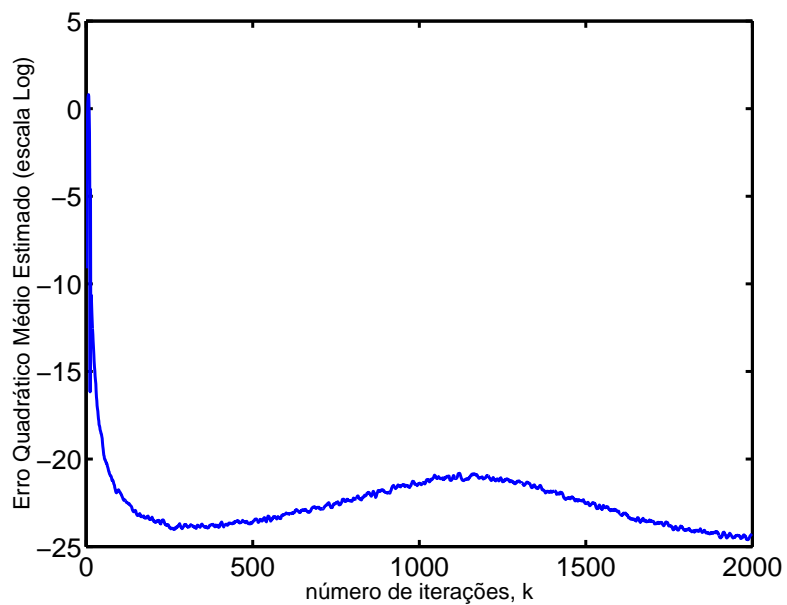


Figura 4.4: Curva de aprendizagem do algoritmo SM-APVDR.

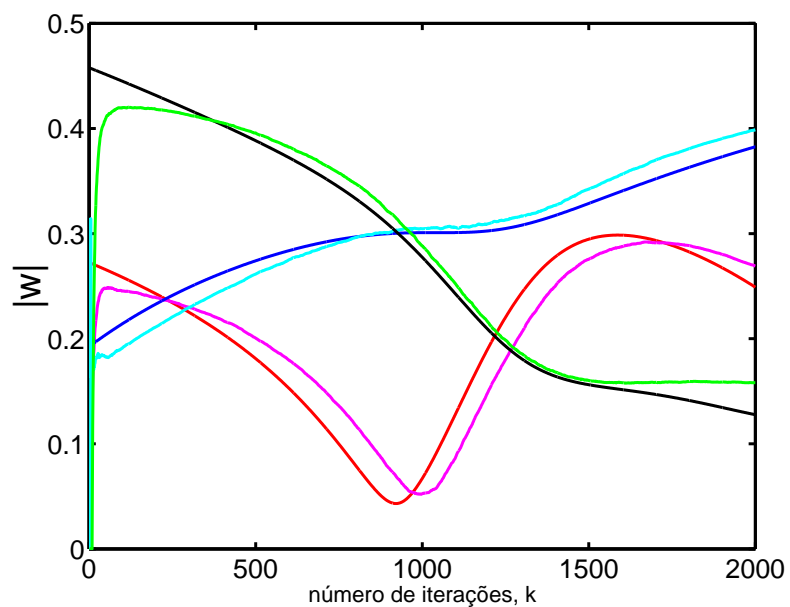


Figura 4.5: Curva de evolução do módulo dos coeficientes 1, 5 e 10 para o SM-APVDR.

4.7 Resumo

Este capítulo fez uma breve descrição da técnica de filtragem do tipo *set-membership* responsável por promover uma substancial redução da complexidade computacional aos algoritmos da filtragem adaptativa que a utilizam através da diminuição da demanda de atualização dos coeficientes do filtro.

Posteriormente, foi abordado o algoritmo SM-Affine Projection convencional recentemente apresentado, enfatizando suas principais características quanto à velocidade de adaptação, desajuste e complexidade computacional.

Finalmente, foi sugerida uma alteração no SM-Affine Projection convencional de forma que este passe a ter o número de projeções variáveis em função de uma quantização da região que contém os valores possíveis para o fator $\alpha(k)$ de tal sorte que, à medida que o algoritmo vai convergindo, sua complexidade computacional diminui.

Capítulo 5

Comparações

5.1 Introdução

Este capítulo destina-se à realização de uma análise comparativa entre o algoritmo SM-Affine Projection com fator de reuso variável (SM-APVDR) e a versão convencional do SM-AP. Será apresentado também o confronto entre seu desempenho e aqueles fornecidos pelos algoritmos clássicos da filtragem adaptativa, discutidos no Capítulo 3, na tarefa de identificar o canal de rádio móvel — canal experimental — obtido no Capítulo 2.

5.2 SM-APVDR *versus* SM-AP

Nesta seção, o algoritmo SM-AP com reuso variável e o SM-AP convencional serão aplicados ao problema de identificação do canal experimental para três valores diferentes da relação sinal-ruído $SNR = 30$ dB, 40 dB e 50 dB.

A Figura 5.1 mostra as curvas de aprendizagem obtidas através de 500 simulações para ambos os algoritmos em uma situação com $SNR = 30$ dB. O limiar para o erro de estimação foi ajustado para $\gamma = \sqrt{5\sigma_n^2}$, onde σ_n^2 é a variância do ruído gaussiano aditivo e, o número de reusos de dados para o SM-AP convencional foi escolhido como sendo $P = 5$, sendo esta a mesma quantidade escolhida para o P_{max} do SM-APVDR.

Pode-se notar claramente na figura que o desempenho de ambos os algoritmos são equivalentes no que se refere à velocidade de convergência e desajuste.

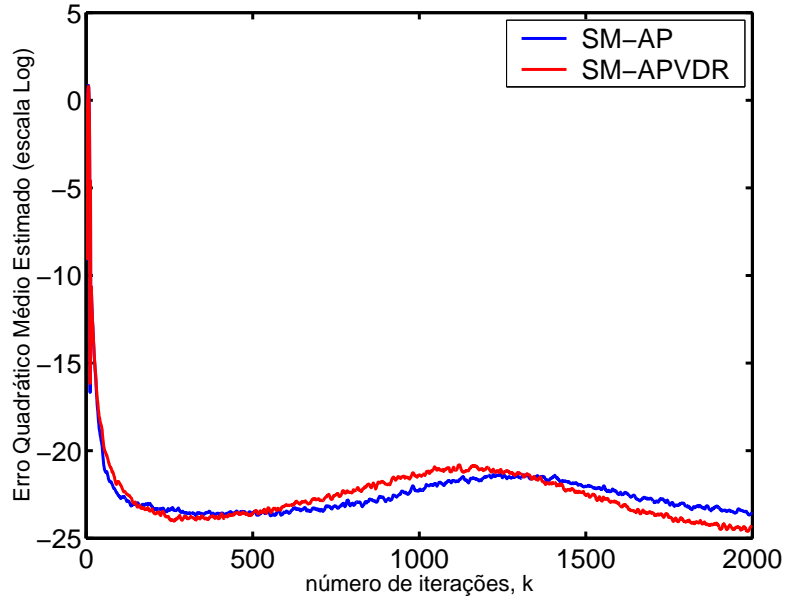


Figura 5.1: Curvas de aprendizagem dos algoritmos SM-AP convencional e SM-APVDR para $SNR = 30$ dB, $\gamma = \sqrt{5\sigma_n^2}$ e $P = 5$.

Em um total de 500 experimentos de 2000 iterações, o número de atualizações por experimento para o SM-APVDR foi de 812 em 2000, enquanto que para o SM-AP convencional esta média foi de 823 em 2000. Assim, em termos de necessidade de atualizações dos coeficientes os algoritmos são equivalentes.

A Tabela 5.1 apresenta a média de iterações que foram realizadas pelo algoritmo empregando o número variável de projeções P_{vr} . É possível observar que somente 15 iterações usaram o número máximo de projeções 5. Tendo em

Tabela 5.1: Atualizações realizadas pelo SM-APVDR utilizando P_{vr} projeções variáveis para $SNR = 30$ dB, $\gamma = \sqrt{5\sigma_n^2}$ e $P = 5$.

P_{vr}	Atualizações
1	311
2	201
3	191
4	94
5	15
Total	812

vista que o SM-AP convencional efetuou 823 iterações utilizando 5 projeções nas suas atualizações, conclui-se, com segurança, que o SM-APVDR apresentou uma complexidade computacional inferior ao SM-AP convencional, obtendo os mesmos resultados.

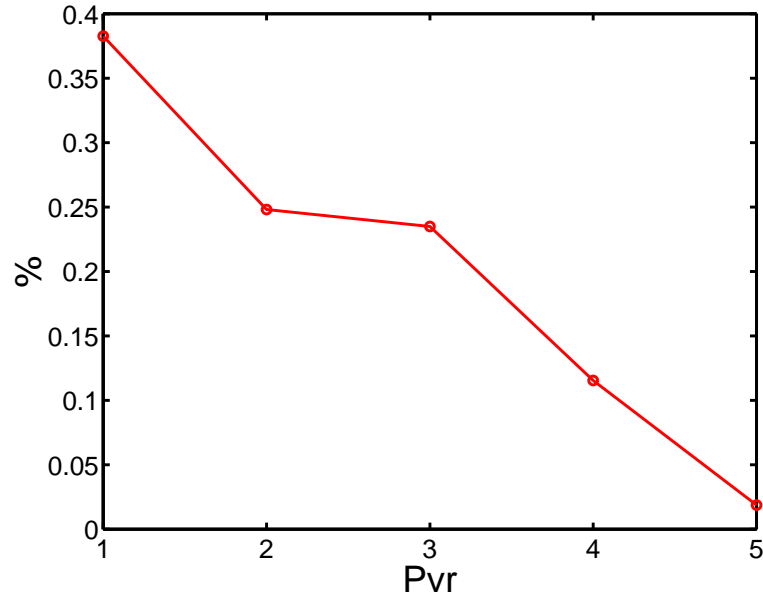


Figura 5.2: Percentagem média de utilização das projeções variáveis P_{vr} , quando foi necessária a atualização, para $SNR = 30$ dB, $\gamma = \sqrt{5\sigma_n^2}$ e $P = 5$.

Na Figura 5.2 encontra-se representada a percentagem média de utilização, por experimento, das projeções de reuso nas vezes em que foi necessário realizar uma atualização dos coeficientes do filtro. Percebe-se que na maioria dos casos, o algoritmo comportou-se como se fosse um SM-NLMS.

A Figura 5.3 apresenta as curvas de aprendizagem resultantes para o caso em que a $SNR = 40$ dB e $\gamma = \sqrt{7\sigma_n^2}$. Os demais parâmetros usados nas simulações foram os mesmos do exemplo anterior. Como se pode observar na figura, as velocidades de convergência e os desajustes obtidos são equivalentes.

Em 2000 iterações, a média de atualizações por experimento do SM-APVDR e do SM-AP convencional foram de 1163 e 1201 respectivamente.

A Tabela 5.2 mostra o uso médio de cada projeção, nas vezes em que houve atualização dos coeficientes do filtro adaptativo. Estes valores encontram-se representados em forma de percentagem na Figura 5.4.

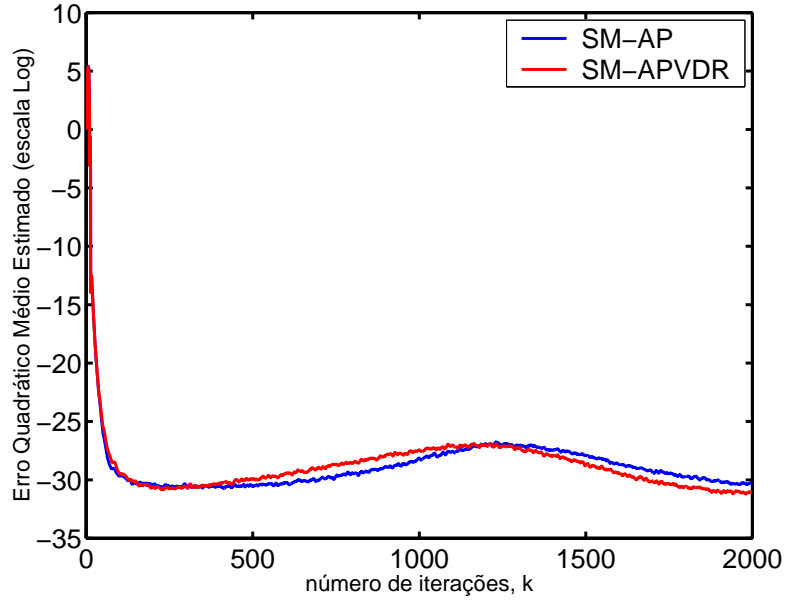


Figura 5.3: Curvas de aprendizagem dos algoritmos SM-AP convencional e SM-APVDR para $SNR = 40$ dB, $\gamma = \sqrt{7\sigma_n^2}$ e $P = 5$.

As curvas de aprendizagem obtidas quando a $SNR = 50$ dB e $\gamma = \sqrt{9\sigma_n^2}$ encontram-se representadas na Figura 5.5. Mantiveram-se os demais parâmetros das simulações anteriores.

Uma vez mais, são extremamente semelhantes os desempenhos dos dois algoritmos quanto à velocidade de convergência e o desajuste.

Tabela 5.2: Atualizações realizadas pelo SM-APVDR utilizando P_{vr} projeções variáveis para $SNR = 40$ dB, $\gamma = \sqrt{7\sigma_n^2}$ e $P = 5$.

P_{vr}	Atualizações
1	303
2	240
3	307
4	258
5	55
Total	1163

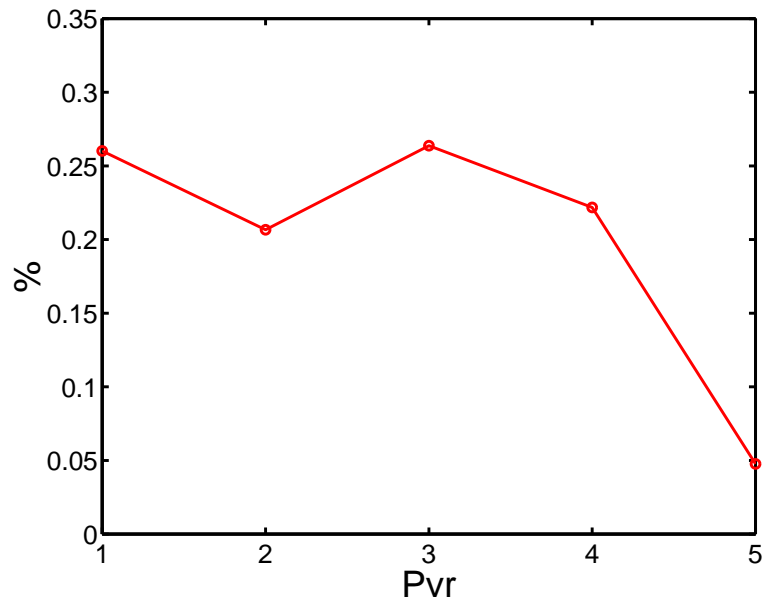


Figura 5.4: Percentagem média de utilização das projeções variáveis P_{vr} , quando foi necessária a atualização, para $SNR = 40$ dB, $\gamma = \sqrt{7\sigma_n^2}$ e $P = 5$.

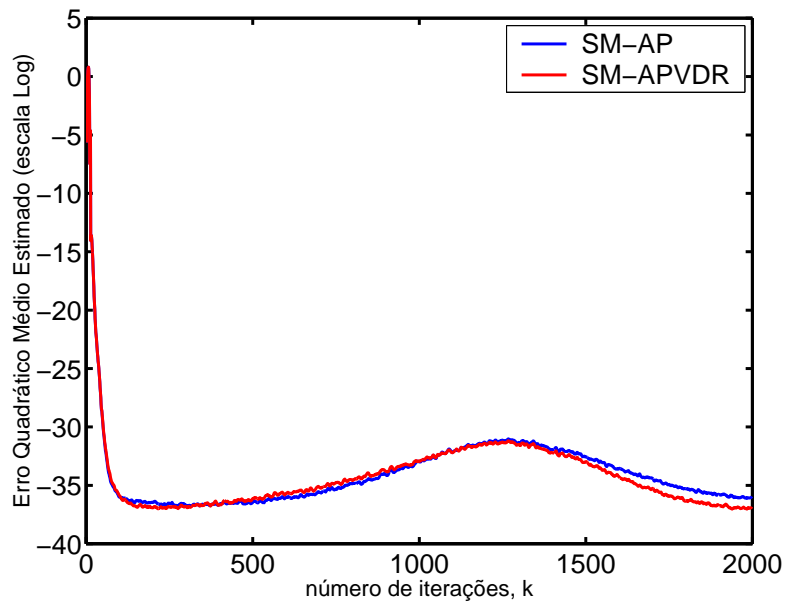


Figura 5.5: Curvas de aprendizagem dos algoritmos SM-AP convencional e SM-APVDR para $SNR = 50$ dB, $\gamma = \sqrt{9\sigma_n^2}$ e $P = 5$.

Tabela 5.3: Atualizações realizadas pelo SM-APVDR utilizando P_{vr} projeções variáveis para $SNR = 50$ dB, $\gamma = \sqrt{9\sigma_n^2}$ e $P = 5$.

P_{vr}	Atualizações
1	204
2	193
3	327
4	519
5	320
Total	1563

É também semelhante a quantidade de atualizações médias requeridas por experimento, uma vez que foram necessárias 1563 e 1626 atualizações em 2000 iterações, para os algoritmos SM-APVDR e SM-AP convencional, respectivamente.

Na Tabela 5.3 encontra-se a utilização média das projeções quando ocorreu atualização do algoritmo SM-APVDR e a sua representação em forma percentual está traçada no gráfico da Figura 5.6.

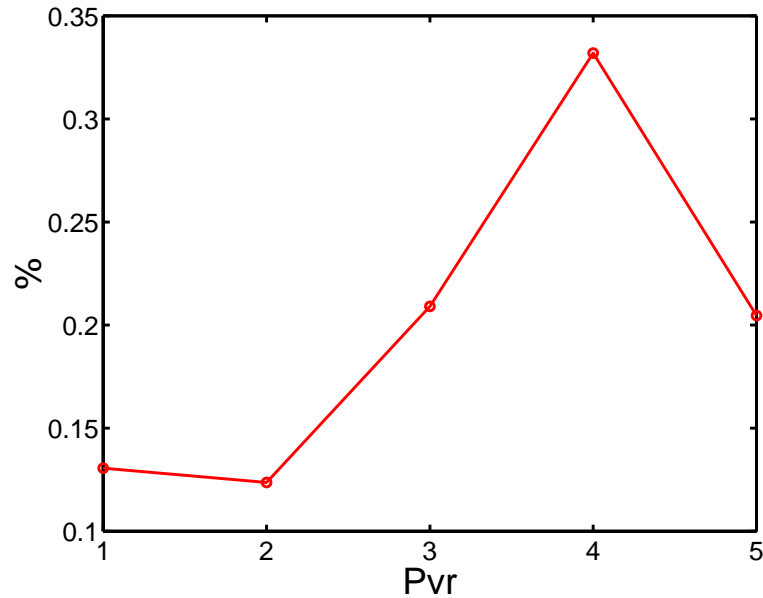


Figura 5.6: Percentagem média de utilização das projeções variáveis P_{vr} , quando foi necessária a atualização, para $SNR = 50$ dB, $\gamma = \sqrt{9\sigma_n^2}$ e $P = 5$.

Observa-se a tendência do algoritmo otimizado comportar-se mais próximo do SM-AP convencional.

Em todos os casos testados comprova-se que resultados análogos são alcançados com uma redução na complexidade computacional pelo emprego do SM-APVDR em comparação com a versão do SM-AP convencional.

5.3 SM-APVDR *versus* Algoritmos Clássicos

Nesta seção, o algoritmo SM-APVDR será confrontado com os algoritmos clássicos apresentados no Capítulo 3, na tarefa de identificar o canal de rádio móvel experimental, submetidos a três situações distintas para a relação sinal-ruído $SNR = 10$ dB, 30 dB e 40 dB.

Os parâmetros usados pelos algoritmos serão os já mencionados quando se efetuou a simulação — para a aplicação — logo em seguida a sua apresentação.

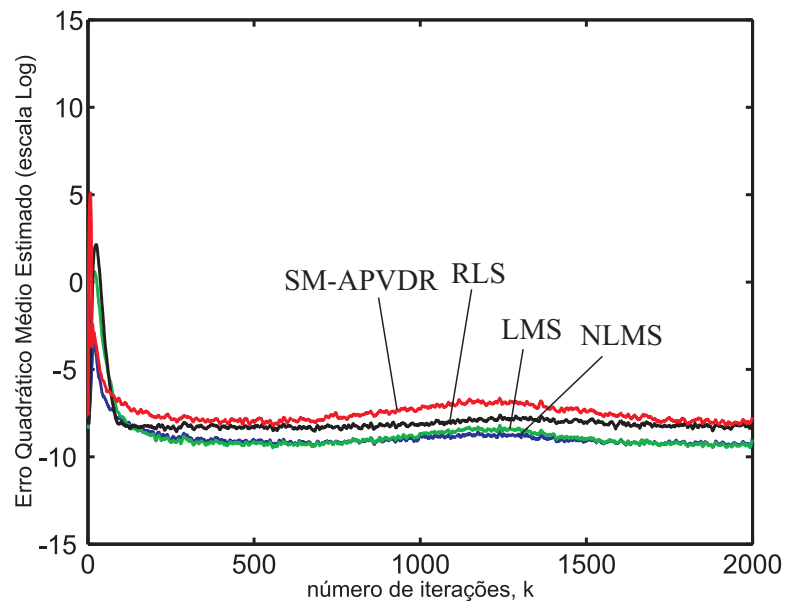


Figura 5.7: Curvas de aprendizagem dos algoritmos NLMS, LMS, RLS e SM-APVDR para $SNR = 10$ dB, $\gamma = \sqrt{3\sigma_n^2}$ e $P = 5$.

A Figura 5.7 contém as curvas médias de aprendizagem obtidas por meio de 500 simulações de 2000 iterações cada, na situação em que $SNR = 10$ dB.

Nota-se que nesta situação o algoritmo SM-APVDR apresentou a maior velocidade de convergência. Percebe-se também seu ótimo desempenho na habilidade de *tracking*.

A Figura 5.8 apresenta as curvas médias de aprendizagem para o caso em que $SNR = 30$ dB.

Observa-se que nesta situação o algoritmo SM-APVDR novamente apresentou a maior velocidade de convergência e um dos melhores desajustes.

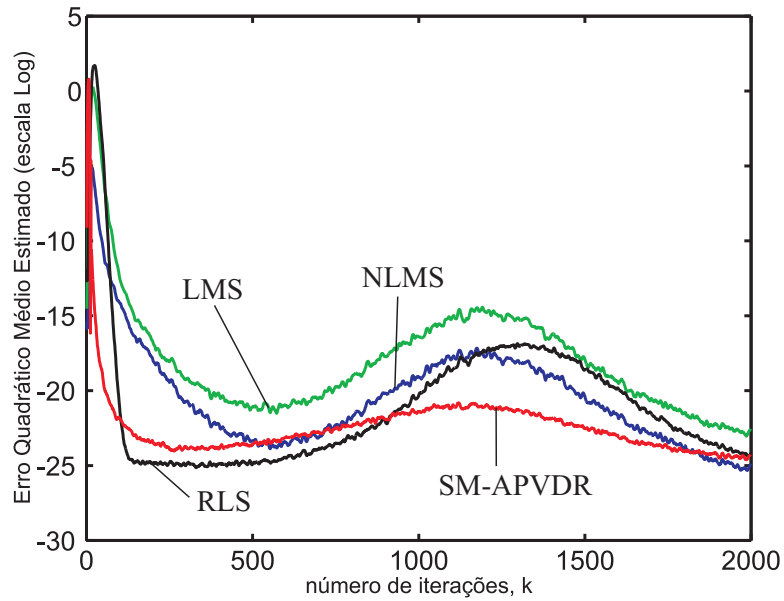


Figura 5.8: Curvas de aprendizagem dos algoritmos NLMS, LMS, RLS e SM-APVDR para $SNR = 30$ dB, $\gamma = \sqrt{5\sigma_n^2}$ e $P = 5$.

Na Figura 5.9 estão traçadas as curvas médias de aprendizagem para as condições similares ao exemplo anterior, porém para uma $SNR = 40$ dB. Neste caso, observa-se o melhor desempenho do algoritmo tanto nos aspectos de velocidade de convergência como no desajuste obtido.

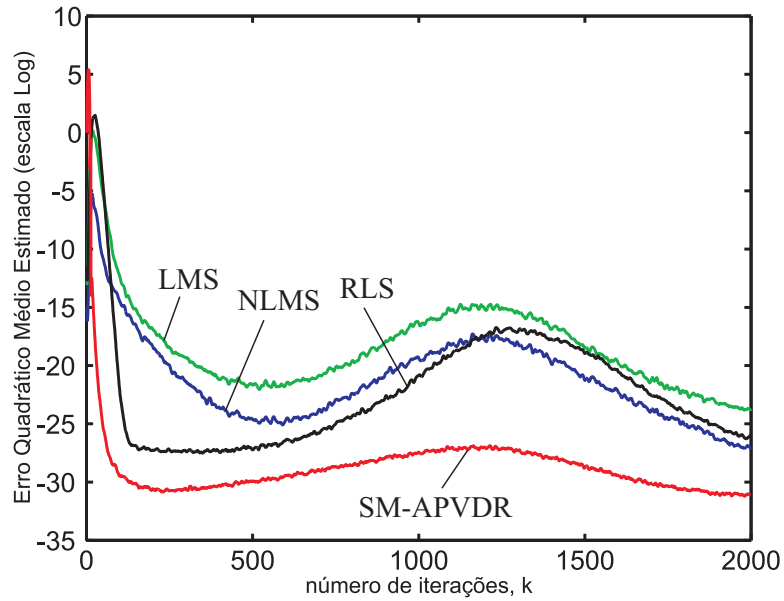


Figura 5.9: Curvas de aprendizagem dos algoritmos NLMS, LMS, RLS e SM-APVDR para $SNR = 40$ dB, $\gamma = \sqrt{7\sigma_n^2}$ e $P = 5$.

5.4 Resumo

O capítulo apresentou os resultados de uma série de simulações envolvendo o SM-APVDR e outros algoritmos da filtragem adaptativa. Inicialmente foi comprovado, através de simulações, que este algoritmo proporciona uma complexidade computacional inferior que o SM-AP convencional. Em seguida o desempenho do SM-APVDR na situação de relação sinal-ruído baixa e média foi comparada aos algoritmos clássicos, onde se mostrou superior em vários aspectos.

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

Neste capítulo serão apresentadas as considerações finais desta tese e serão fornecidas algumas sugestões para trabalhos que poderão ser desenvolvidos a partir dos estudos aqui realizados.

O canal de um sistema móvel celular constitui-se em um exigente ambiente da comunicação digital, tendo em consideração suas características particulares decorrentes dos efeitos de propagação em multipercurso e da dispersão em frequência provocada pelo efeito Doppler que o tornam um sistema linear variante no tempo.

O fato de o canal de rádio móvel ser um ambiente não-estacionário e com estatísticas desconhecidas, *a priori*, conduz-nos naturalmente ao emprego de filtros adaptativos nos receptores.

A escolha do algoritmo de adaptação, para esta aplicação, deve ser extremamente criteriosa quanto ao requisito de alta velocidade de convergência — pois o tamanho da seqüência de treinamento normalmente limita-se a uns poucos *bits* — e quanto à complexidade computacional — especialmente tratando-se da estação móvel, uma vez que o consumo de energia está intimamente relacionado com esta medida de desempenho. Além do mais, devido às altas taxas de transmissão, o tempo para que o DSP (*Digital Signal Processor*) escolhido execute os cálculos necessários é limitado.

Evidentemente o ajuste final não deve ser comprometido pela escolha do algoritmo de adaptação escolhido.

Este trabalho apresentou uma otimização do promissor algoritmo recentemente proposto, *SM-Affine Projection*, para a aplicação de identificar um canal de rádio móvel especificado pelo modelo de propagação COST259, destinado a avaliar sistemas como o GSM.

A modificação introduzida conduziu ao algoritmo SM-APVDR que mantém a boa *performance* do SM-AP convencional no que se refere à rapidez com que converge e ao desajuste, ao mesmo tempo que proporciona uma redução na complexidade computacional. É neste requisito que ocorre a otimização do algoritmo.

6.2 Trabalhos Futuros

A alteração do SM-AP original, para a aplicação de interesse neste trabalho, teve como consequência o surgimento do SM-APVDR que, de fato, constitui-se em uma ampla oportunidade de novos estudos. O trabalho limitou-se ao problema de identificação de sistema, deixando uma lacuna para as demais aplicações onde a filtragem adaptativa vem sendo utilizada com grande sucesso atualmente.

Estudos, também, podem ser realizados aliando o emprego do algoritmo proposto à técnica de filtros adaptativos esparsos na identificação de canais móveis com resposta ao impulso longa [23].

No presente trabalho, o limiar determinístico para o erro de estimação (γ) foi ajustado manualmente em função da potência do ruído gaussiano aditivo σ_n^2 . Entretanto, buscou-se, sem sucesso, estimar este parâmetro de forma automática empregando o seguinte método:

$$\gamma = K_g |c_{yd}(k) - p_d(k)|$$

onde:

K_g é o fator de ajuste para a estimativa de γ

$c_{yd}(k) = \lambda_g c_{yd}(k-1) + (1 - \lambda_g)y(k)d(k)$ é a correlação cruzada da saída do filtro adaptativo e do sinal desejado, no instante k

$p_d(k) = \lambda_g p_d(k-1) + (1 - \lambda_g)d^2(k)$ é a estimativa da potência do sinal desejado, no instante k

λ_g é o fator de esquecimento para a estimativa de γ e $0.01 < \lambda_g < 0.9$

Referências Bibliográficas

- [1] ABRANTES, S. A., *Processo Adaptativo de Sinais*. Lisboa - Portugal, Fundação Calouste Gulbenkian, 2000.
- [2] DINIZ, P. S. R., SILVA, E. A. B., NETTO, S. L., *Digital Signal Processing - System Analysis and Desing*. United Kingdom, Cambridge University Press, Dezembro 2002.
- [3] WERNER, S., *Reduced Complexity Adaptive Filtering with Applications to Communications Systems*. Ph.D. dissertation, Helsinki University of Technology, Finland, November 2002.
- [4] DINIZ, P. S. R., *Adaptive Filtering - Algorithms and Practical Implementation*. United States of America, Kluwer Academic Publishers, 2002.
- [5] PATZOLD, M., *Mobile Fading Channel*. England, John Wiley & Sons, 2002.
- [6] OPPENHEIM, A. V., *Digital Signal Processing*. Englewood Cliffs, New Fersey, Prentice-Hall, 1975.
- [7] RAPPAPORT, T. S., SEIDEL, S. Y., TAKAMIZAWA, K., “Statistical Channel Impulse Response Models for Factory and Open Plan Building Radio Communications System Design”, *IEEE Transactions on Communications*, v. 39, n. 5, pp. 794–806, May 1991.
- [8] TURIN, G. L., “A Statistical Model of Multipath Propagation”, *IEEE Transactions on Vehicular Technology*, v. 21, pp. 1–9, February 1972.
- [9] HASHEMI, H., “The Indoor Radio Propagation Channel.”, *Proceedings of the IEEE*, v. 81, n. 7, pp. 943–968, July 1963.

- [10] TRANTER, W. H., SHANMUGAN, K. S., RAPPAPORT, T. S., *et al.*, *Principles of Communication Systems Simulation with Wireless Applications*. New Jersey, Prentice Hall, 2004.
- [11] HOFSTETTER, H., MOLISH, A. F., STEINBAUER, M., “Implementation of a COST259 Geometry Based Stochastic Channel Model for Macro and Micro Cells”. In: *4th European Personal Mobile Communications Conference (EPMCC 2001)*, Vienna, February 2001.
- [12] STEINBAUER, M., MOLISCH, A. F., MOGENSEN, P., *et al.*, “Directional Channel Modelling in COST259 SWG 2.1”. In: *Millenium Conference on Antennas & Propagation (AP2000)*, Davos, April 2000.
- [13] STEINBAUER, M., ASPLUND, H., COSTER, I. D., *et al.*, “Modelling Unification Workshop”. N. COST259TD(99)061, Vienna, April 1999.
- [14] STEINBAUER, M., ASPLUND, H., COSTER, I. D., *et al.*, *COST259 SWG 2.1 Mission Report: Modelling Unification Workshop*. Vienna, April 1999.
- [15] FAILLI, M., *Digital Land Mobile Communication. COST 207 Final Report*. Luxembourg, 1989.
- [16] PROAKIS, J., *Digital Communications*. New York, McGraw-Hill, 1989.
- [17] HOEHER, P., “A Statistical Discrete-Time Model for the WSSUS Multipath Channel”, *IEEE Transactions on Vehicular Technology*, v. 41, n. 4, pp. 461–468, November 1992.
- [18] SCHULZE, H., “Stochastic Models and Digital Simulation of Mobile Channels”, *U.R.S.I./ITG Conf. in Kleinheubach by German PTT, Darmstadt*, v. IT-18, pp. 363–378, May 1989.
- [19] HAYKIN, S., *Adaptive Filter Theory*. Prentice-Hall, 2th edition, 1991.
- [20] TARRAB, M., FEUER, A., “Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data.”, *IEEE Transactions on Information Theory*, v. 34, n. 4, pp. 680–691, 1988.

- [21] NAGUMO, J. I., NODA, A., “A learning method for system identification.”, *IEEE Transactions on Automatic Control*, pp. 282–287, July 1967.
- [22] GOLLAMUDI, S., NAGARAJ, S., KAPOOR, S., *et al.*, “Set-membership adaptive equalization and updator-shared implementation for multiple channel communications systems”, *IEEE Transactions on Signal Processing*, v. 46, n. 9, pp. 2372–2384, September 1998.
- [23] WERNER, S., APOLINÁRIO Jr, J. A., DINIZ, P. S. R., *et al.*, “A Set-membership Approach to Normalized Proportionate Adaptation Algorithms”. In: *EUSIPCO2005 - European Signal Processing Conference*, Antalya - Turkey, September 2005.

Apêndice A

Código Matlab para o algoritmo SM-APVDR

Este apêndice exhibe o código para o algoritmo SM-APVDR. Este código foi empregado no trabalho para a obtenção dos resultados apresentados.

```
function [W, e, y, iter, usoPvr, Proj] = SMAPVDRgama(W, X, d, n, gama, iter, usoPvr, niveis, Proj, k)
%function [W, e, y, iter, usoPvr, Proj] = SMAPVDRgama(W, X, d, n, gama, iter, usoPvr,niveis, Proj, k)
%
% SET-MEMBERSHIP AFFINE PROJECTION COM FATOR DE REUSO VARIÁVEL
% SET-MEMBERSHIP AFFINE PROJECTION WITH VARIABLE DATA-REUSE FACTOR (SM-APVDR)
%
% Entradas:W          - Nx1 vetor dos coeficientes a ser atualizado
%          X          - NxP matriz de entrada
%          d          - Px1 vetor do sinal desejado
%          n          - Px1 vetor do ruído de medição
%          gama       - 1x1 limiar para o erro
%          iter       - 1x1 número de instantes que foram necessárias as atualizações
%          usoPvr     - Px1 vetor com o número de atualizações para cada projeções
%                   Pvr utilizadas
%          Proj       - 1x1 Numero de projeções utilizadas anteriormente
%          k          - 1x1 índice da amostra de entrada
%
% Saídas: W          - Nx1 vetor dos coeficientes a atualizado
%          e          - 1x1 erro estimação a priori
%          y          - 1x1 saída do filtro
%          iter       - 1x1 número de instantes que foram necessárias as atualizações
%          usoPvr     - Px1 vetor com o número de atualizações para cada projeções
%                   Pvr utilizadas. O número de projeções varia de acordo com
%                   a quantização de alfa.
%
%          Proj       - 1x1 Numero de projeções utilizadas nesta iteração
```



```

delta = 1e-8;
y      = ctranspose(W)*X(:,1);
e      = d(1)+n(1)-y;

if abs(e) > gama
    % É necessário atualizar os coeficientes do filtro

    iter = iter+1;
    alfa = 1-gama/abs(e);
    P     = size(X,2);

    if P == 1
        W = W+alfa*ctranspose(e)*X/(ctranspose(X)*X+delta);
        usoPvr(1) = usoPvr(1)+1;
    else
        % Escolha de Pvr em função de alfa e da tabela de níveis

        m = 1;
        while alfa>niveis(m,1),
            m = m+1;
        end
        Pvr = niveis(m,2);
        Pvr = min(Pvr,k); % Para as situações iniciais, quando k < Pvr

        E = zeros(Pvr,1);
        g = eye(Pvr,1);
        if Pvr>Proj
            for m=2:Pvr
                epsilon(m) = d(m)+n(m)-ctranspose(W)*X(:,m);
                if abs(epsilon(m))>gama
                    g(m,1) = ctranspose(epsilon(m))/abs(epsilon(m))*gama;
                else
                    g(m,1) = ctranspose(epsilon(m));
                end
                E(m,1) = ctranspose(epsilon(m));
            end
        end

        u1 = E-g;
        u1(1)=ctranspose(e)*alfa;
        X   = X(:,1:Pvr);
        t   = inv(ctranspose(X)*X+delta*eye(Pvr))*u1; %*u1*alfa*ctranspose(e);
        W   = W+X*t;
        usoPvr(Pvr) = usoPvr(Pvr)+1;
        Proj = Pvr;
    end
end
end

```

O fragmento de código a seguir apresenta como se obtém a matriz **niveis**, destinada à quantização de alfa:

```
niveis = 0.2-4/P*log(1-(0:P-2)/P*(1-exp(-P/4)));  
niveis(P) = 1;  
niveis = [niveis; 1:P]';
```