

CANCELAMENTO DE ECO EM LINHAS BASEADAS NO PADRÃO G.168
UTILIZANDO O ALGORITMO SM-AFFINE PROJECTION

Rozalvo Pereira Braga

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

Aprovada por:

Prof. Paulo Sérgio Ramirez Diniz, Ph.D.

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

Prof. José Antonio Apolinário Jr., D.Sc.

RIO DE JANEIRO, RJ - BRASIL

ABRIL DE 2005

BRAGA, ROZALVO PEREIRA

Cancelamento de Eco em Linhas
baseadas no Padrão G.168 utilizando
o Algoritmo SM-Affine Projection [Rio de
Janeiro] 2005

xiii,107 pp 29,7 cm (COPPE/UFRJ,
M.Sc., Engenharia Elétrica, 2005)

Tese - Universidade Federal do Rio de
Janeiro, COPPE

1.Filtragem Adaptativa 2.Recomendação
G.168 do ITU-T 3.Modelo de Caminho do
Eco 4.Cancelamento de Eco 5.Algoritmo
SM-Affine Projection

I.COPPE/UFRJ II.Título (série)

Agradecimentos

Em primeiro lugar gostaria de agradecer à minha esposa e minha filha pela compreensão que tiveram comigo durante o tempo que me dediquei à elaboração desta tese, o apoio recebido delas foi uma grande motivação para que eu continuasse persistindo no meu objetivo. Quero também agradecer especialmente aos professores Paulo Sérgio Ramirez Diniz e Stefan Werner que me orientaram na elaboração desta tese, dando o suporte necessário e incentivando-me sempre a continuar em frente.

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

CANCELAMENTO DE ECO EM LINHAS BASEADAS NO PADRÃO G.168
UTILIZANDO O ALGORITMO SM-AFFINE PROJECTION

Rozalvo Pereira Braga

Abril/2005

Orientador: Paulo Sérgio Ramirez Diniz

Programa: Engenharia Elétrica

A aplicação dos filtros digitais em vários campos da Engenharia é atualmente uma das áreas de pesquisa que ainda é bastante ativa. Alguns exemplos de utilização dos filtros digitais são em telecomunicações, informática, equipamentos biomédicos, instrumentos musicais e eletrodomésticos, entre outros.

O objetivo deste trabalho é estudar a aplicação de um tipo especial de filtro digital, os filtros adaptativos, na área de telecomunicações, com o objetivo de realizar o cancelamento de eco em linhas digitais, utilizando-se como parâmetros para análise e comparação, os ambientes da Recomendação G.168 do ITU-T. O eco, nas linhas digitais de telecomunicações é o resultado, principalmente da utilização de um dispositivo de conversão de linha, chamado híbrida, que faz com que retorne para quem originou o sinal, parte da energia transmitida.

Serão analisados a natureza do eco nas redes digitais, os tipos de filtros digitais utilizados para executar a aplicação em estudo e os algoritmos LMS, RLS, NLMS, Affine Projection, SM-BNDRLMS e SM-Affine Projection, que serão utilizados para fazer a atualização dos coeficientes dos filtros adaptativos. Para cada um destes itens a serem analisados, será feita uma breve abordagem teórica dos mesmos, procurando-se focar as partes que mais tenham relação com a aplicação em estudo.

Ao final do trabalho serão analisados o desempenho de cada um dos algoritmos de atualização dos coeficientes, para a aplicação de cancelamento de eco, enfocando-se especialmente o algoritmo Set-Membership Affine Projection (SM-AP), que será otimizado, buscando-se o melhor desempenho deste algoritmo em relação aos demais tipos de algoritmos disponíveis para a atualização dos coeficientes dos filtros adaptativos, utilizando-se como parâmetros experimentais aqueles definidos na Recomendação G.168 do ITU-T.

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

ECHO CANCELLATION IN TELEPHONE LINES USING THE
RECOMENDATION G.168 USING SM-AFFINE PROJECTION ALGORITHMS

Rozalvo Pereira Braga

April/2005

Advisor: Paulo Sérgio Ramirez Diniz

Department: Electrical Engineering

The application of the digital filters in different areas of Engineering is nowadays a research area that has witnessed continuous growth. Some examples where digital filters can be found are telecommunications, computer science, medical equipments, musical instruments and household appliances, just mention a few.

The objective of this thesis is to study one special type of digital filter, the adaptive filters, in the field of telecommunication, in order to perform echo cancellation in digital subscriber lines, using the ITU-T G.168 Recommendation setup parameters. The echo on the telecommunications digital lines, originates from the use of a line conversion device, called hybrid, which allows the energy from transmission to return to the transmitter signal.

In this thesis, it is analyzed how echos are generated in digital network and the types of adaptive filtering algorithms available for echo cancellation. The algorithms considered are the LMS, the RLS, the NLMS, the Affine Projection, the SM-BNDRLMS and the SM-Affine Projection. After a brief introduction to each algorithm, their application to echo cancellation is discussed.

Then the performance of each algorithm in echo cancellation is studied, specially focusing the Set-Membership Affine Projection (SM-AP) algorithm, that will be optimized for the application. The best performance is searched for the setup defined in ITU-T G.168 Recommendation.

Sumário

1	Filtragem Adaptativa - Características e Aplicações	1
1.1	Introdução à Filtragem Adaptativa	1
1.2	Processamento Adaptativo de Sinal	2
1.3	Algoritmos Adaptativos	3
1.4	Aplicações da Filtragem Adaptativa	6
1.5	Comentários	8
2	Descrição do Padrão G.168 e do Ambiente Experimental	9
2.1	Geração do Eco em Redes Telefônicas	9
2.2	A Percepção de Atrasos e Ecos	11
2.3	Métodos de Cancelamento de Eco em Linhas Telefônicas	12
2.3.1	Inserção de Perdas na Rede	13
2.3.2	Uso de Supressores de Eco	14
2.3.3	Uso de Canceladores de Eco	15
2.4	O Padrão de Cancelamento de Eco G.168	16
2.4.1	O Eco nas Redes Digitais de Telecomunicações	16
2.4.2	Características dos Canceladores de Eco para o Padrão G.168	17
2.4.3	Ambiente Experimental do Padrão G.168	18
2.5	Comentários	21
3	Desempenho dos Algoritmos Adaptativos em Canceladores de Eco	22
3.1	Os Filtros FIR e suas características	22

3.1.1	O Filtro FIR no Cancelamento de Eco	23
3.2	Aplicação do Algoritmo LMS em Canceladores de Eco	24
3.2.1	Desempenho do Cancelador de Eco LMS com os Setups do Padrão G.168	26
3.3	Aplicação do Algoritmo NLMS em Canceladores de Eco	28
3.3.1	Desempenho do Cancelador de Eco NLMS com os Setups do Padrão G.168	29
3.4	Aplicação do Algoritmo RLS em Canceladores de Eco	31
3.4.1	Desempenho do Cancelador de Eco RLS com os Setups do Padrão G.168	33
3.5	Aplicação do Algoritmo AP(Affine Projection) em Canceladores de Eco . .	35
3.5.1	Desempenho do Cancelador de Eco Affine Projektion com os Setups do Padrão G.168	37
3.6	Conceito e Aplicação da Filtragem Set-Membership	39
3.6.1	A Filtragem Set-Membership (SMF)	39
3.6.2	Aplicação do Algoritmo Set-Membership Binormalized Data Reuse LMS(SM-BNDRLMS) em Canceladores de Eco	41
3.6.3	Desempenho do Cancelador de Eco SM-BNDRLMS com os Setups do Padrão G.168	45
3.7	Comentários	46
4	Avaliação e Otimização do Algoritmo SM-Affine Pro-	
	jection em Canceladores de Eco	48
4.1	O Algoritmo Set-Membership Affine Projection (SM-AP)	49
4.1.1	O Vetor de Parâmetro $\mathbf{g}(k)$	51
4.2	Desempenho do Cancelador de Eco Set-Membership Affine Projection no Ambiente do Padrão G.168	52
4.3	Otimização do Cancelador de Eco SM-AP para o Padrão G.168	55
4.4	Desempenho do Cancelador de Eco Set-Membership Affine Projektion Otimizado	59
4.5	Desempenho do Cancelador de Eco SM-Affine Projektion Otimizado em Ambiente Double Talk	60

4.6	Comentários	62
5	Comparações do Desempenho dos Canceladores de Eco para o Padrão G.168	64
5.1	Os algoritmos LMS, NLMS, RLS, AP, SM-BNDRLMS, SM-AP	66
5.2	O algoritmo SM-AP Otimizado versus o RLS e SM-AP	66
5.3	Comparação do Cancelador SM-Affine Projektion e SM-Affine Projection Otimizado em Ambiente Double Talk	68
5.4	Comentários	72
6	Conclusões	73
	Referências Bibliográficas	75
A	Modelos dos Canais Digitais - Recomendação G.168	77
B	Gráficos de Desempenho dos Algoritmos	85
C	Programas em MATLAB utilizados nas simulações	134

Capítulo 1

Filtragem Adaptativa - Características e Aplicações

1.1 Introdução à Filtragem Adaptativa

O desenvolvimento das técnicas de processamento digital de sinais permitiu grandes avanços na área de telecomunicações nos últimos trinta anos. Entre as principais alavancas que impulsionaram o desenvolvimento desta técnica, estão o avanço dos circuitos integrados em muito larga escala (VLSI) e a informática [3].

Os circuitos integrados mais utilizados para a execução dos algoritmos de processamento digital de sinais são os DSP (Digital Signal Processors). No atual estágio do desenvolvimento tecnológico, estes componentes tornaram-se confiáveis, precisos, de tamanho reduzido e bastantes flexíveis, possibilitando seu uso em diversos campos da área de telecomunicações e em outras áreas da engenharia.

A técnica de filtragem é uma das aplicações do processamento digital de sinais, que consiste em mapear a sequência na entrada de um filtro para uma outra sequência na sua saída, de modo a se extrair as informações desejadas contidas na sequência de entrada. O filtro é dito invariante no tempo quando seus parâmetros são fixos, e linear quando o sinal de saída é uma função linear do sinal de entrada.

Os filtros lineares invariantes possuem uma função de transferência, que por sua vez está ligada a um algoritmo de implementação. Se as especificações fixas são desconhecidas ou as especificações são variantes, se torna necessária a utilização de filtros chamados de adaptativos. Estes filtros possuem características que variam com a entrada. Entretanto, se em determinado instante do tempo se congelar os parâmetros do filtro, o mesmo pode ser considerado linear neste instante.

Os filtros adaptativos são variantes no tempo, pois seus parâmetros variam

continuamente para se adaptar ao sinal de entrada e atender às especificações. Esta solução é empregada devido ao fato de que os sinais de entrada e os sinais de referência não serem bem definidos no ambiente de utilização do filtro. A característica principal da filtragem adaptativa é a atualização em tempo real dos parâmetros do filtro através de um algoritmo simples utilizando informações do ambiente.

Valendo-se do progresso alcançado na área de circuitos integrados, especialmente microprocessadores e processadores digitais de sinais (DSP), a técnica de filtragem adaptativa é atualmente empregada em diversas aplicações, principalmente na área de telecomunicações. A aplicação a ser analisada neste trabalho é o emprego da filtragem adaptativa no cancelamento de eco em sistemas de telecomunicações, conforme a Recomendação G.168 do ITU-T (International Telecommunications Union). Para isto, serão considerados somente sinais discretos no tempo e cuja geração, ou seja, amostragem e quantização de um sinal contínuo no tempo, estejam de acordo com o Teorema de Nyquist (frequência de amostragem maior que duas vezes a maior frequência do sinal).

1.2 Processamento Adaptativo de Sinal

A filtragem adaptativa é empregada nas situações onde os coeficientes do filtro precisam se adaptar continuamente às condições ambientais. Para atualização dos coeficientes do filtro, o algoritmo necessita de uma informação adicional, que neste caso é o sinal de referência desejado. Este sinal a ser usado, será escolhido de acordo com a aplicação. Abaixo, tem-se um esquema de um filtro adaptativo.

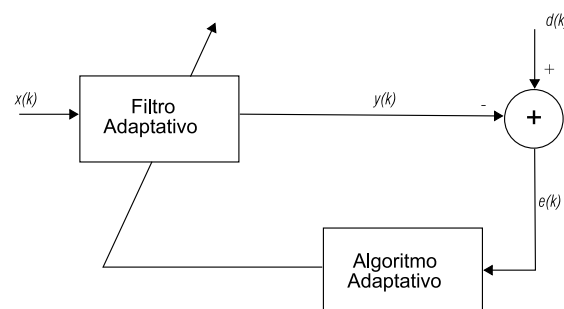


Figura 1.1: Esquema de um filtro adaptativo.

Define-se:

$x(k)$: Sinal de entrada

$y(k)$: Sinal de saída do filtro

$d(k)$: Sinal desejado (referência)

$e(k)$: Sinal erro $[d(k) - y(k)]$

Em geral, o ambiente de uma filtragem adaptativa é composto dos seguintes sinais: o sinal de entrada do filtro, o sinal de saída do filtro adaptativo, o sinal desejado ou de referência e o sinal erro, que é a diferença entre o sinal desejado e o sinal de saída do filtro. Uma função incluindo o sinal erro é usada para fazer a atualização dos coeficientes do filtro. Quanto menor o valor do sinal erro, melhor a saída do filtro acompanhando o sinal desejado. Fazem parte também deste ambiente a estrutura do filtro e o algoritmo usado para atualizar os coeficientes. O sinal de entrada e o sinal desejado são funções da aplicação na qual se deseja usar a filtragem adaptativa. Alguns exemplos de aplicações são: cancelamento de eco, equalização de canais, identificação de sistemas, melhoramento de sinais, cancelamento de ruído e controle. A estrutura do filtro adaptativo é escolhida visando reduzir a complexidade computacional e alcançar o desempenho desejado com o menor número de iterações possível.

Existem duas classes de filtros adaptativos que podem ser diferenciados pela forma da sua resposta ao impulso. A primeira classe são os filtros FIR (Finite-duration Impulse Response), que são implementados com estruturas não recursivas. A segunda classe é a dos filtros IIR (Infinite-duration Impulse Response), que utilizam estruturas recursivas nas suas implementações.

O algoritmo de adaptação é o procedimento que será usado para atualizar os coeficientes do filtro de forma a se minimizar uma função do sinal de erro. Ao se escolher o algoritmo, deve-se considerar alguns aspectos importantes para o processo, sendo um dos principais o tempo de convergência para a solução ótima em função do número de iterações.

1.3 Algoritmos Adaptativos

Conforme anteriormente visto, o objetivo do algoritmo é atualizar os coeficientes do filtro de modo a se atingir o ponto mínimo da função objetivo. Esta função objetivo varia com os seguintes parâmetros: o sinal de entrada, que será chamado $x(k)$, o sinal desejado ou de referência, denotado $d(k)$ e o sinal de saída do filtro,

aqui chamado de $y(k)$. Logo, a função objetivo será: $F = F[x(k), d(k), y(k)]$. Esta função deve obedecer a seguinte propriedade:

- Não-negatividade, isto é: $F[x(k), d(k), y(k)] \geq 0, \forall y(k), x(k)$ e $d(k)$;

O objetivo do algoritmo no processo de filtragem é minimizar a função $F[\cdot]$, de modo que $y(k)$ se aproxime de $d(k)$. Os coeficientes do filtro $\theta(k)$ convergirão para θ_0 , onde θ_0 é o conjunto ótimo de coeficientes que minimizam a função objetivo. Pode-se considerar que o algoritmo compõe-se de três itens básicos: definição do algoritmo de minimização, definição do sinal de erro e definição da função objetivo. A definição do algoritmo de minimização da função F , é o item mais importante do processo de otimização dos coeficientes do filtro adaptativo e está ligado diretamente à velocidade de convergência e à complexidade computacional do processo. Os métodos de otimização mais utilizados são os seguintes:

- Método de Newton: neste método procura-se o mínimo de uma aproximação de segunda ordem da função objetivo usando uma fórmula de atualização iterativa para o vetor parâmetro dado por:

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) - \mu \mathbf{H}_{\theta}^{-1}\{F[e(k)]\} \mathbf{g}_{\theta}\{F[e(k)]\} \quad (1.1)$$

Onde:

$\boldsymbol{\theta}(k)$: É o vetor de parâmetros do filtro

μ : É um fator que controla o tamanho do passo do algoritmo, ou seja, determina a velocidade de mudança dos parâmetros do vetor

$\mathbf{H}_{\theta}\{F[e(k)]\}$: É a matriz Hessiana da função objetivo, ou seja, a matriz das derivadas de segunda ordem de $F[e(k)]$

$\mathbf{g}_{\theta}\{F[e(k)]\}$: É o gradiente da função objetivo com relação aos coeficientes do filtro

- Método Quasi-Newton: Este método é uma simplificação do método de Newton. Como o anterior, tenta minimizar a função objetivo usando uma estimativa calculada recursivamente do inverso da matriz Hessiana, ou seja:

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) - \mu \mathbf{S}(k) \mathbf{g}_{\theta}\{F[e(k)]\} \quad (1.2)$$

Onde $\mathbf{S}(k)$ é uma estimativa de $\mathbf{H}_{\theta}^{-1}\{F[e(k)]\}$, de modo que:

$$\lim_{k \rightarrow \infty} \mathbf{S}(k) = \mathbf{H}_{\theta}^{-1}\{F[e(k)]\} \quad (1.3)$$

O vetor gradiente também é normalmente substituído por uma estimativa eficiente computacionalmente.

- Método gradiente: Neste método o ponto mínimo da função objetivo é procurado, seguindo-se na direção oposta do vetor gradiente da função. A equação de otimização então é:

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) - \mu \mathbf{g}_{\theta}\{F[e(k)]\} \quad (1.4)$$

Ao se implementar os métodos de otimização procura-se um compromisso entre facilidade de implementação e rapidez de convergência. O método gradiente por exemplo, é o mais fácil de implementar, mas o método de Newton normalmente necessita de um número menor de iterações para convergir para o ponto mínimo. O método quasi-Newton é o compromisso médio entre os dois anteriores. Entretanto ele está sujeito à instabilidades devido à forma como são geradas as estimativas da matriz Hessiana inversa. Deve ficar claro entretanto, que o fator mais importante no algoritmo de otimização, independente do método escolhido, é o fator μ . Este fator controla a estabilidade, a rapidez da convergência e também erros residuais do processo adaptativo. Portanto, a escolha deste fator requer um bom conhecimento da aplicação a ser estudada, de modo a se chegar à melhor solução com os menores custos possíveis.

O passo seguinte do processo de filtragem adaptativa é a definição da função objetivo $F[e(k)]$. Esta função deve satisfazer as propriedades de otimização e não negatividade. A escolha afeta os cálculos da matriz Hessiana e a complexidade do vetor gradiente. Os algoritmos da função objetivo apresentados a seguir, estão ordenados conforme suas complexidades computacionais (do mais complexo para o menos complexo, em muitos casos). Estes algoritmos são os mais comumente usados em filtragem adaptativa.

- Erro Médio Quadrático (MSE-Mean Square Error): $F[e(k)] = E[|e(k)|^2]$;
- Mínimo Quadrado (LS-Least Squares): $F[e(k)] = \frac{1}{(k+1)} \sum_{i=0}^k |e(k-i)|^2$;
- Mínimo Quadrado Ponderado (WLS-Weighted Least Squares): $F[e(k)] = \sum_{i=0}^k \lambda^i |e(k-i)|^2$, λ é uma constante menor que 1
- Valor Quadrático Instantâneo (ISV-Instantaneous Squares Value): $F[e(k)] = |e(k)|^2$;
- Set-Membership (SM):

$$J(w) = \begin{cases} 0 & \text{se } |d - \mathbf{w}^T \mathbf{x}|^2 \leq \gamma^2 \forall (x, d) \in S \\ 1 & \text{Para outros valores} \end{cases}$$

O Erro Médio Quadrático (MSE), é uma função objetivo inviável de ser medida, visto que precisa de uma quantidade infinita de informações para ser medido. O MSE é aproximado na prática pelo erro instantâneo. As funções objetivo diferem quanto à complexidade de implementação resultando em diferenças na velocidade de convergência. O Valor Quadrático Instantâneo (ISV) é o algoritmo de mais fácil implementação, mas pode resultar em convergência lenta, visto que se trata de uma função objetivo muito simplificada. O Mínimo Quadrado (LS) é melhor aplicado em ambientes estacionários, enquanto que o Mínimo Quadrado Ponderado (WLS), é usado em aplicações onde o ambiente varia lentamente.

O terceiro e último item importante para a definição do algoritmo de filtragem adaptativa é a definição do sinal erro $e(k)$. A escolha do erro é crucial para todo o algoritmo de filtragem, visto que ele pode afetar sua complexidade computacional, velocidade de convergência e robustez. Para o caso dos filtros IIR (Infinite duration Impulse Response), o mais importante a considerar, quando se faz a escolha do sinal de erro, é a ocorrência de múltiplos mínimos locais para a função objetivo.

Os três itens descritos anteriormente, ou seja, o algoritmo de minimização, a função objetivo e o sinal de erro, compõem a estrutura básica para o estudo da filtragem adaptativa. Neste trabalho, será investigada a adequação de se utilizar o algoritmo *Set-Membership Affine Projection* [13] na aplicação de cancelamento de eco e fazer uma análise da forma mais adequada de se configurar para conseguir o melhor resultado para esta aplicação em particular.

1.4 Aplicações da Filtragem Adaptativa

As aplicações mais conhecidas onde tem sido usada a filtragem adaptativa são: identificação de sistemas, equalização de sinal, melhoramento de sinal, predição e cancelamento de eco. Canceladores adaptativos de ruído para a redução do nível de ruído em cabines de avião, a utilização de filtros adaptativos em sistemas biomédicos, sistemas de telecomunicações e controles industriais, são também aplicações práticas da filtragem adaptativa [8]. Dentre tais aplicações, o cancelamento de eco em linhas de telecomunicações, baseado no padrão G.168 do ITU-T, será o objeto de estudo deste trabalho.

A solução para a aplicação de cancelamento de eco, é o uso de um dispositivo

cancelador de eco ideal que usaria o sinal a ser transmitido à outra extremidade da conexão para gerar uma réplica exata do sinal de eco. Deste modo, o eco pode ser eliminado do sinal recebido da outra extremidade da conexão. Como exemplo tem-se a utilização de modems em linhas telefônicas à 2 fios, onde a solução para o problema do eco é a inserção de um cancelador de eco entre o transmissor e o receptor do modem, fazendo com que em operação "full-duplex", ambos os modems da conexão, possam utilizar toda largura de banda do canal telefônico [12]. Como o ambiente da conexão ou conversação pode apresentar variações de suas características com o tempo, é necessário que o cancelamento de eco seja realizado por um filtro adaptativo.

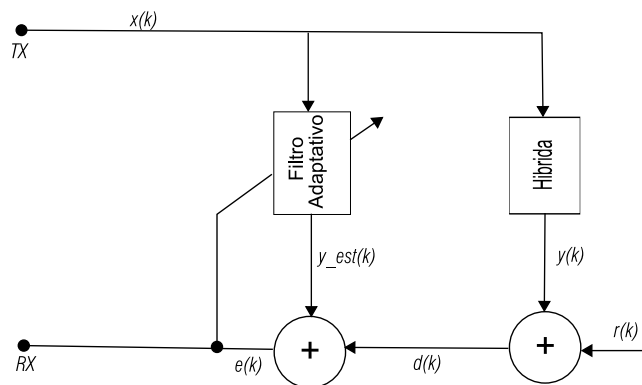


Figura 1.2: Cancelamento de eco com filtro adaptativo.

As características do eco em cada situação podem ser bastantes diversas, o que tem uma influência importante na escolha das técnicas de filtragem adaptativa a serem usadas no cancelamento. Com este trabalho, serão testados todos os algoritmos clássicos, disponíveis para a execução da filtragem (LMS, NLMS, RLS), analisando-se especialmente os resultados da utilização do algoritmo *Set-Membership Affine Projection* para esta aplicação.

Como padrão de referência para o cancelamento de eco, será usada a Recomendação G.168 do ITU-T (International Telecommunications Union). Esta Recomendação descreve as características de um cancelador de eco, incluindo mecanismos de controle como o tom de desabilitação. A G.168 também recomenda os testes de laboratório que devem ser executados em um cancelador de eco para avaliar sua performance nas condições reais de utilização nas redes de telecomunicações.

1.5 Comentários

Neste capítulo fez-se uma abordagem geral em relação ao método de filtragem adaptativa, tratando-se inicialmente da evolução do método, passando-se em seguida por uma breve descrição do processamento adaptativo de sinal, os algoritmos e métodos de otimização utilizados, fechando-se o capítulo com as possíveis aplicações da filtragem adaptativa.

Capítulo 2

Descrição do Padrão G.168 e do Ambiente Experimental

2.1 Geração do Eco em Redes Telefônicas

A principal causa do eco de linha em redes telefônicas é um dispositivo analógico que faz a transformação da linha de 2 para 4 fios e vice-versa, chamado híbrida. A híbrida é um par de transformadores que usa acoplamento indutivo para dividir um sinal duplex em dois sinais simplex. Este equipamento é necessário visto que somente pares simplex de fios, são permitidos para processamento de sinal em redes de longa distância, como equalização e amplificação. As híbridas estão fisicamente localizadas nas centrais de comutação das operadoras de telecomunicações. Devido à perda de corrente na híbrida, uma parte da energia do sinal é refletida de volta à sua fonte original. Esta reflexão em conjunto com os atrasos do caminho, fazem com que os usuários em ambos os lados da conexão ouçam um eco da sua própria voz (ver figura abaixo).

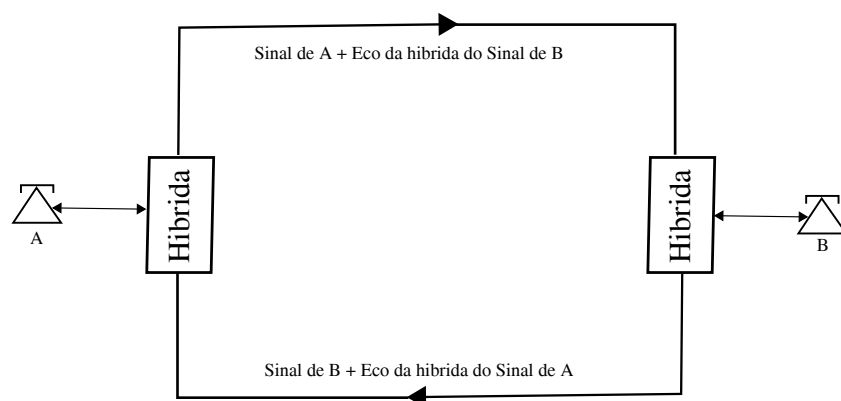


Figura 2.1: Esquema de geração do eco.

O eco, efeito da perda de corrente na híbrida, pode ser dividido em dois tipos:

O eco próximo, que é devido à reflexão do sinal na híbrida local (ver Figura 2.1), sendo o mais significativo devido à sua maior intensidade e o eco distante, que se origina devido à reflexão do sinal na híbrida remota [1]. A distância entre o usuário que está falando e a híbrida do outro lado, assim como também a quantidade de energia refletida pela híbrida, influenciam significativamente na formação do eco. Se o atraso no caminho é pequeno e a intensidade baixa, o usuário normalmente não distingue o eco do ruído presente nas linhas telefônicas. Entretanto, se o atraso excede algumas dezenas de milissegundos, que pode ser o caso em longas distâncias ou comunicações via satélite, o eco é bastante desagradável para o usuário. O outro parâmetro que determina o impacto do eco de linha é a perda na híbrida, ou seja, que quantidade do sinal transmitido é refletido de volta. Normalmente assume-se que a perda é de pelo menos 6 dB [10].

Uma técnica anterior de lidar com o problema de eco de linha envolvia supressão do eco, ou seja, o caminho de retorno do usuário B para o usuário A era fisicamente desconectado quando o dispositivo de supressão do usuário A detectava que o mesmo estava falando e o usuário B estava ouvindo. Estas técnicas não eram eficientes no caso de conversação simultânea (usuários A e B falando ao mesmo tempo) e resultavam em cortes no sinal de áudio.

A técnica para lidar com o problema, descrita neste trabalho, é baseada em um identificador de sistema que, dado um sinal de referência pode retirar o modelo de geração do eco e então subtraí-lo do sinal que é afetado com o eco de linha. Esta solução para o cancelamento de eco, envolve o uso de um dispositivo cancelador de eco ideal que usa o sinal recebido da outra extremidade da conexão (no caso da Figura 2.1), para gerar uma réplica exata do sinal de eco. Deste modo, o eco pode ser eliminado do sinal enviado à outra extremidade da conexão. Como o ambiente da conexão ou conversação pode apresentar variações de suas características com o tempo, é necessário que o cancelamento de eco seja realizado por um filtro adaptativo.

As características do eco em cada situação podem ser bastantes diversas, o que tem uma influência importante na escolha das técnicas de filtragem adaptativa a serem usadas no cancelamento. Com este trabalho, serão testados todos os algoritmos clássicos disponíveis para execução da filtragem, analisando-se especialmente

os resultados da utilização do algoritmo *Set-Membership Affine Projection* para esta aplicação. Como padrão de referência para cancelamento de eco, será usada a Recomendação G.168 do ITU-T. Esta Recomendação descreve as características de um cancelador de eco, incluindo mecanismos de controle do cancelador de eco. Ela também recomenda os testes de laboratório que devem ser executados em um cancelador de eco para avaliar sua performance nas condições reais de utilização nas redes de telecomunicações.

2.2 A Percepção de Atrasos e Ecos

Estudos de percepção feitos nos Laboratórios Bell em 1953 [11] mostraram que atrasos de até 1200 ms em uma chamada telefônica, não causam muita insatisfação entre os usuários com pouca experiência no uso de sistemas de telecomunicações com grandes atrasos. Por outro lado, atrasos de 2400 ms causaram insatisfação entre os usuários testados. Estes resultados têm impacto particular nos sistemas sem fio atuais, que usam topologias de redes baseadas em multi-saltos, visto que eles introduzem atrasos com valores que são relativamente grandes. Embora, relativamente, atrasos grandes da ordem de 1200 ms entre duas pessoas possa ainda ser considerada uma conversação natural, o mesmo não acontece quando ecos são inseridos. Alguma perda é necessária no caminho do eco, visto que de outro modo, o usuário que estiver falando pode ouvir sua própria voz atrasada após algum tempo. Ecos podem até forçar os usuários que estão em conversação a gaguejar, quando houver um atraso de 250 ms ou maior. Ecos com atrasos de 250 ms podem produzir padrões de conversação completamente fora do normal. Considerando que atrasos introduzidos por comunicações via satélite podem ser tão grandes quanto 600 ms, fica claro que, algum tipo de redução do eco é necessária, neste caso.

Os estudos dos Laboratórios Bell [11] mostraram que o tamanho da perda necessária no caminho do eco é uma função do tempo necessário para que o sinal chegue ao lado distante e retorne como eco (caminho de ida e volta). Estes resultados são mostrados na tabela abaixo. Pode-se ver que mesmo ecos com atrasos de 20 ms, necessitam de algum nível de atenuação. A atenuação necessária é cada vez maior, conforme os ecos sejam originados com atrasos maiores.

Atraso Total(ms) - Ida e Volta	Perda Média Necessária (dB)
0	1.4
20	11.1
40	17.7
60	22.7
80	27.2
100	30.9

Tabela 2.1: Perda Média necessária em função do atraso total do eco.

2.3 Métodos de Cancelamento de Eco em Linhas Telefônicas

Na Figura abaixo tem-se o esquema de uma conexão telefônica de longa distância. Pode-se verificar que os telefones em cada um dos lados da conexão, são conectados a uma linha telefônica à dois fios.

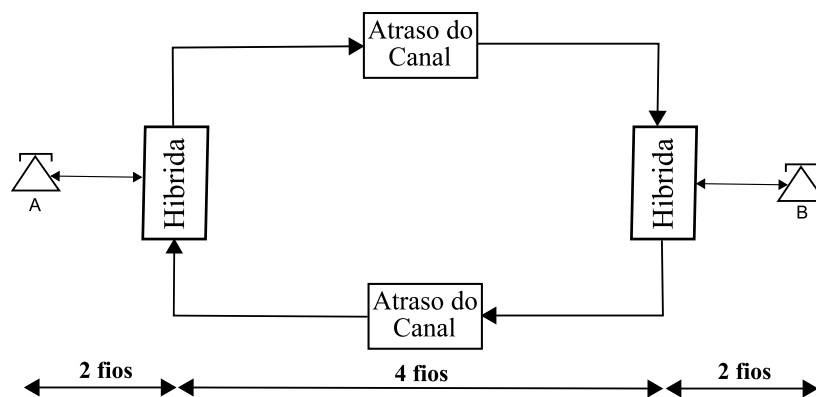


Figura 2.2: Representação de uma conexão telefônica de longa distância.

Um aparelho telefônico basicamente tem um fone de ouvido e um microfone. O microfone é um transdutor que converte sinais acústicos em sinais elétricos que são transmitidos para o loop local do circuito telefônico. O fone de ouvido é um dispositivo que converte sinais elétricos, que chegam do loop local, em sinais sonoros audíveis, normalmente na faixa de 0-4 KHz. O loop local transporta tanto o sinal sendo transmitido quanto o sinal recebido, simultaneamente, em dois fios.

Visto que as informações envolvidas em uma comunicação, são sinais elétricos

transmitidos via linhas telefônicas ou ondas de rádio sendo transmitidas no espaço, atrasos entre os usuários A e B sempre existirão. Na central de comutação, cada um dos circuitos a dois fios, que ficam em ambos os lados da conexão telefônica, são convertidos em dois circuitos a dois fios, usando uma híbrida. Esta conversão é feita para permitir o uso de circuitos unidirecionais que podem ser multiplexados no tempo e na frequência, como por exemplo os amplificadores e os multiplexadores.

O circuito da híbrida representado na Figura 2.3, é um circuito de ponte passiva que usa quatro transformadores. Nesta figura, Z é uma impedância balanceada. Quando Z é igual a impedância da rede à dois fios, o sinal que chega na porta IN é passado exclusivamente para o lado à dois fios da rede, no lado direito $b(n)$. Quando a impedância Z não casa com a impedância da parte à dois fios, parte do sinal que chega à porta IN será transmitido para a porta OUT . Visto que este sinal chegará para o usuário que o gerou através de sua porta IN , com os atrasos introduzidos pelo meio, este usuário perceberá este sinal como um eco da sua própria voz.

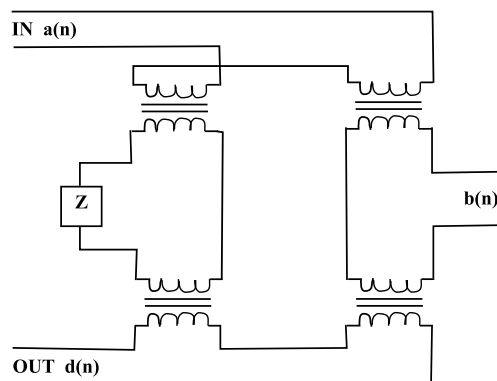


Figura 2.3: Representação simplificada de um circuito híbrida.

A seguir são discutidas algumas possíveis soluções para se construir o cancelamento de eco em linhas telefônicas. Alguns métodos e resultados aplicados ao sistema telefônico podem ser estendidos para outros problemas de cancelamento de eco, como por exemplo, o cancelamento de eco acústico.

2.3.1 Inserção de Perdas na Rede

Uma das primeiras e mais simples soluções para o problema do eco em linhas telefônicas é a inserção de perdas na rede telefônica no caminho do sinal do emissor para o receptor e vice-versa. Com a inserção das perdas, existe uma atenuação do

sinal do usuário que está falando para o receptor de "L" dB. Visto que o eco faz um caminho completo na rede (ida e volta), espera-se que o eco alcance o usuário que gerou o sinal original com uma atenuação de "2L" dB, aumentando a razão sinal/eco em "L" dB.

2.3.2 Uso de Supressores de Eco

Outra solução para o problema do eco é o uso de supressores. Os primeiros supressores de eco foram propostos por volta de 1920 por Clark [11]. Supressores de eco são chaves que usam a informação de voz/silêncio baseada nos detectores de sinal de voz. Eles controlam uma chave de transmissão e outra de recepção, como mostrado na Figura 2.4. Supressores de eco trabalham como descrito a seguir:

- Quando o usuário distante (far-end) está falando e o usuário local (near-end) está em silêncio, a chave de recepção está na posição sem perda e a chave de transmissão está na posição aberta (Disable).
- Quando o usuário distante (far-end) está em silêncio, a chave de transmissão move-se para a posição fechada (Enable).
- Quando o usuário distante (far-end) é interrompido pelo usuário local (near-end), a chave de transmissão é fechada (Enable) e a chave de recepção é comutada para a posição de atenuação.

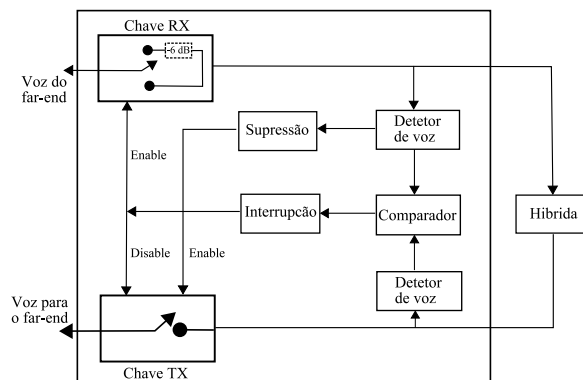


Figura 2.4: Representação esquemática de um supressor de eco.

O nível de ajuste dos detectores de voz, comparadores, ruído ambiental e nível nominal do sinal no circuito mostrado na figura acima, afetam bastante a performance do supressor de eco. Supressores de eco que não trabalham com níveis ótimos podem gerar efeitos indesejáveis, como por exemplo o picotamento do sinal

do usuário local (near-end). Isto acontece quando ecos longos são introduzidos na rede.

2.3.3 Uso de Canceladores de Eco

Os canceladores de eco foram propostos inicialmente por volta de 1970. Os canceladores de eco são uma opção poderosa para substituir os supressores de eco, visto que eles tentam reduzir o eco, estimando a saída do circuito híbrida baseado na sua entrada através do uso de um filtro adaptativo. Baseado na estimativa do eco obtida pelo filtro digital, uma estimativa da saída da híbrida, sem o eco, pode ser obtida.

Um cancelador de eco está representado na Figura 2.5, onde $a(k)$ é o sinal de voz do usuário A, $b(k)$ é o sinal de voz do usuário B e $d(k)$ é o sinal que deveria chegar ao usuário A se não houvesse cancelamento de eco. Neste caso, $d(k)$ é:

$$d(k) = Ga(k) * h(k) + b(k) \quad (2.1)$$

onde $h(k)$ é a resposta ao impulso do circuito híbrida e G é uma constante que modela as perdas introduzidas em um circuito telefônico antes da híbrida. Uma resposta ao impulso $h(k)$ típica da híbrida seria um atraso puro seguido pela resposta ao impulso da híbrida, com um total de 4 ms de comprimento.

Um filtro adaptativo pode ser usado para estimar $Ga(k) * h(k)$ a partir de $a(k)$, ajustando-se os coeficientes de um filtro transversal. Um filtro adaptativo tenta minimizar

$$e(k) = d(k) - \hat{y}(k) \quad (2.2)$$

No melhor caso, \hat{y} seria igual à $Ga(k) * h(k)$. O propósito do filtro adaptativo é, deste modo, obter uma estimativa do sinal de eco a partir do sinal de voz original.

É importante notar que a função de transferência da híbrida, representada por $h(k)$ na figura acima, pode mudar lentamente no tempo. Isto significa que uma boa habilidade para acompanhar as mudanças é necessária para o algoritmo de filtragem adaptativa. Estimar a função de transferência $h(k)$, do circuito da híbrida bem no início da conversação e usar este valor até o seu final é, como se vê, um modo ineficiente de atacar o problema. A performance de muitos algoritmos de filtragem adaptativa, degradam quando o usuário B está ativo e $b(k)$ não pode

ser considerado como ruído branco. Por esta razão, canceladores de eco na prática inibem a adaptação dos coeficientes do filtro quando um sinal significativo $b(k)$ do usuário estiver presente (conversa o simult nea ou "double talk") [1].

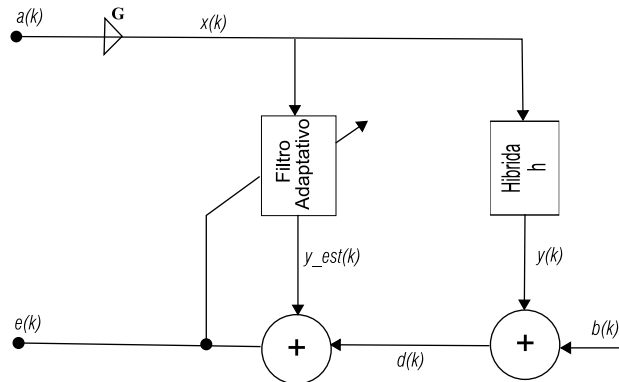


Figura 2.5: Cancelador de eco para linhas telefônicas com filtro adaptativo.

2.4 O Padrão de Cancelamento de Eco G.168

A seguir serão apresentadas as principais características relativas à recomendação G.168 do ITU-T. Esta recomendação define os parâmetros e requisitos necessários a serem utilizados nos projetos de canceladores de eco, que devem ser utilizados nas redes digitais de telecomunicações. A recomendação também define os testes a serem executados nos canceladores de eco, assim como também os sinais de testes a serem utilizados.

2.4.1 O Eco nas Redes Digitais de Telecomunicações

O eco nas redes de telecomunicações têm um grande efeito sobre a qualidade dos circuitos de voz. Dentre as diversas causas de eco nestas redes pode-se destacar as reflexões dos componentes da rede, como por exemplo os conversores de dois (2) para quatro (4) fios (híbrida), o processamento do sinal de voz e os atrasos de transmissão. O eco, dependendo da sua intensidade, pode causar dificuldade na conversa o entre os usu rios em uma chamada telefônica, assim como também prejudica bastante as transmissões de dados, fax e textos na banda de voz.

Os canceladores de eco para redes digitais de telecomunicações são projetados para eliminar o eco acústico para os usu rios e também possibilitar a transmiss o

com qualidade de dados, fax e texto na banda de voz. A Recomendação G.168 tem as especificações técnicas para os canceladores de eco que permitem a execução desta tarefa. Os canceladores de eco projetados de acordo com esta recomendação devem ser compatíveis entre si e também com aqueles projetados segundo às Recomendações G.165 e G.164.

2.4.2 Características dos Canceladores de Eco para o Padrão G.168

A Recomendação G.168 aplica-se ao projeto de canceladores de eco. Estes devem ter os seguintes requisitos fundamentais: Convergência rápida, baixo nível de retorno do eco quando somente um usuário falar, baixa divergência durante conversação simultânea ("double talk"), detecção de conversação simultânea assegurada, operação confiável para facsimile e transmissão de dados em baixa velocidade na banda de voz (<9,6 kpbs). Além destes requisitos fundamentais, tem-se tornado muito comum a utilização de canceladores de eco operando em *tandem*, especialmente em aplicações de telefonia móvel [9].

Quando os canceladores de eco localizam-se no lado do assinante, os tons de sinalização não passam através dos mesmos, de modo que não é necessária nenhuma ação especial. Quando os canceladores estão do lado do equipamento de sinalização, eles são normalmente desabilitados pelo comutador durante os intervalos de troca de sinalização, a fim de evitar distorções dos tons de sinalização pelo cancelador de eco. Quando os tons de sinalização aparecem simultaneamente nas portas de transmissão e recepção do cancelador (double talk), o sinal de recepção será processado pelo mesmo. A estimativa de sinal produzida pelo cancelador, pode distorcer suficientemente o sinal do lado de transmissão, de modo que ele não será reconhecido adequadamente pela unidade de recepção de sinalização.

De acordo com a Recomendação G.168 os canceladores de eco devem ser desabilitados durante a transmissão das sinalizações padronizadas pelo ITU-T (ITU-T N° 5, N° 6 e N° 7). Isto deve ser feito, visto que a presença de canceladores de eco no circuito poderia interferir com o processo de verificação contínuo durante a sinalização. Assim, é necessário desabilitar os canceladores durante a verificação e reabilitá-los novamente após o processo de verificação da sinalização ter sido encerrado.

2.4.3 Ambiente Experimental do Padrão G.168

2.4.3.1 O sinal de teste CSS (Composite Source Signal)

Os testes recomendados pelo Padrão G.168, usam alguns sinais especiais como por exemplo ruído, tons, sinais de fax grupo 3 e um subconjunto de sinais de fontes compostas (CSS). Neste trabalho, será utilizado para teste o sinal CSS. Este sinal será usado como sinal de entrada para os canceladores de eco a serem testados.

O sinal CSS simula as características da voz, tanto para fala de somente um usuário (“single talk”) quanto conversação simultânea (“double talk”) e seu uso como um sinal de teste facilita a medida de performance dos canceladores de eco para sinais de voz. O sinal CSS, em geral, consiste de sequências diferentes incluindo, sinais de voz e sinais que não são de voz e também pausas. O sinal CSS “*single talk*” será utilizado para teste neste trabalho. A figura abaixo mostra o sinal CSS “*single talk*”.

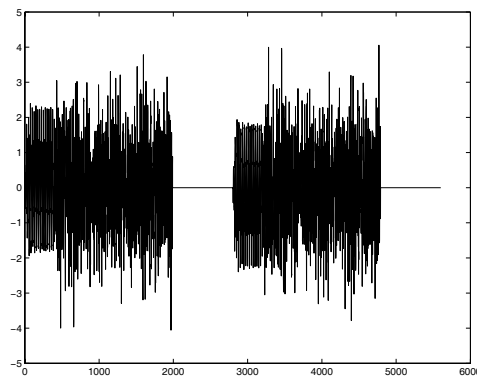


Figura 2.6: Sinal CSS single talk.

O sinal de voz, que é parte do CSS tem como objetivo ativar detectores de voz em sistemas controlados por voz e reproduzir sons de voz real. Como a duração, início e fim do sinal de voz são conhecidos exatamente, este sinal também pode ser usado para medir o tempo de mudança para a direção de transmissão sob teste. Por meio da forma do sinal no domínio do tempo, o tempo de mudança e o tempo de atraso do sistema inteiro pode ser determinado. A duração deste sinal, dentro do CSS, tem duração de aproximadamente 50 ms.

O sinal presente após o sinal artificial de voz no CSS, é o sinal de ruído fictício (Pseudo Noise Signal). Este sinal tem certas características de ruído. A magnitude

da sua transformada de Fourier é inicialmente constante com a frequência enquanto a fase está mudando. Para testes, normalmente somente a magnitude da função de transferência é de interesse, a fase, apesar de não ser importante, também pode ser determinada.

A terceira parte do sinal CSS é uma pausa. Vendo-se o sinal CSS como um sinal de medida que reproduz importantes características da voz real, a pausa tem a função de proporcionar uma modulação de amplitude para o sinal composto. Além disso ela reproduz as pausas reais que ocorrem em um sinal de voz durante uma conversação. Isto significa que haverá um certo período de tempo sem sinal, o que possibilita a análise de ruídos ou outros sinais produzidos pelo sistema que está em teste. O tamanho da pausa dentro do sinal composto é escolhido entre 100 ms e 150 ms.

A figura abaixo mostra como é composto o sinal CSS para o caso “*single talk*”, que é o sinal de teste que será usado nas simulações dos canceladores de eco deste trabalho.

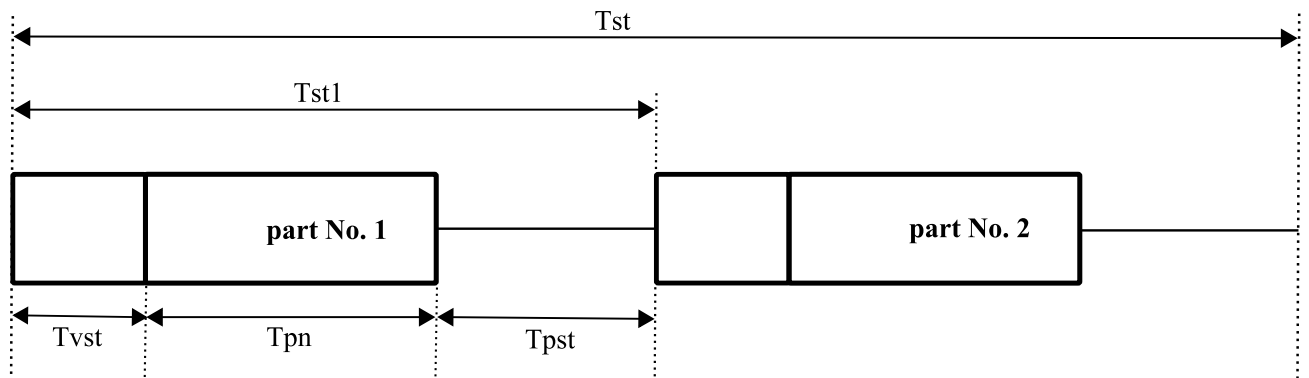


Figura 2.7: Composição do sinal CSS single talk.

Onde:

T_{vst} (Sinal de voz): 48,62 ms

T_{pn} (pseudo ruído): 200,00 ms

T_{pst} (Pausa): 101,38 ms

T_{st1} (Meio período): 350,00 ms

T_{st} (Período completo): 700,00 ms

2.4.3.2 Modelos de Caminhos de Eco para Canceladores de Eco

Os modelos de caminhos de eco apresentados a seguir podem ser usados para os testes previsto na Recomendação G.168. O caminho do eco é simulado por um filtro digital linear com resposta ao impulso $g(k)$. Para se fazer a estimativa para atrasos de vários tamanhos, diferentes valores de ERL (Echo Return Loss) e diferentes características de dispersão e tempo, $g(k)$ é escolhida como uma versão atrasada e atenuada, de qualquer uma das sequências $m_i(k), i = 1, 2, \dots, 8$, para os modelos de canais de 1 à 8 que serão descritos a seguir.

$$g(k) = (10^{-ERL/20} K_i) m_i(k - \delta) \quad (2.3)$$

As sequências $m_i(k)$ representam os vários tipos de caminhos de eco (1 à 8). O atraso δ deve ser escolhido de modo que somente valores diferentes de zero de $g(k)$ possam ser utilizados pelo cancelador de eco. O fator escalar K_i depende do sinal de entrada usado no teste. Os valores de K_i para os modelos de caminho de eco estão mostrados na tabela abaixo. Estes valores são para testes que usam o sinal CSS ou ruído branco como sinais de entrada e resultam em um ERL que é equivalente ao valor usado na equação acima quando medido com o sinal CSS.

Modelo de Caminho	Fator K_i	Mínimo ERL para CSS(dB)
1	1,39E-05	6
2	1,44E-05	6.55
3	1,52E-05	6
4	1,77E-05	6
5	9,33E-05	6
6	1,51E-05	6
7	2,33E-05	11.06
8	1,33E-05	9.27

Tabela 2.2: Fator escalar K_i para os oitos modelos de caminho de eco digital.

As sequências $m_i(k)$ para $i = 1, 2, 3$ e 4 são modelos de caminhos de eco gerados a partir de um simulador de híbrida da rede. O modelo $m_1(k)$ tem um

pequena dispersão, o $m_2(k)$ tem uma dispersão entre pequena e média, o $m_3(k)$ tem uma dispersão entre média e grande e o modelo $m_4(k)$ tem uma grande dispersão. As tabelas com os valores das amostras da resposta ao impulso para os modelos $m_i(k)$ para $i = 1, 2, 3$ e 4 e suas respectivas curvas de resposta ao impulso, são apresentadas no Apêndice A deste trabalho [9].

As sequências $m_i(k)$ para $i = 5, 6, 7$ e 8 são modelos de caminhos de eco medidos em redes telefônicas reais. As tabelas com os valores das amostras da resposta ao impulso para os modelos $m_i(k)$ para $i = 5, 6, 7$ e 8 e suas respectivas curvas de resposta ao impulso, são apresentadas no Apêndice A deste trabalho.

O modelo 5 de caminho de eco, tem uma única reflexão com uma largura de dispersão em torno de 6 ms. Este modelo tem uma característica no domínio da frequência muito próxima da média do valor da resposta medida do caminho do eco.

O modelo 6 de caminho de eco, tem uma única reflexão. Ele tem uma largura de dispersão maior, em torno de 9 ms. A largura de dispersão grande é devido ao pico espectral em torno de 250 Hz.

O modelo 7 de caminho de eco tem duas reflexões. As respostas ao impulso das duas reflexões são sobrepostas uma à outra. A largura de dispersão deste modelo está em torno de 6 ms.

2.5 Comentários

Neste capítulo tem-se uma descrição do padrão G.168 e do seu ambiente experimental. Esta descrição incluiu a apresentação do sinal de teste CSS (Composite Source Signal) e dos modelos de canais usados nos testes da Recomendação G.168. Fez-se também uma descrição de como ocorre a geração do eco em redes telefônicas e como os atrasos e ecos são percebidos pelos usuários da rede. A seguir, tratou-se da evolução dos diversos métodos utilizados para se fazer o cancelamento de eco em linhas telefônicas, partindo-se inicialmente da inserção de perdas nas linhas até o método atual, que é a utilização de canceladores de eco utilizando os filtros adaptativos.

Capítulo 3

Desempenho dos Algoritmos Adaptativos em Canceladores de Eco

Neste capítulo, será feita uma breve apresentação dos principais algoritmos adaptativos utilizados na aplicação de cancelamento de eco. Em seguida será feita uma avaliação do desempenho de cada um dos algoritmos apresentados, para a aplicação em análise, no que concerne aos parâmetros dos algoritmos. Os algoritmos a serem avaliados são os seguintes: LMS (Least Mean Square), NLMS (Normalized Least Mean Square), RLS (Recursive Least Square), AP (Affine Projection), SM-BNDRLMS (Set-Membership Binormalized Data Reusing LMS) e SM-AP (Set-Membership Affine Projection). Todos os algoritmos a serem estudados, utilizarão como filtro adaptativo o filtro FIR (forma transversal), devido às suas características de estabilidade e confiabilidade.

Para caracterizar a convergência dos coeficientes do filtro nas simulações efetuadas para todos os algoritmos avaliados em *single talk*, considerou-se o seguinte critério: quando o MSE (Mean Square Error) atingiu aproximadamente três vezes o valor máximo da potência instantânea do ruído de referência, que para os experimentos executados é de aproximadamente 0.004.

3.1 Os Filtros FIR e suas características

Um filtro FIR é caracterizado pela seguinte equação diferença,

$$y(n) = \sum_{k=0}^{M-1} h_k x(n-k), \quad (3.1)$$

que tem a seguinte função de transferência,

$$H(z) = \sum_{k=0}^{M-1} h_k z^{-k}. \quad (3.2)$$

O filtro FIR pode ser implementado usando uma estrutura de forma direta (ver Figura 3.1) [4]. Esta implementação é normalmente chamada de um filtro *tapped-delay-line*, onde cada coeficiente do filtro constitui um peso/ponderador (*tap weight*). O filtro FIR é normalmente chamado de um sistema de média móvel, visto que a saída é uma média móvel (ponderada) das últimas M entradas.

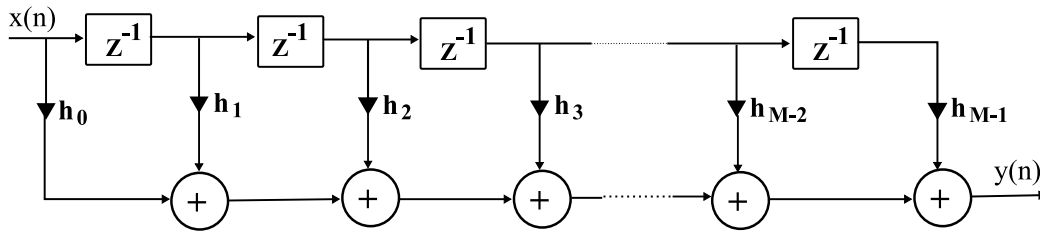


Figura 3.1: Forma direta do filtro FIR.

O filtro FIR tem a vantagem de que ele é naturalmente estável. Ele também é muito bom para remover frequências fixas, visto que a função de transferência pode ter zeros para frequências fixas. Uma desvantagem é que a resposta ao impulso do filtro FIR é limitada no tempo pelo número de multiplicadores (*tap weights*) no filtro. Este é um problema no cancelamento de eco, onde o caminho do eco pode ser bastante longo, necessitando assim de uma longa resposta ao impulso que, com um filtro FIR, só pode ser feito por um grande número de ponderadores/multiplicadores (*tap weights*). Outra desvantagem é que ele não é bom para modelar bandas de frequência estreitas, visto que a função de transferência não tem pólos [2].

3.1.1 O Filtro FIR no Cancelamento de Eco

A propriedade do filtro FIR que o faz naturalmente estável, é muito desejável na aplicação de cancelamento de eco, visto que nenhuma ação especial é necessária para se assegurar a estabilidade do filtro. Deste modo, os filtros FIR são normalmente os mais usados para cancelamento de eco.

3.2 Aplicação do Algoritmo LMS em Canceladores de Eco

O algoritmo LMS (Least Mean Square) foi proposto inicialmente por Widrow e Hoff em 1960 sendo o algoritmo de filtragem adaptativa mais amplamente utilizado na prática. Sua simplicidade devido à ausência de funções de correlação e complicados cálculos de matrizes, são algumas razões que tornou o LMS um padrão com o qual outros algoritmos de filtragem adaptativa são comparados. Outra vantagem do algoritmo LMS é sua robustez à variações de sinal [8].

A convergência dos filtros adaptativos baseados em LMS, é sensível ao espalhamento dos auto-valores da matriz de autocorrelação do sinal de entrada. Isto é particularmente importante no cancelamento de eco, visto que o problema se reduz a uma identificação de sistema que pode ter voz como o sinal de entrada para o filtro adaptativo. As equações recursivas para o algoritmo LMS são mostradas abaixo.

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k) \quad (3.3)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{x}(k)e(k) \quad (3.4)$$

Nas equações acima, $\mathbf{x}(k)$ representa o sinal de entrada, $d(k)$ representa o sinal de referência, $\mathbf{w}(k)$ é a estimativa instantânea do vetor de coeficientes, $e(k)$ é o sinal erro e μ é o fator de convergência (fator de aprendizagem ou tamanho do passo).

O vetor $\mathbf{x}(k)$ é definido como,

$$\mathbf{x}(k) = [x(k), x(k-1), \dots, x(k-N)]^T \quad (3.5)$$

onde N é a ordem do filtro adaptativo. No caso ótimo, quando $\mathbf{w}(k)$ é igual à função de transferência do caminho do eco, o sinal erro $e(k)$ será composto somente do sinal gerado pelo usuário local, eliminando-se assim totalmente o eco.

O LMS é um algoritmo do tipo gradiente que atualiza o vetor coeficiente dando um passo, em cada ciclo de iteração, na direção do gradiente negativo da função objetivo, ou seja:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{\mu}{2} \frac{\partial J_{\mathbf{w}}}{\partial \mathbf{w}(k)} \quad (3.6)$$

Na equação acima $J_{\mathbf{w}}$ é a função objetivo, $\mathbf{w}(k)$ é o vetor de coeficientes do filtro e μ é o tamanho do passo que controla a estabilidade, velocidade de convergência e desajuste do algoritmo. Para encontrar uma estimativa do gradiente, o algoritmo

usa como função objetivo a estimativa instantânea do MSE (Mean-Squared Error), ou seja $J_{\mathbf{w}} = e^2(k)$, resultando na estimativa do gradiente

$$\frac{\partial J_{\mathbf{w}}}{\partial \mathbf{w}(k)} = -2e(k)\mathbf{x}(k). \quad (3.7)$$

O código a ser utilizado para implementação computacional do algoritmo LMS, está mostrado na tabela abaixo.

Tabela 3.1: Código do algoritmo LMS.

Para cada k

{

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

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

}

As propriedades de estabilidade e convergência do algoritmo são determinadas pelo parâmetro μ . Se μ for muito grande, o algoritmo não irá convergir na média quadrática. Por outro lado, se μ for muito pequeno, o algoritmo então ficará muito lento. O algoritmo então convergirá, se e somente se μ estiver dentro da faixa,

$$0 < \mu < \frac{2}{\lambda_{max}} \quad (3.8)$$

onde λ_{max} é o maior auto-valor da matriz $E[\mathbf{x}(k)\mathbf{x}^T(k)]$, que é o traço da matriz de autocorrelação de $\mathbf{x}(k)$.

O algoritmo LMS é muito eficiente computacionalmente. Esta é a principal razão para sua frequente utilização na aplicação de cancelamento de eco, embora ele tenha uma velocidade de convergência menor do que muitos outros algoritmos, como o RLS por exemplo. Quando se implementa o algoritmo LMS, é importante que o tamanho do fator de convergência não seja escolhido muito pequeno, procurando-se prevenir deste modo que o algoritmo tenha uma taxa de convergência muito baixa, e no caso de implementações em precisão finita sem congelamentos [2].

3.2.1 Desempenho do Cancelador de Eco LMS com os Setups do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo LMS, obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 a 8, encontram-se no Apêndice B.

Tabela 3.2: Desempenho do cancelador de eco LMS.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coefficientes	70	70	70	100	80	120	100	100
Fator (μ)	0,01	0,01	0,01	0,005	0,005	0,005	0,008	0,007
Simulações	100	100	100	100	100	100	100	100
Iterações	5531	5531	5531	5501	5521	5481	5501	5501
Atualizações	5531	5531	5531	5501	5521	5481	5501	5501
Convergência	900	850	850	1100	900	1100	1000	1100
ERLE Max(dB)	40	30	30	30	40	30	25	25

Coefficientes: número de coeficientes do filtro

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

3.2.1.1 Gráficos de Desempenho do Algoritmo LMS

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

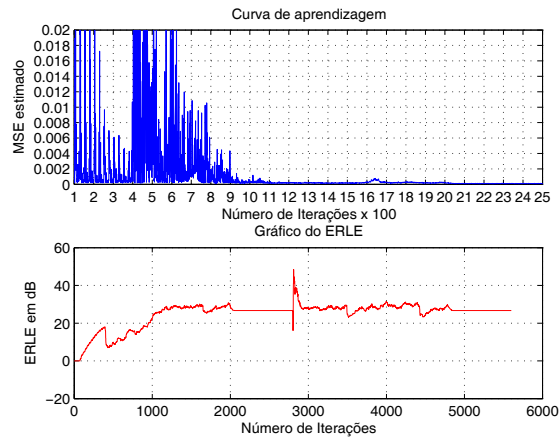


Figura 3.2: Curva de aprendizagem e valor médio do ERLE.

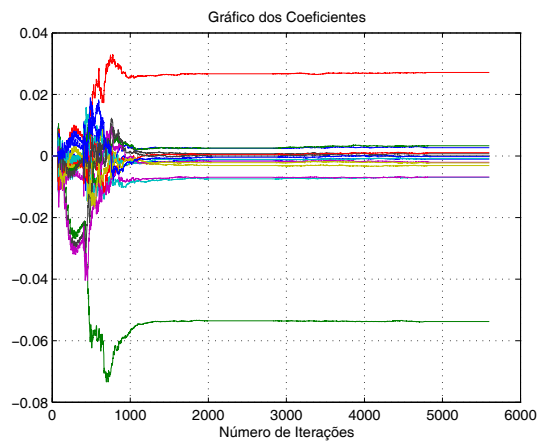


Figura 3.3: Gráfico de convergência dos coeficientes.

3.3 Aplicação do Algoritmo NLMS em Canceladores de Eco

O algoritmo NLMS é uma extensão do algoritmo LMS. Um dos problemas com o algoritmo LMS é que a correção do vetor de parâmetros estimado $\mathbf{w}(k)$ é proporcional ao vetor sinal de entrada $\mathbf{x}(k)$. Deste modo o algoritmo LMS provoca a amplificação do gradiente de ruído. Para evitar este problema, o algoritmo LMS pode ser modificado de modo que o termo de correção $\mu\mathbf{x}(k)e(k)$ é normalizado pela norma Euclidiana da entrada do filtro $\mathbf{x}(k)$. Para evitar problemas com a divisão por zero quando $\|\mathbf{x}(n)\|$ é zero, uma pequena constante de valor positivo α é também adicionada à norma [2].

O algoritmo NLMS é o algoritmo LMS com estas modificações aplicadas, sendo o mesmo resumido nas seguintes equações:

$$y(k) = \mathbf{w}^T(k)\mathbf{x}(k) \quad (3.9)$$

$$e(k) = d(k) - y(k) \quad (3.10)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu}{\alpha + \|\mathbf{x}(k)\|^2} \mathbf{x}(k)e(k) \quad (3.11)$$

Neste caso α é uma constante positiva de pequeno valor e μ é o passo de aprendizagem. O critério de convergência que para o algoritmo LMS é $0 < \mu < \frac{2}{\lambda_{max}}$, torna-se para o algoritmo NLMS o seguinte:

$$0 < \mu < 2 \quad (3.12)$$

Nota-se que a norma Euclidiana $\|\mathbf{x}(k)\|^2$ do vetor de entrada, pode ser calculada recursivamente usando a seguinte equação:

$$\|\mathbf{x}(k)\|^2 = \|\mathbf{x}(k-1)\|^2 + \mathbf{x}(k)\mathbf{x}(k) - \mathbf{x}(k-M)\mathbf{x}(k-M), \quad (3.13)$$

Onde $\mathbf{x}(k)$ é o mais recente valor de entrada escalar e assumiu-se que o comprimento do filtro é M .

Na tabela abaixo, tem-se o código a ser utilizado para implementação computacional do algoritmo NLMS.

Tabela 3.3: Código do algoritmo NLMS.

Para cada k { $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu}{\alpha + \mathbf{x}^T(k)\mathbf{x}(k)}\mathbf{x}(k)e(k)$ }

3.3.1 Desempenho do Cancelador de Eco NLMS com os Setups do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo NLMS, obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 a 8, encontram-se no Apêndice B.

Tabela 3.4: Desempenho do cancelador de eco NLMS.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coefficientes	70	70	70	100	80	120	100	100
Fator (μ)	0,8	0,8	0,8	0,9	0,8	0,8	0,9	0,9
Simulações	100	100	100	100	100	100	100	100
Iterações	5531	5531	5531	5501	5521	5481	5501	5501
Atualizações	5531	5531	5531	5501	5521	5481	5501	5501
Convergência	800	800	800	900	900	1100	1000	1000
ERLE Max(dB)	30	30	30	30	35	30	25	25

Coefficientes: número de coeficientes do filtro

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

3.3.1.1 Gráficos de Desempenho do Algoritmo NLMS

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

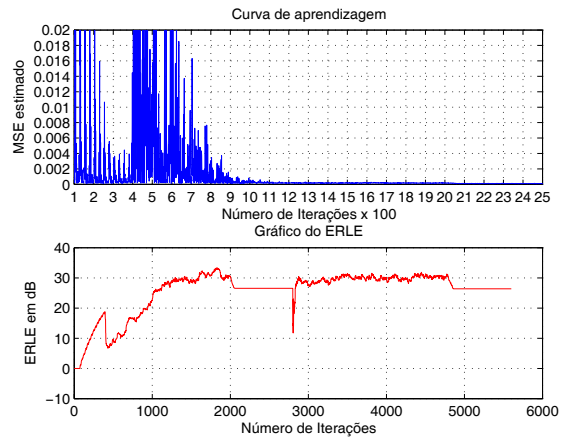


Figura 3.4: Curva de aprendizagem e valor médio do ERLE.

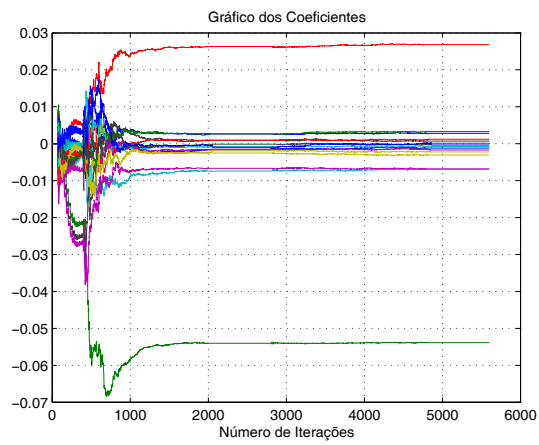


Figura 3.5: Gráfico de convergência dos coeficientes.

3.4 Aplicação do Algoritmo RLS em Canceladores de Eco

Para solucionar o problema da baixa velocidade de convergência do algoritmo LMS, pode-se utilizar a implementação do algoritmo RLS (Recursive Least-Squares). Este algoritmo é uma implementação recursiva da solução LS (Least-Squares), ou seja, ele minimiza a função objetivo mínimos quadrados. Os recursos para a implementação mais comum do algoritmo, que está apresentada na Tabela 4.3, são um resultado da função objetivo WLS (Weighted Least-Squares)

$$J_w = \sum_{i=1}^k \lambda^{k-1} e^2(i) \quad (3.14)$$

Diferenciando a função objetivo J_w em relação a $\mathbf{w}(k)$ e encontrando-se o mínimo para a seguinte equação:

$$\left[\sum_{i=1}^k \lambda^{k-i} \mathbf{x}^T(i) \right] \mathbf{w}(k) = \sum_{i=1}^k \lambda^{k-i} \mathbf{x}(i) d(i) \quad (3.15)$$

onde $0 < \lambda \leq 1$ é um fator de escala exponencial chamado de fator de esquecimento.

Definindo-se as quantidades

$$\mathbf{R}(k) = \sum_{i=1}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \quad (3.16)$$

e

$$\mathbf{p}(k) = \sum_{i=1}^k \lambda^{k-i} \mathbf{x}(i) d(i) \quad (3.17)$$

A solução obtida é:

$$\mathbf{w}(k) = \mathbf{R}^{-1}(k) \mathbf{p}(k) \quad (3.18)$$

As implementações eficientes são um resultado das recursões:

$$\mathbf{R}(k) = \lambda \mathbf{R}(k-1) + \mathbf{x}(k) \mathbf{x}^T(k) \quad (3.19)$$

e

$$\mathbf{p}(k) = \lambda \mathbf{p}(k-1) + \mathbf{x}(k) d(k). \quad (3.20)$$

A matriz inversa $\mathbf{R}^{-1}(k)$ pode ser obtida recursivamente em termos de $\mathbf{R}^{-1}(k-1)$, usando o lema de inversão da matriz, evitando assim a inversão direta de $\mathbf{R}(k)$ em cada instante de tempo k .

Os principais problemas com o algoritmo RLS são o comportamento de divergência em potencial em ambientes de precisão finita e a alta complexidade computacional, que é da ordem de N^2 . Os problemas de estabilidade são normalmente um resultado da perda de simetria e da definição positiva da matriz $\mathbf{R}^{-1}(k)$. Existem implementações mais robustas baseadas na fatorização da raiz quadrada ou decomposição QR da matriz $\mathbf{R}^{-1}(k)$. O código a ser utilizado para implementação computacional do algoritmo RLS, está mostrado na tabela abaixo.

Tabela 3.5: Código do algoritmo RLS.

$\mathbf{R}^{-1}(0) = \delta^{-1}\mathbf{I}; \delta \text{ pequeno, constante e positivo}$ <p>Para cada k</p> <p>{</p> $\mathbf{K}(k) = \mathbf{R}^{-1}(k-1)\mathbf{x}(k)$ $\mathbf{K}'(k) = \frac{\mathbf{K}(k)}{\lambda + \mathbf{x}^T(k)\mathbf{K}(k)}$ $\mathbf{R}^{-1}(k) = \frac{1}{\lambda}[\mathbf{R}^{-1}(k-1) - \frac{\mathbf{K}(k)\mathbf{K}^T(k)}{\lambda + \mathbf{x}^T(k)\mathbf{K}(k)}]$ $e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + e(k)\mathbf{K}'(k)$ <p>}</p>

3.4.1 Desempenho do Cancelador de Eco RLS com os Setups do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo RLS, obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 à 8, encontram-se no Apêndice B.

Tabela 3.6: Desempenho do cancelador de eco RLS.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coefficientes	70	70	70	100	80	120	100	100
Fator (λ)	1	1	1	1	1	1	1	1
Simulações	100	100	100	100	100	100	100	100
Iterações	5531	5531	5531	5501	5521	5481	5501	5501
Atualizações	5531	5531	5531	5501	5521	5481	5501	5501
Convergência	400	400	400	450	450	500	500	450
ERLE Max(dB)	50	40	40	40	45	45	40	40

Coefficientes: número de coeficientes do filtro

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

3.4.1.1 Gráficos de Desempenho do Algoritmo RLS

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

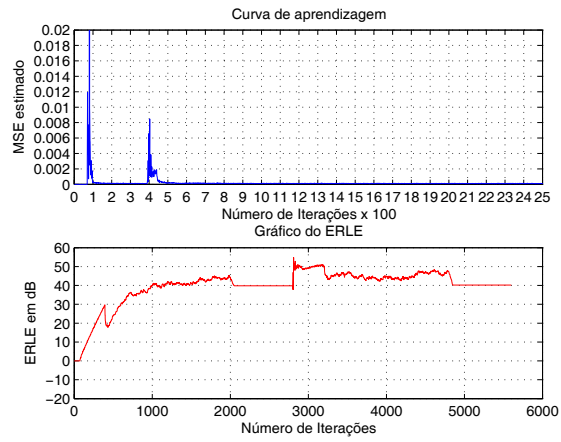


Figura 3.6: Curva de aprendizagem e valor médio do ERLE.

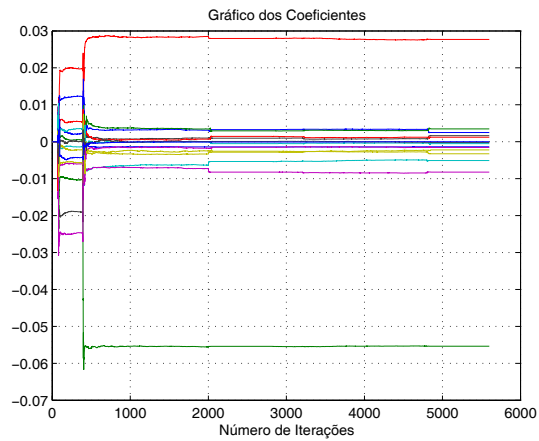


Figura 3.7: Gráfico de convergência dos coeficientes.

3.5 Aplicação do Algoritmo AP(Affine Projection) em Canceladores de Eco

O algoritmo Affine Projection é uma generalização do algoritmo NLMS. Interpretando deste modo, cada atualização do vetor de coeficientes do NLMS é vista como um Affine Projection unidimensional. No Affine Projection as projeções são feitas em múltiplas dimensões, conforme o número de projeções aumenta, aumenta também a velocidade de convergência do vetor de coeficientes e infelizmente também aumenta a complexidade computacional do algoritmo [5].

Enquanto o algoritmo NLMS atualiza os coeficientes baseando-se somente no vetor de entrada atual $\mathbf{x}(k)$, o algoritmo Affine Projection atualiza os coeficientes baseando-se no vetor de entrada atual e em mais P vetores de entrada anteriores adicionais denominados $\mathbf{x}(k), \mathbf{x}(k-1), \mathbf{x}(k-2), \dots, \mathbf{x}(k-P)$. Para $\mu = 1$, o algoritmo Affine Projection atualiza os coeficientes de modo que os erros a *posteriori*

$$\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{w}^T(k+1)\mathbf{X}(k) \quad (3.21)$$

são simultaneamente forçados para zero. De modo equivalente a nova estimativa do coeficiente $\mathbf{w}(k+1)$ gerada pelo APA é a projeção ortogonal da estimativa atual $\mathbf{w}(k)$ na interseção dos hiperplanos $\pi_k, \pi_{k-1}, \dots, \pi_{k-P}$, ou seja, o AP projeta ortogonalmente $\mathbf{w}(k)$ em

$$\bigcap_{n=0}^P \pi_{k-n}. \quad (3.22)$$

A equação de adaptação dos coeficientes do algoritmo AP é dada por

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu \mathbf{X}(k) [\mathbf{X}^T(k) \mathbf{X}(k)]^{-1} \mathbf{e}(k) \quad (3.23)$$

onde $\mathbf{X}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-P)]$ é a matriz formada pelos vetores de entrada atual e passados usada para adaptação e $\mathbf{e}(k) = [e(k/k), e(k-1/k), \dots, e(k-P/k)]^T$ é o vetor formado pela estimativa de erros a *priori* associados. Para evitar problemas numéricos na inversão da matriz, adiciona-se à matriz $\mathbf{X}^T(k) \mathbf{X}(k)$ uma constante de pequeno valor (δ) multiplicada pela matriz identidade \mathbf{I} . Quando P é zero o algoritmo AP reduz-se ao algoritmo NLMS.

Outra interessante interpretação do algoritmo AP é que o vetor usado para adaptar o filtro (ou seja, a direção ao longo da qual os coeficientes do filtro são atualizados) é normalmente chamado de vetor de informação. Para $\mu = 1$ e $\delta =$

0, a adaptação AP é equivalente a usar o componente de $\mathbf{x}(k)$ que é ortogonal à $\mathbf{x}(k-1), \mathbf{x}(k-2), \dots, \mathbf{x}(k-P)$ como vetor de informação, enquanto o NLMS usa o próprio $\mathbf{x}(k)$ como o vetor de informação. Os coeficientes são adaptados de modo que o erro a *posteriori* $e(k/k+1)$ torna-se zero. Além disso, visto que a adaptação é ortogonal à $\mathbf{x}(k-1), \mathbf{x}(k-2), \dots, \mathbf{x}(k-P)$, os erros a *priori* $e(k/k-1)$, para $k = k-1, k-2, \dots, k-P$, permanecem os mesmos que os seus correspondentes erros a *posteriori*, que são zero. Usando o componente de $\mathbf{x}(k)$ que é ortogonal aos P vetores do passado, este tende a não se correlacionar com os elementos individuais do vetor de informação, especialmente se $\mathbf{x}(k)$ for um processo auto-regressivo de ordem pelo menos P .

O código a ser utilizado para implementação computacional do Algoritmo Affine Projection (APA), está mostrado na Tabela 3.7 [13].

Tabela 3.7: Código do algoritmo Affine Projection.

<p>Para cada k</p> <p>{</p> $\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k)$ $\mathbf{t}(k) = [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1}\mathbf{e}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{X}(k)\mathbf{t}(k)$ <p>}</p>
--

3.5.1 Desempenho do Cancelador de Eco Affine Projcton com os Setups do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo AP, obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 à 8, encontram-se no Apêndice B.

Tabela 3.8: Desempenho do cancelador de eco Affine Projection.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coefficientes	70	70	70	100	80	120	100	100
Fator (μ)	0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1
Reusos	10	10	10	10	10	10	10	10
Simulações	100	100	100	100	100	100	100	100
Iterações	5520	5520	5520	5490	5510	5470	5490	5490
Atualizações	5520	5520	5520	5490	5510	5470	5490	5490
Convergência	600	650	700	750	700	750	700	650
ERLE Max(dB)	35	30	30	30	35	30	30	30

Coefficientes: número de coeficientes do filtro

Reusos: número de dados anteriores reutilizados

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

3.5.1.1 Gráficos de Desempenho do Algoritmo AP

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

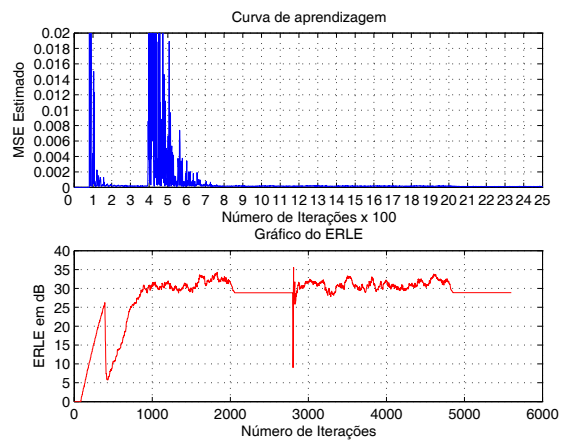


Figura 3.8: Curva de aprendizagem e valor médio do ERLE.

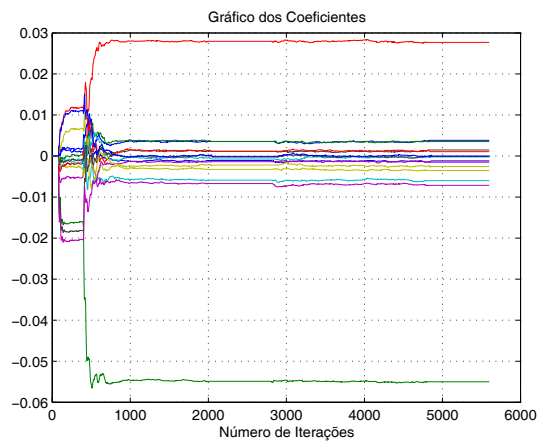


Figura 3.9: Gráfico de convergência dos coeficientes.

3.6 Conceito e Aplicação da Filtragem Set-Membership

A filtragem convencional envolve determinar ou otimizar os coeficientes de um filtro através da otimização de uma função objetivo. A escolha da função objetivo é normalmente feita de modo a facilitar a simplificação analítica e computacional, além de refletir a performance desejada e um conhecimento *a priori*. Como exemplo tem-se o critério estatístico LMS (Least Mean-Square) e o critério determinístico LS (Least-Squares). Em muitas aplicações, entretanto, o projetista tem um conhecimento *a priori* sobre o sistema físico que está gerando as informações de entrada. A filtragem Set-Membership utiliza um conjunto teórico de informações do sistema para estimar um conjunto de parâmetros do filtro que satisfaçam um limite especificado para o erro estimado [7].

3.6.1 A Filtragem Set-Membership (SMF)

Um modelo geral para filtragem linear é mostrado na Figura 3.10. A entrada do filtro no instante de tempo k é o vetor N-dimensional $\mathbf{x}(k)$, a saída do filtro é o valor escalar $y(k)$ e a saída desejada correspondente ao vetor de entrada $\mathbf{x}(k)$ é $d(k)$. A operação do filtro é descrita por

$$y(k) = \mathbf{w}^T \mathbf{x}(k) \quad (3.24)$$

$$e(k) = d(k) - y(k) \quad (3.25)$$

onde \mathbf{w} é o vetor de parâmetros a ser estimado e $e(k)$ é o erro estimado. As classes de filtros modelados pelas equações acima incluem os filtros FIR e IIR que são normalmente utilizados em processamento de sinal e sistemas de telecomunicações.

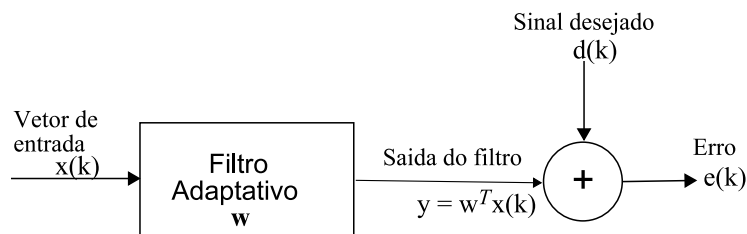


Figura 3.10: Diagrama de representação da filtragem linear.

A filtragem convencional estima o vetor de coeficientes \mathbf{w} de modo a fazer com que o erro estimado $e(k)$ fique pequeno o suficiente a fim de minimizar uma função objetivo.

O objetivo da filtragem Set-Membership (SMF) é alcançar um limite especificado no valor da estimativa de erro e . Qualquer coeficiente estimado que resulte em um erro menor do que o limite especificado, para todos os pares possíveis de entrada e saída desejadas, é uma solução aceitável. Portanto, a solução para o problema da filtragem Set-Membership é um conjunto no espaço dos coeficientes, ao invés de um ponto a ser estimado neste espaço. Chamaremos o conjunto de todos os pares possíveis de entrada e saída desejadas (ou o conjunto de todos os pares de interesse) de espaço de dados, representado por S . Se γ é o limite especificado para a estimativa de erro, o critério da filtragem SMF tem que satisfazer o seguinte:

$$|e|^2 \leq \gamma^2 \quad \forall (x, d) \in S \quad (3.26)$$

De forma equivalente a filtragem SM define uma função custo no espaço S dado por

$$J = \begin{cases} 0 & \text{se } |d - \mathbf{w}^T \mathbf{x}|^2 \leq \gamma^2 \quad \forall (x, d) \in S \\ 1 & \text{Para outros valores.} \end{cases} \quad (3.27)$$

É importante enfatizar que a desigualdade da equação (3.26) é uma especificação do filtro adaptativo. Deste modo, pode-se falar somente de um conjunto de coeficientes prováveis que satisfazem alguns modelos de especificação, como por exemplo o definido na equação (3.26).

A região estimada desejada, chamada de o conjunto de possibilidades \mathbf{w} (coeficientes), é o conjunto de vetores de coeficientes cujo resultado da função objetivo satisfaz à condição abaixo.

$$\mathbf{w} = \bigcap_{(x,d) \in S} \mathbf{w} \in \mathbb{R}^N : |d - \mathbf{w}^T \mathbf{x}|^2 \leq \gamma^2 \quad (3.28)$$

onde \mathbb{R}^N é o espaço Euclidiano N-dimensional. Se o limite do erro γ é escolhido adequadamente, então o conjunto de possibilidades não é vazio e qualquer ponto dentro dele é uma estimativa válida. Se, entretanto, o limite é muito restrito, o conjunto de possibilidades pode ser vazio, para um certo espaço de dados de interesse. Neste caso, pode-se generalizar a formulação do problema substituindo-se o espaço de dados S por um sub-conjunto de S , para que o conjunto de possibilidades

não seja vazio. Questões relacionadas com o conjunto de possibilidades vazio podem ser evitadas colocando-se um problema diferente para a filtragem SMF, que é, minimizar a função objetivo

$$G(\mathbf{w}) = \sup_{(x,d) \in S} |d - \mathbf{w}^T \mathbf{x}|^2. \quad (3.29)$$

A formulação acima procura encontrar o conjunto de possibilidades que minimiza a norma L_∞ do erro e . Neste trabalho, assume-se que γ é grande o suficiente de modo que o conjunto de possibilidades não seja vazio e usa-se a formulação definida na equação (3.26)[6].

3.6.2 Aplicação do Algoritmo Set-Membership Binormalized Data Reuse LMS(SM-BNDRLMS) em Canceladores de Eco

O algoritmo Set-Membership Binormalized Data Reuse LMS(SM-BNDRLMS) utiliza dois conjuntos restritos consecutivos para executar cada atualização dos coeficientes. Estes conjuntos são chamados $\mathcal{H}(k)$ e $\mathcal{H}(k-1)$, sendo que o conjunto $\mathcal{H}(k)$ é definido como sendo o conjunto que contém todos os vetores \mathbf{w} , para os quais o erro de saída associado, no instante de tempo k , tem limite superior em magnitude dado por γ , ou seja,

$$\mathcal{H}(k) = \{\mathbf{w} \in \mathbb{R}^N : |d(k) - \mathbf{w}^T \mathbf{x}(k)| \leq \gamma\}. \quad (3.30)$$

O conjunto $\mathcal{H}(k)$ é chamado de conjunto restrito e os seus limites são hiperplanos [13].

O que este algoritmo procura fazer é minimizar a distância Euclidiana entre os coeficientes anteriores e a nova atualização, fazendo com que os novos coeficientes calculados, estejam contidos na interseção dos conjuntos restritos $\mathcal{H}(k)$ e $\mathcal{H}(k-1)$. Ele utiliza dois passos para executar a atualização dos coeficientes do filtro. Quando os coeficientes calculados no primeiro passo do algoritmo estiverem contidos nos conjuntos restritos $\mathcal{H}(k)$ e $\mathcal{H}(k-1)$, não é feita a atualização do segundo passo do algoritmo. Se a solução fica fora dos conjuntos restritos, o segundo passo de atualização do algoritmo é executado, fazendo com que a solução se desloque para a interseção dos conjuntos $\mathcal{H}(k)$ e $\mathcal{H}(k-1)$. O que o algoritmo Set-Membership Binormalized Data Reuse LMS (SM-BNDRLMS) faz é minimizar a distância $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2$, de modo que $\mathbf{w}(k+1) \in \mathcal{H}(k) \cap \mathcal{H}(k-1)$.

Para o primeiro passo de atualização, utiliza-se uma projeção ortogonal de $\mathbf{w}(k)$ no limite de \mathcal{H} , ou seja

$$\mathbf{w}'(k) = \mathbf{w}(k) + \alpha(k) \frac{e(k)\mathbf{x}(k)}{\|\mathbf{x}(k)\|^2} \quad (3.31)$$

onde

$$\alpha(k) = \begin{cases} 1 - \frac{\gamma}{|e(k)|}, & \text{se } |e(k)| > \gamma \\ 0, & \text{para outros valores} \end{cases}$$

e

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

Se $\mathbf{w}'(k)$ estiver contido em $\mathcal{H}(k-1)$, ou seja o erro *a priori* for menor que γ , então não ocorre uma segunda atualização do algoritmo e $\mathbf{w}(k+1) = \mathbf{w}'(k)$. Se isto não ocorrer, o segundo passo de atualização do algoritmo é executado, de modo que a solução fique na interseção de $\mathcal{H}(k)$ e $\mathcal{H}(k-1)$. Este segundo passo de adaptação do algoritmo SM-BNDRLMS estará na direção de $\mathbf{x}^\perp(k)$ sendo ortogonal ao primeiro passo de adaptação. Abaixo tem-se a equação de atualização do segundo passo do algoritmo.

$$\mathbf{w}(k+1) = \mathbf{w}'(k) + \beta(k) \frac{\epsilon(k-1)\mathbf{x}^\perp(k)}{\|\mathbf{x}^\perp(k)\|^2} \quad (3.32)$$

onde

$$\begin{aligned} \mathbf{x}^\perp(k) &= I - \frac{\mathbf{x}(k)\mathbf{x}^T(k)}{\|\mathbf{x}(k)\|^2} \mathbf{x}(k-1) \\ \epsilon(k-1) &= d(k-1) - \mathbf{w}^T(k)\mathbf{x}(k-1) \\ \beta(k) &= 1 - \frac{\gamma}{\epsilon(k-1)} \end{aligned} \quad (3.33)$$

As equações para o primeiro e segundo passos de atualização do algoritmo estão listados a seguir.

$$\mathbf{w}'(k) = \mathbf{w}(k) + \alpha(k) \frac{e(k)\mathbf{x}(k)}{\|\mathbf{x}(k)\|^2} \quad (3.34)$$

$$\mathbf{w}(k+1) = \mathbf{w}'(k) + \lambda_1 \mathbf{x}(k) + \lambda_2 \mathbf{x}(k-1) \quad (3.35)$$

onde

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

$$\epsilon(k-1) = d(k-1) - \mathbf{w}^T(k)\mathbf{x}(k-1)$$

$$\lambda_1 = -\frac{\beta(k)\epsilon(k-1)\mathbf{x}^T(k-1)\mathbf{x}(k)}{\|\mathbf{x}(k)\|^2\|\mathbf{x}(k-1)\|^2 - [\mathbf{x}^T(k-1)\mathbf{x}(k)]^2}$$

$$\lambda_2 = \frac{\beta(k)\epsilon(k-1)\|\mathbf{x}(k)\|^2}{\|\mathbf{x}(k)\|^2\|\mathbf{x}(k-1)\|^2 - [\mathbf{x}^T(k-1)\mathbf{x}(k)]^2}$$

$$\alpha(k) = \begin{cases} 1 - \frac{\gamma}{|e(k)|}, & \text{se } |e(k)| > \gamma \\ 0, & \text{para outros valores.} \end{cases} \quad (3.36)$$

$$\beta(k) = \begin{cases} 1 - \frac{\gamma}{|\epsilon(k-1)|}, & \text{se } |e(k)| > \gamma e |\epsilon(k-1)| > \gamma \\ 0, & \text{para outros valores.} \end{cases} \quad (3.37)$$

O código a ser utilizado para implementação computacional do algoritmo Set-Membership Binormalized Data Reuse LMS(SM-BNDRLMS), está mostrado na Tabela 3.9 [13].

Tabela 3.9: Código do algoritmo set-membership binormalized data reuse LMS.

<p>Para cada k</p> $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k)$ <p>Se $e(k) > \gamma$</p> <p>{</p> $\alpha(k) = 1 - \gamma/ e(k) $ $a = \mathbf{x}^T(k)\mathbf{x}(k)$ $\mathbf{w}'(k) = \mathbf{w}(k) + \alpha(k)e(k)\mathbf{x}(k)/a$ $\epsilon(k-1) = d(k-1) - \mathbf{x}^T(k-1)\mathbf{w}'(k)$ <p>Se $\epsilon(k-1) > \gamma$</p> <p>{</p> $b = \mathbf{x}^T(k-1)\mathbf{x}(k-1)$ $c = \mathbf{x}^T(k)\mathbf{x}(k-1)$ $\beta(k) = 1 - \gamma/ \epsilon(k-1) $ $den = ab - c^2$ $\lambda_1 = -\epsilon(k-1)c/den$ $\lambda_2 = \epsilon(k-1)a/den$ $\mathbf{w}(k+1) = \mathbf{w}'(k) + \beta(k)[\lambda_1\mathbf{x}(k) + \lambda_2\mathbf{x}(k-1)]$ <p>}</p> <p>}</p> <p>Senão</p> <p>{</p> $\mathbf{w}(k+1) = \mathbf{w}(k)$ <p>}</p>

3.6.3 Desempenho do Cancelador de Eco SM-BNDRLMS com os Setups do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo SM-BNDRLMS, obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para cada um dos canais.

O valor do *threshold* utilizado nas simulações, corresponde a cinco vezes o valor máximo da potência instantânea do ruído de referência.

Tabela 3.10: Desempenho do cancelador de eco SM-BNDRLMS.

Parâmetros	Modelos de Canal-G.168							
	1	2	3	4	5	6	7	8
Coeficientes	70	70	70	100	80	120	100	100
Simulações	100	100	100	100	100	100	100	100
Iterações	5531	5531	5531	5501	5521	5481	5501	5501
Atualizações	1736	1725	1785	1795	1796	1930	1878	1897
Convergência	650	650	650	750	700	900	800	700
ERLE Max(dB)	30	30	30	30	35	30	25	30

Threshold (γ): 7.5E-3

Coeficientes: número de coeficientes do filtro

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

3.6.3.1 Gráficos de Desempenho do Algoritmo SM-BNDRLMS

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

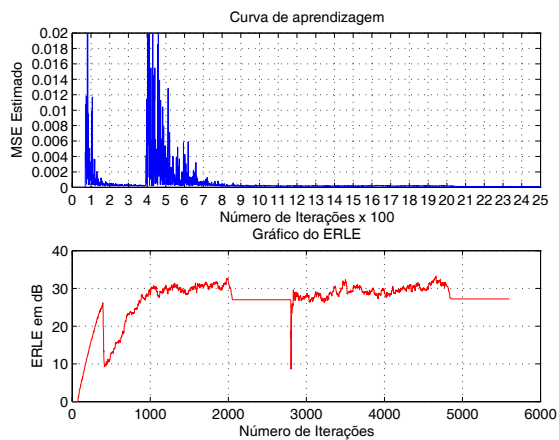


Figura 3.11: Curva de aprendizagem e valor médio do ERLE.

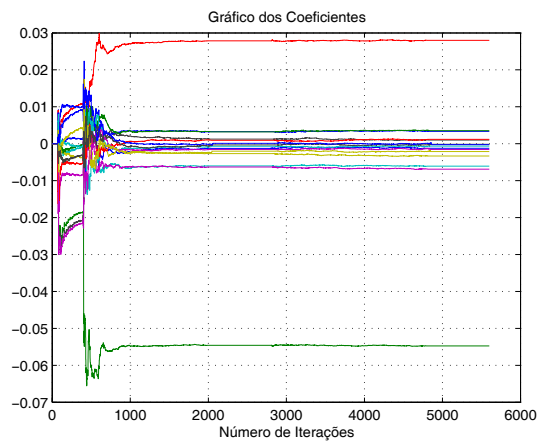


Figura 3.12: Gráfico de convergência dos coeficientes.

3.7 Comentários

Neste capítulo fez-se um estudo do desempenho dos canceladores de eco, utilizando os algoritmos LMS, NLMS, RLS, AP e SM-BNDRLMS, através de simulações com os sinais de teste e modelos de canais da Recomendação G.168. Foram deduzidas as equações utilizadas pelos algoritmos e após as simulações foi apresentada para cada algoritmo, uma tabela com os dados das simulações, assim como também as curvas de desempenho dos mesmos. Foi também feita uma apresentação

dos filtros FIR, que são os filtros utilizados nos canceladores de eco neste trabalho.

Capítulo 4

Avaliação e Otimização do Algoritmo SM-Affine Projection em Canceladores de Eco

O algoritmo Set-Membership Affine-Projection (SM-AP) é mais um algoritmo para filtragem adaptativa, assim como o SM-NLMS (Set-Membership NLMS) e o SM-BNDRLMS (Set-Membership Binormalized Data-Reusing LMS), que utiliza o conceito de Set-Membership para escolha dos coeficientes de um filtro adaptativo. O SM-AP se baseia na utilização de dados dos sinais de entrada e sinal desejado passados, como o objetivo de melhorar a performance do filtro em relação à velocidade de convergência e complexidade computacional. Este algoritmo é uma extensão Set-Membership do algoritmo Affine-Projection (AP), estudado anteriormente. Segundo [13], o algoritmo apresenta, em simulações, especialmente para sinais de entrada coloridos, convergência rápida com desajuste final e complexidade computacional reduzidos.

Os algoritmos que reutilizam dados passados, são fundamentais para a redução da complexidade computacional, quando o sinal de entrada é correlacionado, em comparação ao RLS, que por sua vez apresenta alta velocidade de convergência. O RLS possui alta complexidade computacional. O algoritmo Affine Projection (AP) é um desses algoritmos que permite um compromisso entre velocidade de convergência e baixa complexidade computacional. Com o ajuste da quantidade de dados reutilizados, obtém-se desempenhos melhores do que as dos algoritmos NLMS e RLS. Um dos problemas que pode ocorrer quando se reutiliza dados anteriores, é um aumento no nível de desajuste do algoritmo. Assim, introduz-se um fator de convergência para reduzir este desajuste. A utilização do conceito de filtragem Set-Membership(SMF), é uma outra forma de se controlar o problema do desajuste

nos algoritmos com reutilização de dados passados. Para isto, como se viu anteriormente, especifica-se um limite para o erro o que reduz a complexidade computacional e evita o desajuste do algoritmo.

Pode-se entender o algoritmo Set-Membership Affine-Projection (SM-AP), como uma generalização da reutilização de dados usada nos algoritmos SM-NLMS e SM-BNDRLMS, adotando-se a utilização de P pares de dados anteriores. Os algoritmos SM-NLMS e SM-BNDRLMS utilizam $P = 1$ e $P = 2$, respectivamente. O algoritmo Affine-Projection normal é também um caso particular do algoritmo Set-Membership Affine-Projection (SM-AP), quando o limite definido do erro vai para zero.

4.1 O Algoritmo Set-Membership Affine Projection (SM-AP)

Define-se o conjunto $\Psi(k)$ como sendo a interseção dos P conjuntos restritos anteriores, onde $\mathcal{H}(i)$ é o conjunto que contém todos os vetores \mathbf{w} para os quais o erro de saída está dentro do limite γ , ou seja

$$\psi(k) = \bigcap_{i=1}^{k-P} \mathcal{H}(i) \bigcap_{j=(k-P+1)}^k \mathcal{H}(j) = \psi^{k-P}(k) \cap \psi^P(k) \quad (4.1)$$

Na equação acima $\psi^P(k)$ é a interseção dos P conjuntos restritos anteriores e $\psi^{k-P}(k)$ é a interseção dos $k-P$ conjuntos restritos. Para o algoritmo Set-Membership Affine Projection, os coeficientes atualizados devem estar contidos dentro dos P conjuntos restritos anteriores, o que significa que, $\mathbf{w} \in \psi^P(k)$.

Definindo os parâmetros $g_i(k)$ como $d(k-i+1) - \mathbf{w}^T \mathbf{x}(k-i+1) = g_i(k)$, para $i = 1, \dots, P$, sendo que os vetores \mathbf{w} devem estar contidos em um hiperplano $S_i(k)$, que é o hiperplano que contém todos os vetores \mathbf{w} utilizados para calcular os parâmetros $g_i(k)$. Para que este hiperplano esteja contido em $\mathcal{H}(k-i+1)$, $g_i(k)$ deve obedecer à seguinte condição: $|g_i(k)| \leq \gamma$.

Para atualizar o vetor de coeficientes, sempre que este não esteja contido em $\psi^P(k)$, ou seja, $\mathbf{w}(k) \notin \psi^P(k)$, tem-se o seguinte critério de atualização:

$$\begin{aligned} \mathbf{w}(k+1) &= \min_{\mathbf{w}} \|\mathbf{w} - \mathbf{w}(k)\|^2 \\ \text{tal que : } \quad &\mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w} = \mathbf{g}(k) \end{aligned} \quad (4.2)$$

Nesta equação, $\mathbf{d}(k)$ contém as P saídas desejadas anteriores, $\mathbf{g}(k)$ é o ponto em

$\psi^P(k)$ e $\mathbf{X}(k)$ contém os P vetores de entrada anteriores $\mathbf{x}(k)$. Nas equações abaixo define-se $\mathbf{g}(k)$, $\mathbf{d}(k)$ e $\mathbf{X}(k)$.

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

Para o Set-Membership Affine Projection (SM-AP), a função a ser minimizada é:

$$\mathbf{J}_{\mathbf{w}} = \|\mathbf{w} - \mathbf{w}(k)\|^2 + \boldsymbol{\lambda}^T(k)[\mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w} - \mathbf{g}(k)] \quad (4.4)$$

Na equação (4.4) $\boldsymbol{\lambda}(k)$ é um vetor de multiplicadores de Lagrange. Tomando-se o gradiente de $\mathbf{J}_{\mathbf{w}}$ em relação a \mathbf{w} igual a zero, tem-se que

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)\boldsymbol{\lambda}(k) \quad (4.5)$$

Utilizando-se o critério de otimização da equação (4.2), obtém-se a equação abaixo.

$$\begin{aligned}\mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k+1) &= \mathbf{g}(k) \\ &= \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k) - \mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k)\end{aligned}\quad (4.6)$$

Fazendo-se a arrumação da equação anterior de modo a isolar-se a expressão do erro tem-se:

$$\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k) - \mathbf{g}(k) = \mathbf{e}(k) - \mathbf{g}(k)$$

Calcula-se

$$\boldsymbol{\lambda}(k) = [\mathbf{X}^T(k)\mathbf{X}(k)]^{-1}[\mathbf{e}(k) - \mathbf{g}(k)] \quad (4.7)$$

O vetor erro é dado pela equação a seguir e a expressão $\epsilon(k-i)$ é o erro *a posteriori* na iteração $k-i$.

$$\mathbf{e}(k) = [e(k)\epsilon(k-1)\dots\epsilon(k-P+1)]^T \quad (4.8)$$

A equação de atualização dos coeficientes, utilizando-se as equações deduzidas anteriormente, é apresentada a seguir. Nesta equação, utilizou-se a equação (4.5) para atualizar os coeficientes e o vetor dos multiplicadores de Lagrange utilizada foi a equação (4.7).

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{X}(k)]^{-1}[\mathbf{e}(k) - \mathbf{g}(k)] & \text{se } |e(k)| > \gamma \\ \mathbf{w}(k), & \text{para outros valores} \end{cases} \quad (4.9)$$

4.1.1 O Vetor de Parâmetro $\mathbf{g}(k)$

Apesar de existir um número infinito de possíveis escolhas para os parâmetros $g_i(k)$, para este trabalho se utilizará o caso em que $g_i(k) = \epsilon(k - i + 1)$. Com esta escolha, para $i \neq 0$, serão cancelados todos os elementos $g_i(k)$ da equação (4.7), exceto $g_0(k)$. O valor de $g_0(k)$ é escolhido de modo que a solução fique próxima ao limite de $\mathcal{H}(k)$, ou seja

$$g_1(k) = \gamma \frac{e(k)}{|e(k)|} \quad (4.10)$$

A equação (4.7), para este caso, após as manipulações necessárias reduz-se a seguinte equação

$$\mathbf{X}^T(k)\mathbf{X}(k)\lambda(k) = \alpha(k)e(k)\mathbf{u}_1 \quad (4.11)$$

onde

$$\alpha(k) = 1 - \frac{\gamma}{|e(k)|}$$

$$\mathbf{u}_1 = [1 \ 0 \ \dots \ 0]^T$$

Com isto, as equações de atualização do algoritmo Set-Membership Affine Projection a serem utilizadas neste trabalho são [13]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{X}(k)]^{-1}\alpha(k)e(k)\mathbf{u}_1 \quad (4.12)$$

$$\alpha(k) = \begin{cases} 1 - \frac{\gamma}{|e(k)|}, & \text{se } |e(k)| > \gamma \\ 0, & \text{para outros valores} \end{cases}$$

O código a ser utilizado para implementação computacional do algoritmo Set-Membership Affine Projection (SM-AP), está mostrado na Tabela a seguir.

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

Para cada k { $\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{x}^T(k)\mathbf{w}(k)$ Se $ e(k) > \gamma$ { $\alpha(k) = 1 - \frac{\gamma}{ e(k) }$ $\mathbf{t}(k) = [\mathbf{X}^T(k)\mathbf{X}(k) + \delta I]^{-1}\alpha(k)e(k)\mathbf{u}_1$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)\mathbf{t}(k)$ } } Senão { $\mathbf{w}(k+1) = \mathbf{w}(k)$ } } }

4.2 Desempenho do Cancelador de Eco Set-Membership Affine Projection no Ambiente do Padrão G.168

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo Set-Membership Affine Projection (SM-AP), obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 a 8, encontram-se no Apêndice B.

O valor do *threshold* utilizado nas simulações, corresponde a cinco vezes o valor máximo da potência instantânea do ruído de referência.

Tabela 4.2: Desempenho do cancelador de eco Set-Membership Affine-Projection.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coefficientes	70	70	70	100	80	120	100	100
Reusos	10	10	10	10	10	10	10	10
Simulações	100	100	100	100	100	100	100	100
Iterações	5521	5521	5521	5491	5511	5471	5491	5491
Atualizações	3262	3286	3346	3351	3346	3408	3350	3352
Convergência	550	550	550	600	600	700	650	650
ERLE Max(dB)	30	30	30	25	30	25	25	25

Threshold(γ): 7.5E-3

Coefficientes: número de coeficientes do filtro

Reusos: número de dados anteriores reutilizados

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

4.2.0.1 Gráficos de Desempenho do Algoritmo SM-AP

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

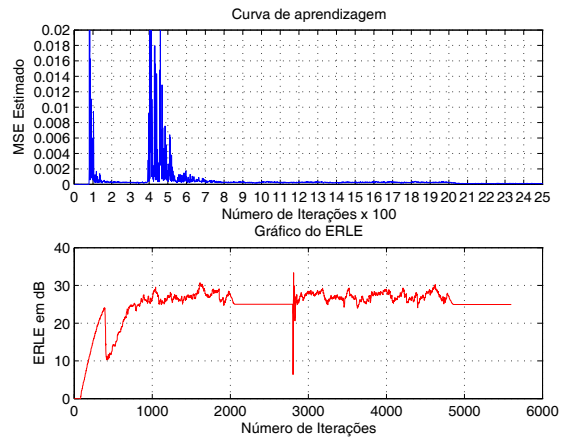


Figura 4.1: Curva de aprendizagem e valor médio do ERLE.

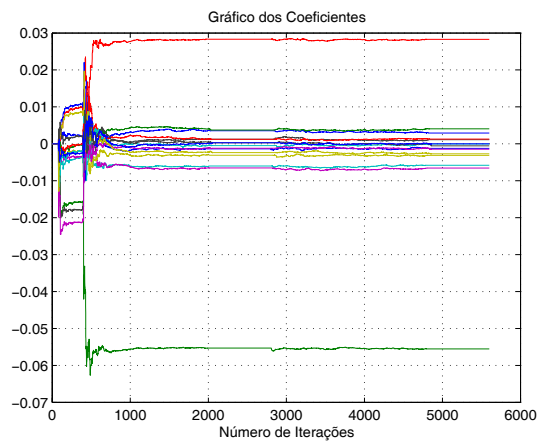


Figura 4.2: Gráfico de convergência dos coeficientes.

4.3 Otimização do Cancelador de Eco SM-AP para o Padrão G.168

Como critério de otimização para o cancelador de eco SM-AP, será utilizada uma função objetivo para $\mathbf{g}(k)$, que faça com que o cancelador melhore sua performance em relação à velocidade de convergência, estabilidade e nível do eco residual após o cancelamento. Utilizaremos, neste caso para o "threshold", ao invés de um único valor γ , uma faixa de valores entre dois limites γ_1 e γ_2 , onde $\gamma_1 > \gamma_2$. Deste modo, procura-se evitar que alguma atualização seja feita para próximo de erros *a posteriori* zero, que não interessam para a aplicação de cancelamento de eco, visto que em um cancelador de eco o objetivo não é zerar o erro e sim fazer com que ele seja exatamente igual ao sinal que vem do usuário distante, livre do eco gerado no caminho. A dedução das equações para se chegar ao algoritmo de atualização, está a seguir.

De [13] tem-se que:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)\boldsymbol{\lambda}(k) \quad (4.13)$$

e

$$\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) = \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k) - \mathbf{g}(k) = \mathbf{e}(k) - \mathbf{g}(k)$$

onde

$$\mathbf{e}(k) = [e(k)\epsilon(k-1)\dots\epsilon(k-P+1)]^T \quad (4.14)$$

é o vetor que contém o erro atual e os erros *a posteriori* na iteração $k-i$, ou seja

$$\epsilon(k-i) = d(k-i) - \mathbf{x}^T(k-i)\mathbf{w}(k) \quad (4.15)$$

Considere que

$$\mathbf{g}(k) = [g_1(k)g_2(k)\dots g_p(k)]^T \quad (4.16)$$

é o vetor de parâmetros \mathbf{g} , calculado de modo que todos os vetores \mathbf{w} , pertençam ao limite de restrição estabelecido, onde $g_i(k) = d(k-i+1) - \mathbf{w}^T\mathbf{x}(k-i+1)$, para $i = 1, \dots, P$ [13].

O que se quer é que: $\gamma_2 \leq |g_i(k)| \leq \gamma_1$.

Tem-se que:

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

Fazendo-se $g_i(k) = \epsilon(k-i+1)$ para $i = 2, \dots, P$, ter-se-á o erro *a posteriori* mantido constante para $i = 2, \dots, P$. Nesse caso

$$\begin{aligned}\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) &= [e(k), \epsilon(k-1), \dots, \epsilon(k-P+1)]^T - [g_1(k), \epsilon(k-1), \dots, \epsilon(k-P+1)]^T \\ &= \mathbf{e}(k) - \mathbf{g}(k)\end{aligned}\quad (4.17)$$

onde devido às considerações feitas anteriormente, $\mathbf{e}(k) - \mathbf{g}(k)$, é um vetor cujo primeiro elemento é $e(k) - g_1(k)$ e os outros elementos do vetor são zero.

Escolhe-se $g_1(k)$ de modo que seu valor fique sempre dentro do limite da faixa de variação de γ que é $\gamma_2 \leq |g_1(k)| \leq \gamma_1$, ou seja:

$$g_1(k) = \gamma_1 \frac{e(k)}{|e(k)|}, \quad \text{para } |e(k)| > \gamma_1 \quad (4.18)$$

e

$$g_1(k) = \gamma_2 \frac{e(k)}{|e(k)|}, \quad \text{para } |e(k)| < \gamma_2 \quad (4.19)$$

Substituindo-se γ_1 e γ_2 pelo fator γ_i nas equações acima tem-se:

$$g(k) = g_1(k) = \gamma_i \frac{e(k)}{|e(k)|}, \quad i = 1, 2 \quad (4.20)$$

Fazendo-se a substituição da equação anterior em (4.17) tem-se,

$$\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) = e(k) - \gamma_i \frac{e(k)}{|e(k)|}$$

$$\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) = e(k) \left[1 - \frac{\gamma_i}{|e(k)|}\right]$$

Fazendo-se $\gamma(k) = 1 - \frac{\gamma_i}{|e(k)|}$ e $\mathbf{u}_1 = [1 \ 0 \ \dots \ 0]^T$ tem-se,

$$\mathbf{X}^T(k)\mathbf{X}(k)\boldsymbol{\lambda}(k) = \mathbf{e}(k)\gamma(k)\mathbf{u}_1$$

$$\boldsymbol{\lambda}(k) = \frac{\mathbf{e}(k)\gamma(k)\mathbf{u}_1}{\mathbf{X}^T(k)\mathbf{X}(k)}$$

Fazendo-se a substituição na equação de atualização dos coeficientes tem-se,

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)[\mathbf{X}^T(k)\mathbf{X}(k)]^{-1}\mathbf{e}(k)\gamma(k)\mathbf{u}_1 \quad (4.21)$$

onde

$$\gamma(k) = \begin{cases} 1 - \frac{\gamma_1}{|e(k)|}, & \text{se } |e(k)| > \gamma_1 \\ 1 - \frac{\gamma_2}{|e(k)|}, & \text{se } |e(k)| < \gamma_2 \\ 0, & \text{se } \gamma_2 \leq |e(k)| \leq \gamma_1 \end{cases}$$

O procedimento de atualização acima descrito, está representado graficamente na figura abaixo e o código a ser utilizado para implementação computacional do algoritmo Set-Membership Affine Projection (SM-AP), otimizado para a aplicação de cancelamento de eco, com uma faixa de variação para o "threshold", está mostrado na Tabela a seguir.

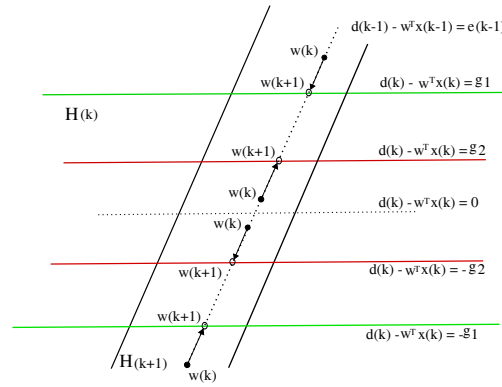


Figura 4.3: Representação gráfica do critério de atualização.

Tabela 4.3: Código do algoritmo Set-Membership Affine Projection Otimizado.

```

Para cada  $k$ 
{
 $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k)$ 
Se  $|e(k)| > \gamma_1$  ou  $|e(k)| < \gamma_2$ 
{
Se  $|e(k)| > \gamma_1$ 
 $\gamma(k) = 1 - \frac{\gamma_1}{|e(k)|}$ 
Senão
 $\gamma(k) = 1 - \frac{\gamma_2}{|e(k)|}$ 
 $\mathbf{t}(k) = [\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}]^{-1}e(k)\gamma(k)\mathbf{u}_1$ 
 $\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{X}(k)\mathbf{t}(k)$ 
}
Senão
{
 $\mathbf{w}(k+1) = \mathbf{w}(k)$ 
}
}

```

4.4 Desempenho do Cancelador de Eco Set-Membership Affine Projection Otimizado

A seguir tem-se uma tabela com os parâmetros mostrando o desempenho do algoritmo SM-AP otimizado para cancelamento de eco obtido durante as simulações, para cada um dos oito modelos de canal utilizando-se o sinal CSS *single talk*. Tem-se também as curvas de aprendizagem e ERLE médio e o gráfico de convergência dos coeficientes, para o modelo de canal 1. Os gráficos correspondentes aos modelos de canais 2 à 8, encontram-se no Apêndice B.

Os valores do *threshold* γ_1 e γ_2 utilizados, correspondem respectivamente a cinco vezes o valor máximo da potência instantânea do ruído de referência (γ_1) e ao valor máximo da potência instantânea dividido por três mil (γ_2).

Tabela 4.4: Desempenho do cancelador de eco SM Affine Projection Otimizado.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
Coeficientes	70	70	70	100	80	120	100	100
Reusos	10	10	10	10	10	10	10	10
Simulações	100	100	100	100	100	100	100	100
Iterações	5521	5521	5521	5491	5511	5471	5491	5491
Atualizações	3256	3297	3351	3358	3341	3406	3362	3348
Convergência	500	550	550	600	600	700	650	600
ERLE Max(dB)	30	30	30	25	30	25	25	25

Threshold(γ_1): 7.5E-3

Threshold(γ_2): 5.0E-7

Coeficientes: número de coeficientes do filtro

Reusos: número de dados anteriores reutilizados

Simulações: número de simulações por experimento

Iterações: número de iterações em cada simulação

Atualizações: número de atualizações dos coeficientes por simulação

Convergência: número aproximado de iterações em que o algoritmo convergiu

4.4.0.2 Gráficos de Desempenho do Algoritmo SM-AP Otimizado

A seguir tem-se os gráficos correspondentes ao modelo 1 de canal.

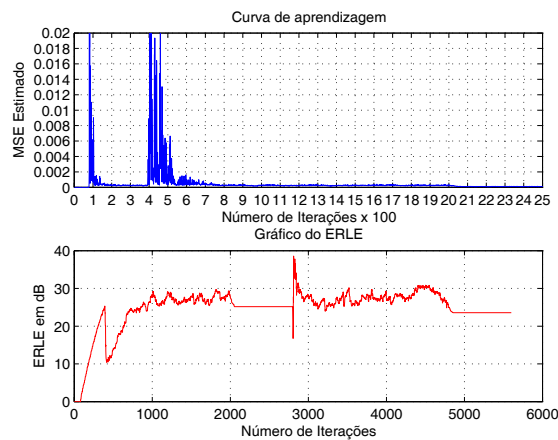


Figura 4.4: Curva de aprendizagem e valor médio do ERLE.

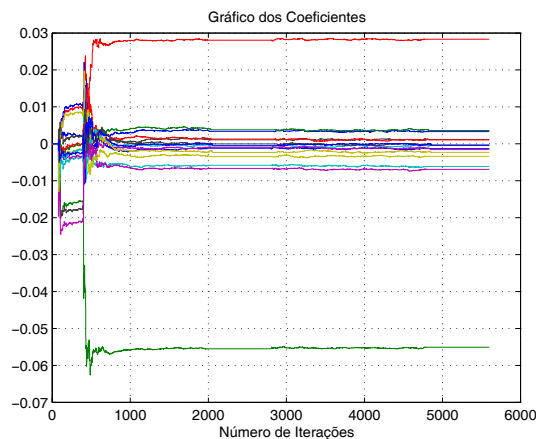


Figura 4.5: Gráfico de convergência dos coeficientes.

4.5 Desempenho do Cancelador de Eco SM-Affine Projection Otimizado em Ambiente Double Talk

Em um ambiente "double talk", o usuário local e o distante estão falando ao mesmo tempo, isto faz com que existam dois sinais chegando ao cancelador de eco simultaneamente. O primeiro sinal é o eco devido à reflexão na híbrida do sinal emitido pelo usuário local e o segundo sinal é o sinal emitido pelo usuário distante que se soma ao eco. A função do cancelador de eco é eliminar o eco e fazer com que

chegue ao usuário local somente o sinal emitido pelo usuário distante. Para avaliar o desempenho do cancelador de eco utilizando o algoritmo SM-AP otimizado, utilizou-se como sinal do usuário local, o sinal CSS (Composite Source Signal) definido na Recomendação G.168 do ITU-T e como sinal do usuário distante um sinal de voz. Para cálculo do "threshold" utilizou-se o desvio padrão da potência do sinal distante e seu valor médio. O valor superior do "threshold" (γ_1), foi calculado como sendo a potência média do sinal distante mais três vezes o desvio padrão da potência do sinal ($m+3s$). Como limite inferior do "threshold" (γ_2) utilizou-se o menor valor de potência instantânea do sinal do usuário distante.

Tem-se a seguir os gráficos das curvas de aprendizagem e ERLE médio e os gráficos referentes aos sinais de erro e ao sinal de voz proveniente do usuário distante. Deve-se observar que passando a existir sinal distante, a curva de aprendizagem, após a convergência, passa a representar o valor médio quadrático (potência) deste sinal. Esta avaliação do cancelador foi feita utilizando-se o modelo de canal 1 com um total de 25 experimentos por simulação.

4.5.0.3 Gráficos de Desempenho do Algoritmo SM-AP Otimizado em ambiente Double Talk

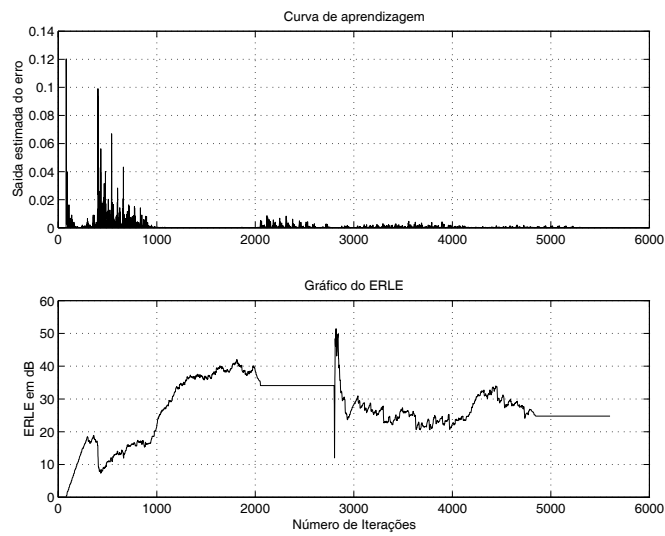


Figura 4.6: Curva de aprendizagem e valor médio do ERLE.

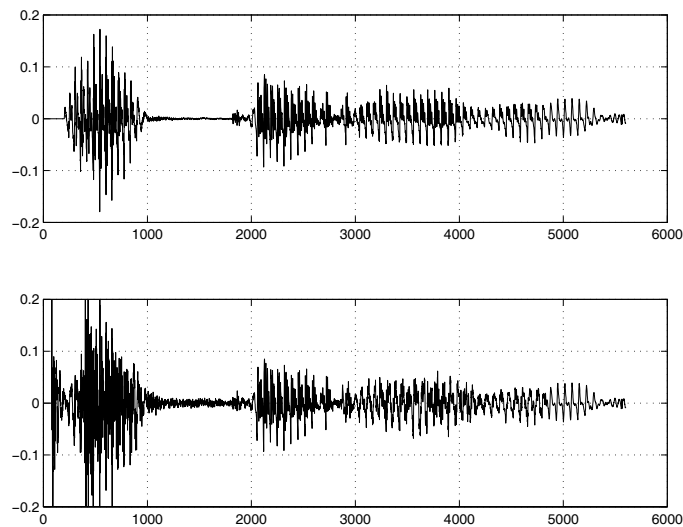


Figura 4.7: Sinal do usuário distante e sinal erro após o cancelador.

4.6 Comentários

Neste capítulo fez-se a apresentação do algoritmo SM-AP e a dedução das equações utilizadas pelo mesmo. Fez-se também a avaliação do desempenho do algoritmo em canceladores de eco utilizando os sinais da Recomendação G.168. A seguir

apresentou-se uma forma de otimizar o algoritmo SM-AP para a aplicação de cancelamento de eco e avaliou-se o desempenho deste algoritmo otimizado, apresentando em seguida uma tabela e gráficos para mostrar a performance do novo algoritmo.

Capítulo 5

Comparações do Desempenho dos Canceladores de Eco para o Padrão G.168

A seguir está apresentada uma tabela que mostra o desempenho dos algoritmos, obtidos a partir das simulações executadas com o sinal *single talk* da Recomendação G.168 e os modelos de canais, de 1 à 8 da Recomendação.

Tabela 5.1: Desempenho dos canceladores de eco para o Padrão G.168.

Parâmetros	Modelos de Canal - G.168							
	1	2	3	4	5	6	7	8
No Coeficientes	70	70	70	100	80	120	100	100
No Simulações	100	100	100	100	100	100	100	100
Iterações-LMS	5531	5531	5531	5501	5521	5481	5501	5501
NLMS	5531	5531	5531	5501	5521	5481	5501	5501
RLS	5531	5531	5531	5501	5521	5481	5501	5501
AP	5520	5520	5520	5490	5510	5470	5490	5490
SM-BNDRLMS	5531	5531	5531	5501	5521	5481	5501	5501
SM-AP	5521	5521	5521	5491	5511	5471	5491	5491
SM-AP Otim.	5521	5521	5521	5491	5511	5471	5491	5491
Atualizações-LMS	5531	5531	5531	5501	5521	5481	5501	5501
NLMS	5531	5531	5531	5501	5521	5481	5501	5501
RLS	5531	5531	5531	5501	5521	5481	5501	5501
AP	5520	5520	5520	5490	5510	5460	5480	5480
SM-BNDRLMS	1736	1725	1785	1795	1796	1930	1878	1897
SM-AP	3362	3286	3346	3351	3346	3408	3350	3352
SM-AP Otim.	3256	3297	3351	3358	3341	3406	3362	3348
Convergência-LMS	900	850	850	1100	900	1100	1000	1100
NLMS	800	800	800	900	900	1100	1000	1000
RLS	400	400	400	450	450	500	500	450
AP	600	650	700	750	700	750	700	650
SM-BNDRLMS	650	650	650	750	700	900	800	700
SM-AP	550	550	550	600	600	700	650	650
SM-AP Otim.	500	550	550	600	600	700	650	600
Reusos - AP	10	10	10	10	10	10	10	10
SM-AP	10	10	10	10	10	10	10	10
SM-AP Otim.	10	10	10	10	10	10	10	10

- Threshold (SM-BNDRLMS e SM-AP): 7.5E-03
- Threshold SM-AP Otimizado (γ_1): 7.5E-03
- Threshold SM-AP Otimizado (γ_2): 5.0E-07

5.1 Os algoritmos LMS, NLMS, RLS, AP, SM-BNDRLMS, SM-AP

Comparando-se na tabela anterior os parâmetros referentes ao desempenho dos algoritmos acima citados, nota-se que os algoritmos que têm o melhor desempenho, na aplicação de cancelamento de eco em relação à velocidade de convergência são respectivamente o RLS e o SM-AP. O RLS possui velocidade de convergência ligeiramente melhor do que o SM-AP, onde o número de iterações para convergência do algoritmo vai de cerca de 400, para os modelos de canais 1, 2 e 3, seu melhor desempenho, até 500 iterações, para os modelos de canal 6 e 7, que são os piores desempenhos para o algoritmo. Já a velocidade do algoritmo SM-AP, tem uma variação de cerca de 550 iterações, para os modelos de canais 1, 2 e 3, que é o melhor desempenho para o algoritmo até 700 iterações, que é o pior desempenho do algoritmo para o modelo de canal 6. Logo, em termos de velocidade de convergência, verifica-se que o RLS é superior a todos os algoritmos testados na aplicação. Como, entretanto, é sabido, o algoritmo RLS pode apresentar problemas de estabilidade devido ao condicionamento da matriz de correlação. O algoritmo SM-AP, apesar de ter um desempenho ligeiramente inferior ao RLS, tem a vantagem de não ter problemas de estabilidade, ter uma menor complexidade computacional e também necessitar de menos iterações para atualizar os coeficientes do filtro. Dependendo do valor do "threshold", o número de atualizações dos coeficientes do filtro será maior ou menor, conforme visto no conceito "Set-Membership".

5.2 O algoritmo SM-AP Otimizado versus o RLS e SM-AP

Conforme visto no capítulo 4, o algoritmo SM-AP Otimizado para a aplicação de cancelamento de eco, utiliza o conceito "Set-Membership", tendo-se o *threshold* não como um valor fixo, mas como uma faixa, onde se tem um limite inferior e um limite superior para a variação do erro calculado na simulação. Se o valor do erro estiver dentro desta faixa, não ocorre a atualização dos coeficientes do filtro, estando o valor do erro fora da faixa, ocorrerá então a atualização dos coeficientes. Isto faz com que se reduza bastante o tempo de computação exigido para este algoritmo, visto que se não ocorre a atualização dos coeficientes do filtro, não há necessidade do cálculo das matrizes utilizadas pelo mesmo, reduzindo-se muito sua complexidade

computacional e mantendo ao mesmo tempo suas características de estabilidade.

Comparando-se na tabela acima, os parâmetros referentes ao desempenho do algoritmo SM-AP Otimizado com o desempenho dos algoritmos RLS e SM-AP, verifica-se que o mesmo tem uma velocidade de convergência maior que o SM-AP para os canais 1 e 8 e tem a mesma performance para os canais 2 à 7. Com relação ao RLS, o desempenho em relação à velocidade de convergência é um pouco inferior, entretanto, de todos os algoritmos testados, o SM-AP Otimizado é o que mais se aproxima em termos de desempenho do RLS. A grande vantagem do algoritmo SM-AP Otimizado em relação ao RLS é que, enquanto para o RLS cada iteração resulta em uma atualização dos coeficientes do filtro, o número de atualizações necessárias para o SM-AP Otimizado é bem inferior, conforme pode ser visto na tabela acima. Enquanto que, para o algoritmo RLS, o menor número de atualizações necessárias foi de 5481, para o caso do canal 6 quando o filtro estava utilizando 120 coeficientes, para o SM-AP Otimizado, o maior número de atualizações necessárias dos coeficientes do filtro foi de 3406, para o caso do canal 6 quando o filtro estava usando 120 coeficientes. Isto faz com que o algoritmo SM-AP Otimizado, tenha um desempenho semelhante ao RLS com uma complexidade computacional muito inferior ao mesmo.

Conforme observado durante as simulações, uma forma de diminuir o número de iterações necessárias para a convergência do algoritmo SM-AP Otimizado, é aumentar o número de dados de entrada anteriores reutilizados (projeções). Isto entretanto, acarreta um aumento nas dimensões das matrizes a serem calculadas, o que fará com que o algoritmo se torne mais pesado computacionalmente. Há necessidade de se chegar a um meio termo com relação ao número de projeções, de modo que se possa ter a vantagem de uma convergência mais rápida e um tempo de utilização de máquina razoável.

5.3 Comparação do Cancelador SM-Affine Projection e SM-Affine Projection Otimizado em Ambiente Double Talk

A seguir, será apresentada uma tabela que mostra o desempenho dos algoritmos SM-Affine Projection e SM-Affine Projection Otimizado, para a aplicação de cancelamento de eco em ambiente *double talk*. Para avaliar o desempenho dos dois algoritmos, foram utilizados dois sinais de voz, sendo um dos sinais a voz do usuário local que passa através do modelo de canal de eco e o outro o sinal do usuário distante, que é o erro após o cancelador de eco. Como canal de eco foi utilizado o modelo 1 de canal da recomendação G.168.

Para o SM-AP Otimizado, o valor do *threshold* foi calculado a partir da potência instantânea do sinal de voz remoto, sendo utilizado para o limite superior (γ_1), o valor de cinco vezes a máxima potência instantânea. Para o limite inferior do *threshold*, foi utilizado um valor cinco vezes menor que a máxima potência instantânea do sinal remoto. Estes valores para o *threshold* foram os que fizeram com que o algoritmo tivesse o seu melhor desempenho e chegou-se a eles após a realização de vários testes. Para o SM-AP, o valor do *threshold* utilizado foi o valor de cinco vezes a máxima potência instantânea do sinal remoto, de modo a se ter o mesmo valor máximo utilizado no SM-AP Otimizado e ter-se então um ponto comum para comparação.

Para se comparar o desempenho dos dois algoritmos foram utilizados os seguintes parâmetros: Número de iterações (total de iterações executadas durante a simulação), número de atualizações dos coeficientes do filtro realizadas durante o experimento e a diferença entre o sinal erro do cancelador e o sinal remoto (para este parâmetro foi calculado a norma do vetor resultante da diferença). Para o filtro adaptativo usou-se três situações: a primeira com um filtro com vinte coeficientes, a segunda com um filtro com trinta coeficientes e a terceira com um filtro com setenta coeficientes. Na primeira e segunda situação, o número de coeficientes do filtro é menor do que o tamanho do canal de eco e na terceira situação o número de coeficientes do filtro é maior (o tamanho do canal de eco é 64).

Algoritmo SM-AP - Desempenho em Double Talk - Modelo 1 (G.168)									
Parâmetros	N=20			N=30			N=70		
	P=3	P=5	P=7	P=3	P=5	P=7	P=3	P=5	P=7
Iterações	5578	5576	5574	5568	5566	5564	5528	5526	5524
Atualizações	1313	1305	1272	879	866	855	912	922	910
Diferença	0.329	0.328	0.322	0.245	0.239	0.233	0.273	0.267	0.258
Threshold	6.5e-3			6.5e-3			6.5e-3		

SM-AP Otimizado - Desempenho em Double Talk - Modelo 1 (G.168)									
Parâmetros	N=20			N=30			N=70		
	P=3	P=5	P=7	P=3	P=5	P=7	P=3	P=5	P=7
Iterações	5578	5576	5574	5568	5566	5564	5528	5526	5524
Atualizações	1225	1199	1200	1148	1162	1118	1126	1122	1182
Diferença	0.218	0.219	0.208	0.207	0.198	0.191	0.234	0.228	0.223
Threshold- γ_1	6.5e-3			6.5e-3			6.5e-3		
Threshold- γ_2	2.6e-4			2.6e-4			2.6e-4		

NOTAS: Valor máximo da potência instantânea do sinal remoto: 1.3e-3
Diferença: Sinal erro - sinal remoto
N: Número de coeficientes
P: Número de reusos

Observando-se os dados da tabela, verifica-se que o algoritmo SM-AP Otimizado tem um desempenho melhor do que o SM-AP para o parâmetro "Diferença", em todos os experimentos executados e um desempenho equivalente quando se compara o parâmetro "Atualizações".

A seguir tem-se os gráficos que mostram os sinais utilizados para o melhor desempenho do algoritmo SM-Affine Projection no ambiente *double talk* ($N=30$ e $P=7$).

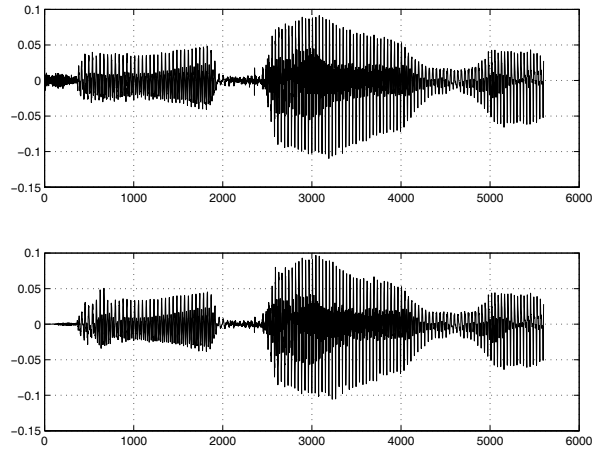


Figura 5.1: Sinal de eco e eco estimado.

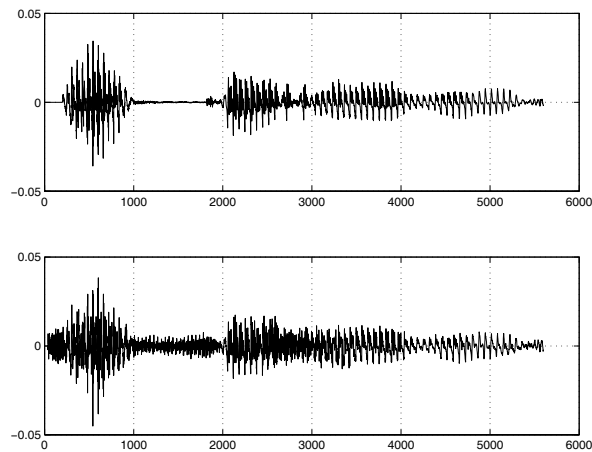


Figura 5.2: Sinal do usuário distante e sinal erro após o cancelador.

A seguir tem-se os gráficos que mostram os sinais utilizados para o melhor desempenho do algoritmo SM-Affine Projection Otimizado no ambiente *double talk* ($N=30$ e $P=7$).

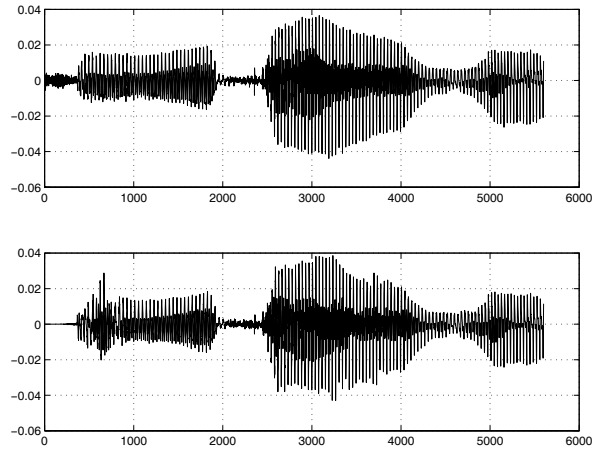


Figura 5.3: Sinal de eco e eco estimado.

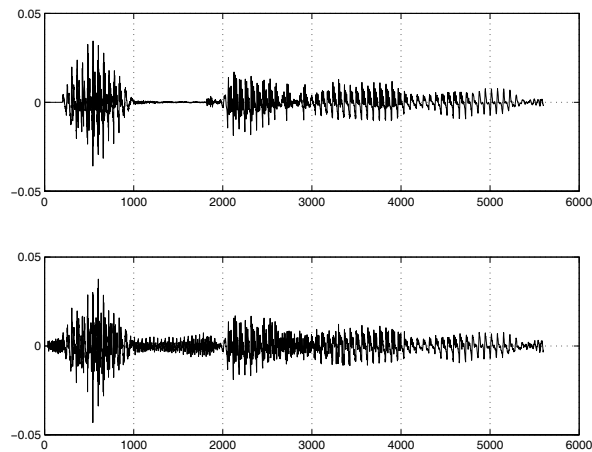


Figura 5.4: Sinal do usuário distante e sinal erro após o cancelador.

5.4 Comentários

Neste capítulo procurou-se fazer uma comparação do desempenho dos diversos algoritmos testados para a aplicação de cancelamento de eco. Verificou-se que os algoritmos RLS, SM-AP Otimizado e SM-AP, são os que alcançam os melhores desempenhos para os parâmetros testados, sendo que o RLS, apesar de ter a melhor performance em relação à convergência, tem problemas de aplicação devido principalmente a sua complexidade computacional, o que não ocorre com os outros dois algoritmos citados. Verificou-se também o desempenho dos canceladores de eco SM-Affine Projection e SM-Affine Projection Otimizado em ambiente *double talk*.

Capítulo 6

Conclusões

Neste trabalho discutiu-se a aplicação de canceladores de eco baseados no padrão G.168, utilizando o algoritmo Set-Membership Affine Projection e buscando uma otimização deste algoritmo para esta aplicação.

As simulações executadas, utilizando-se o sinal de teste *single talk* e os modelos de canais de eco definidos na Recomendação G.168, demonstraram que o cancelador de eco com filtro adaptativo baseado no algoritmo SM-AP, tem um desempenho, em relação à velocidade de convergência, superior a quase todos os algoritmos testados (LMS, NLMS, RLS, AP, SM-BNDRLMS, SM-AP), exceto o RLS. O algoritmo RLS, entretanto, devido à sua alta complexidade computacional, não é recomendado em aplicações onde tempo de processamento é um fator crítico. Assim, o algoritmo SM-AP, torna-se uma boa alternativa para a aplicação de cancelamento de eco.

Para melhorar o desempenho do algoritmo SM-AP para a aplicação em questão, propôs-se um processo de otimização que é baseado na definição de uma faixa para o *threshold* ao invés de um valor definido, conforme proposto em [13]. Este novo algoritmo otimizado, após a dedução das equações e critérios de atualização dos coeficientes do filtro, foi testado submetendo-se o mesmo à simulações no ambiente da Recomendação G.168. O resultado destas simulações, mostrou que o critério de otimização melhorou o desempenho do algoritmo para a aplicação de cancelamento de eco, fazendo com a velocidade de convergência se aproxime ao do RLS. Com isto, demonstrou-se que o algoritmo SM-AP pode ser usado com eficiência para executar cancelamento de eco, apresentando um desempenho melhor que os algoritmos tradicionais, como por exemplo o LMS e o NLMS. Com a otimização proposta, tem-se

uma forma de aumentar a velocidade de convergência e de se garantir a estabilidade que o algoritmo RLS não possui, apesar de ter uma velocidade de convergência maior.

Verificou-se também o desempenho da versão otimizada do algoritmo SM-AP em um ambiente *double talk*. Mesmo neste tipo de ambiente, em que a eliminação do eco é problemática devido ao sinal remoto simultâneo com o sinal local, o algoritmo apresentou um desempenho superior à versão não otimizada, visto que como se viu na tabela de desempenho em *double talk*, mostrada no capítulo anterior, a diferença entre o sinal erro na saída do filtro adaptativo e o sinal remoto, aproxima-se de zero, para o caso de melhor desempenho do algoritmo, mostrando que o eco foi quase totalmente eliminado pelo cancelador.

Deste modo, para a aplicação de cancelamento de eco em linhas baseadas no padrão G.168, a versão otimizada do SM-AP mostrou-se como a melhor solução para os casos analisados neste trabalho, compatibilizando uma alta velocidade de convergência com um número bem menor de atualizações durante os experimentos do que os algoritmos tradicionais.

Referências Bibliográficas

- [1] Silvio A. Abrantes. *Processamento Adaptativo de Sinais*. Fundação Calouste Gulbenkian, 2000.
- [2] Per Ahgren. *An Environment for Real-time Laboratory Exercises in Acoustic Echo Cancellation*. PhD thesis, Uppsala University, Sweden, May 1999.
- [3] Paulo Sérgio Ramirez Diniz. *Adaptive Filtering - Algorithms and Practical Implementation*. Kluwer Academic Publishers, 2nd edition, 2002.
- [4] Paulo Sérgio Ramirez Diniz, Eduardo A. B. Silva, and Sergio L. Netto. *Digital Signal Processing*. Cambridge University Press, Cambridge, UK, 2002.
- [5] Steven L. Gay and Sanjeev Tavathia. The fast affine projection algorithm. *Acoustics Research Department - Lucent Laboratories*.
- [6] Sridhar Gollamudi, Shirish Nagaraj, and Yih-Fang Huang. Smart: A toolbox for set-membership filtering. *European Conference on Circuit Theory and Design*, 1997.
- [7] Sridhar Gollamudi, Shirish Nagaraj, Samir Kapoor, and Yih-Fang Huang. Set-membership filtering and a set-membership normalized lms algorithm with an adaptive step size. *Laboratory for Image and Signal Analysis*.
- [8] Charles Q. Hoang. *LMS Modeling and Estimation of Nonstationary Telecommunication Channels*. PhD thesis, University of Queensland, Australia, October 2000.
- [9] International Telecommunications Union ITU-T. *Digital Network Echo Cancellers*. Telecommunication Standardization Sector of ITU, 2002.

- [10] Jelena Nikolic. Implementing a line-echo canceller using using the block update and nlms algorithms on the tms320c54xdsp. *Associate Technical Staff, DSP Applications SC Group Technical Marketing - Texas Instruments*, April 1997.
- [11] Marcio G. Siqueira. *Adaptative Filtering Algorithms In Acoustic Echo Cancellation and Feedback Reduction*. PhD thesis, University of California Los Angeles, United States of America, 2002.
- [12] David R. Smith. *Digital Transmission Systems*. Van Nostrand Reinhold, New York, USA, 1993.
- [13] Stefan Werner. *Reduced Complexity Adaptive Filtering with Applications to Communications Systems*. PhD thesis, Helsinki University of Technology, Finland, November 2002.

Apêndice A

Modelos dos Canais Digitais - Recomendação G.168

A seguir estão as tabelas com os valores das amostras da resposta ao impulso para os modelos de canais de 1 a 4 e suas respectivas curvas de resposta ao impulso (os valores são lidos em colunas) [9].

Valores da resposta ao impulso do modelo 1 de canal										
-436	46150	390	-3948	-1098	745	1033	899	73	-512	-772
-829	34480	-8191	-2557	-618	716	1091	716	-119	-580	-820
-2797	-10427	-1751	-3372	-340	946	1053	390	-109	-704	-839
-4208	9049	-6051	-1808	-61	880	1042	313	-176	-618	-724
-17968	-1309	-3796	-2259	323	1014	794	304	-359	-685	
-11215	-6320	-4055	-1300	419	976	831	304	-407	-791	

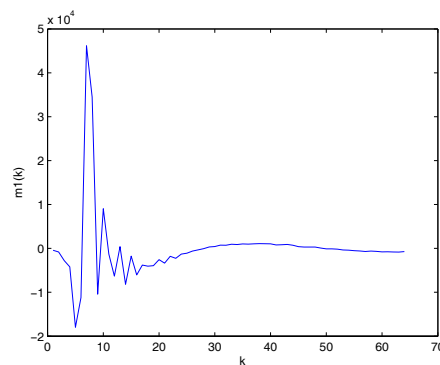


Figura A.1: Curva de resposta ao impulso do modelo 1 de caminho de eco.

Valores da resposta ao impulso do modelo 2 de canal										
-381	-21370	13509	-3858	1316	-1468	789	658	-331	-479	-249
658	-5307	17115	-1979	-693	-1221	954	476	-347	-479	-216
1730	23064	13952	6029	-759	-842	756	377	-430	-512	-249
-51	24020	13952	5616	-1517	-463	839	377	-314	-479	-265
-3511	1020	97	7214	-2176	-298	872	262	-430	-397	-166
-1418	-12374	-9326	6820	-2028	-68	1020	97	-463	-430	-232
7660	-16296	-9046	3935	-2654	64	789	-68	-463	-397	
8861	-19524	-15208	3919	-1814	493	822	-183	-414	-298	
-8106	-7480	-9853	921	-2077	723	558	-232	-381	-265	

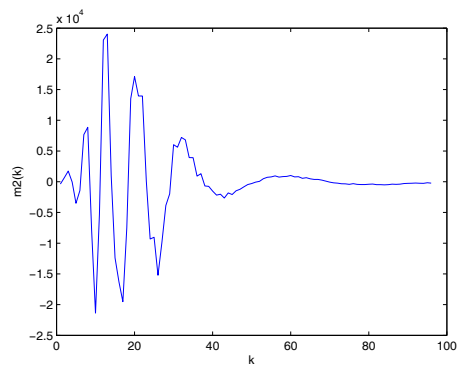


Figura A.2: Curva de resposta ao impulso do modelo 2 de caminho de eco.

Valores da resposta ao impulso do modelo 3 de canal										
-448	-26261	14164	3271	-3101	2172	-139	-1066	-814	-233	-390
-436	-16249	13467	6566	-9269	5387	-573	-1020	-871	-333	-482
2230	21637	4438	4277	-6146	4598	-1100	-1100	-734	-356	-459
2448	25649	8627	11131	-2553	3535	-1157	-1008	-642	-390	-482
-4178	-2267	456	7562	-6272	4004	-1180	-1077	-562	-310	-551
-7050	-10311	-11879	1475	811	2311	-1455	-1088	-356	-265	-573
5846	-4693	-6352	3728	124	2150	-1123	-917	-379	-368	
18581	-12690	-5104	-3525	788	1017	-1386	-917	-345	-310	
2322	-7428	-7496	-7301	5147	330	-1123	-963	-230	-310	

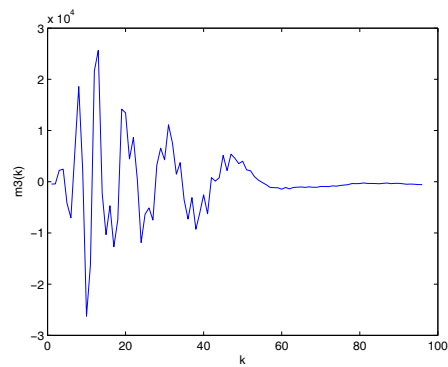


Figura A.3: Curva de resposta ao impulso do modelo 3 de caminho de eco.

Valores da resposta ao impulso do modelo 4 de canal										
160	4041	14624	6850	2393	1592	2654	4617	-220	203	-57
312	14484	-6975	3944	2784	-4752	-881	3576	-306	-111	-24
-241	-1477	-17156	6969	-892	-3646	-4113	2382	257	95	30
-415	-21739	-187	8694	-7366	-5207	-3244	2839	615	-79	-68
897	-4470	149	-4068	-3376	-5577	-7289	-404	225	30	84
908	25356	1515	-3852	-5847	-501	-3830	539	561	84	-155
-1326	11458	14907	-5793	-2399	-1174	-4600	-1803	8	-13	-68
-1499	-19696	4345	-9371	3011	4041	-2508	-1401	344	-68	19
2405	-11800	-7128	453	1537	5647	431	-1705	127	-241	
3347	5766	-2757	1060	6623	4628	-144	-2269	-57	-68	
-3624	789	-10185	3965	4205	7252	4184	-783	182	-24	
-7733	6633	-7083	9463	1602	2123	2372	-1608	41	19	

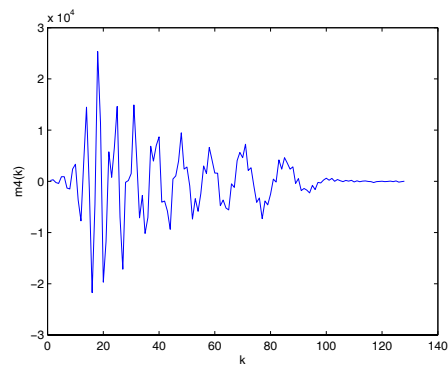


Figura A.4: Curva de resposta ao impulso do modelo 4 de caminho de eco.

A seguir estão as tabelas com os valores das amostras da resposta ao impulso para os modelos de canais de 5 a 8 e suas respectivas curvas de resposta ao impulso (os valores são lidos em colunas).

Valores da resposta ao impulso do modelo 5 de canal										
293	896	20	-22548	3889	-5022	-1608	1640	733	-513	-404
268	604	-938	-43424	7241	-4039	-645	1901	665	-473	-344
475	787	-523	2743	925	-4842	-495	1687	323	-588	-290
460	561	-1438	25897	2018	-4104	279	1803	221	-612	-202
517	538	-1134	7380	-821	-4089	471	1543	-14	-652	-180
704	440	-1887	21499	-2068	-3582	947	1566	-107	-616	-123
581	97	-1727	11983	-2236	-2978	1186	1342	-279	-566	
879	265	-1698	10400	-4283	-2734	1438	1163	-379	-515	
573	-385	-4266	11667	-3406	-1805	1669	963	-468	-485	

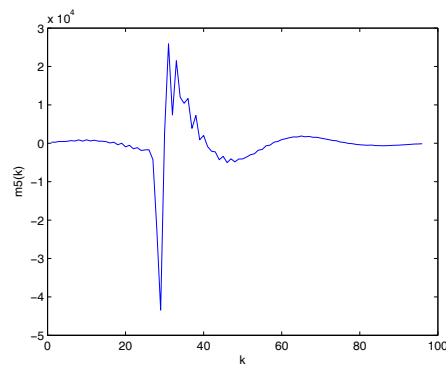


Figura A.5: Curva de resposta ao impulso do modelo 5 de caminho de eco.

Valores da resposta ao impulso do modelo 6 de canal										
29	-450	158	19522	2469	5025	-1117	-1956	1724	482	-838
109	-105	1341	8421	-7994	3946	-2134	-1539	1871	289	-837
-83	-503	195	-50953	490	4414	-2547	-1239	1767	54	-834
198	145	1798	-9043	-3860	4026	-2589	-570	1802	-137	-740
-294	-490	344	18046	-837	3005	-3310	-377	1630	-321	-673
-135	267	1845	-13553	490	3380	-2778	251	1632	-490	-581
-415	-231	629	13336	-636	1616	-3427	331	1379	-638	-493
-202	340	1604	-3471	3682	2007	-2779	964	1271	-764	-436
-444	77	1182	-107	1141	158	-3116	1177	1063	-836	-327
-337	343	940	1788	5019	388	-2502	1449	856	-800	-201
-313	783	5163	-7409	2635	-1198	-2399	1564	711	-859	

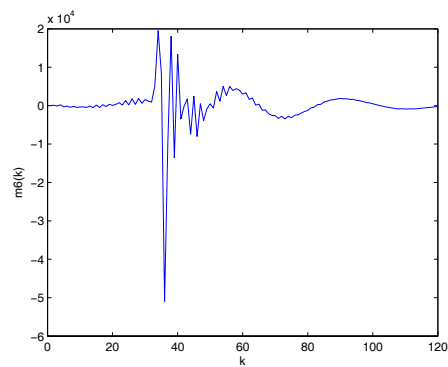


Figura A.6: Curva de resposta ao impulso do modelo 6 de caminho de eco.

Valores da resposta ao impulso do modelo 7 de canal										
258	-343	-1601	8950	18072	-4342	6868	1239	-415	-325	365
-111	-596	-1389	-1574	-14410	-7415	-2195	2	-372	-245	303
337	-177	-13620	758	-7473	7929	3425	-427	-769	-255	251
-319	-1187	-720	3526	19836	-10726	1969	596	-183	-60	230
347	-52	33818	-3118	-16854	6239	-109	-1184	-785	35	209
-434	-1781	-10683	2421	-3115	-2526	3963	551	-270	218	179
192	-147	-6742	-8966	9483	-1317	-1275	-1244	-659	149	
-450	-1959	12489	-4901	-17799	5345	3087	141	-377	340	
-108	-326	-9862	11385	7399	-4565	-892	-743	-523	233	

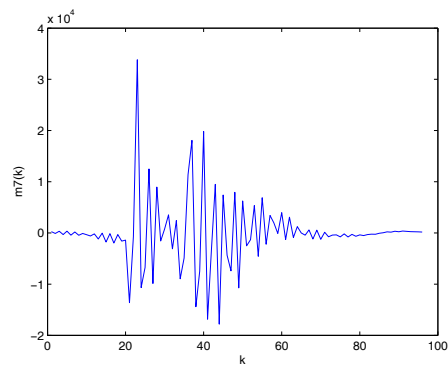


Figura A.7: Curva de resposta ao impulso do modelo 7 de caminho de eco.

Valores da resposta ao impulso do modelo 8 de canal										
80	-102	-1457	33871	-5907	5553	3083	806	452	669	356
31	-26	-229	-176	-10257	-2596	1917	869	538	619	147
4	1002	15659	-16421	5336	3992	1756	471	717	500	107
42	-9250	-6786	18173	-12933	1255	2478	646	723	650	-50
42	-22562	16791	-9669	4348	1450	1027	438	850	615	-88
-61	39321	3860	-10163	-4802	4079	1871	449	756	516	-59
-81	35681	2239	9941	-1791	324	845	432	753	492	-238
-64	-35289	-28730	-19365	3035	4340	1284	473	899	427	-165
-121	25312	-11885	3592	-4433	1059	813	394	555	291	-183

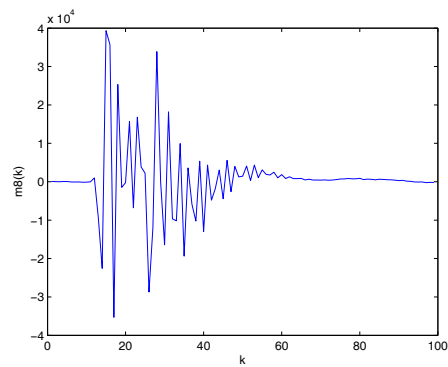


Figura A.8: Curva de resposta ao impulso do modelo 8 de caminho de eco.

Apêndice B

Gráficos de Desempenho dos Algoritmos

1 - Gráficos de desempenho do algoritmo LMS correspondentes aos modelos de canais 2 a 8 (G.168).

Gráficos correspondentes ao modelo 2 de canal.

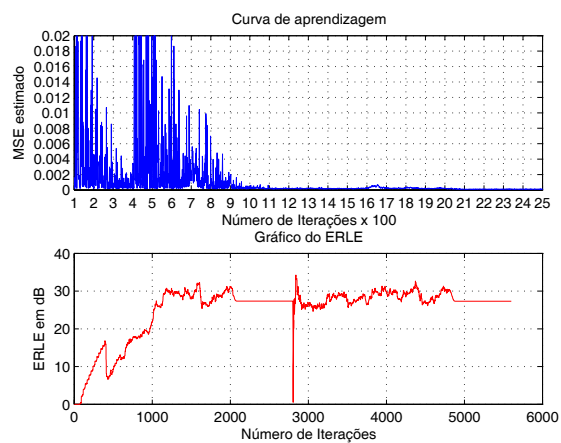


Figura B.1: Curva de aprendizagem e valor médio do ERLE.

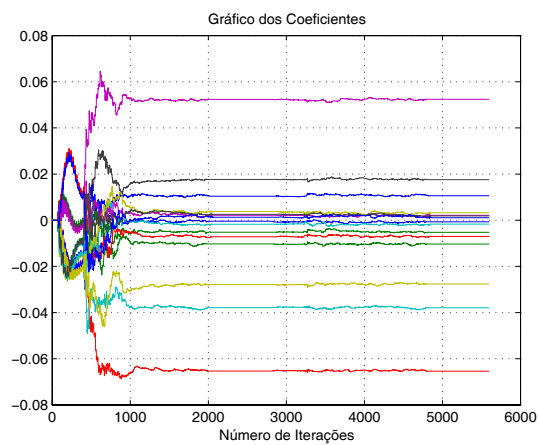


Figura B.2: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

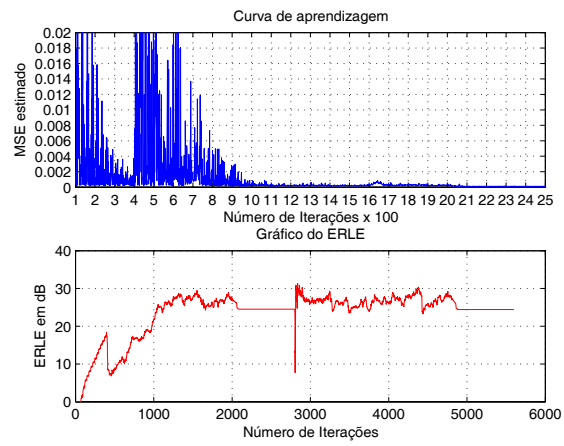


Figura B.3: Curva de aprendizagem e valor médio do ERLE.

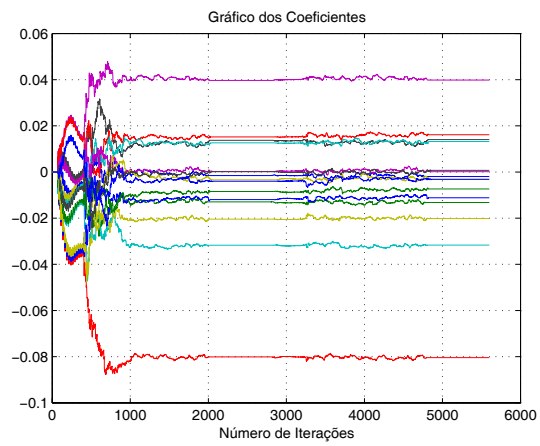


Figura B.4: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

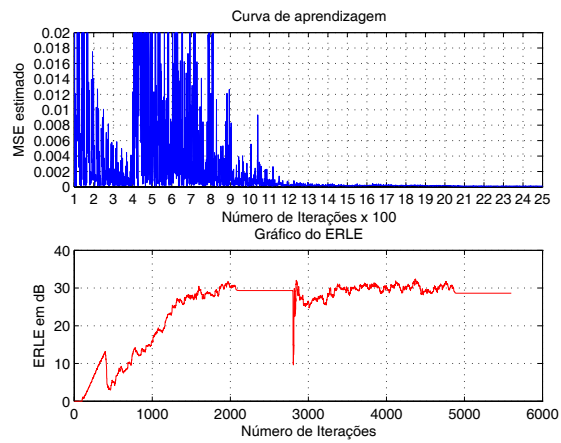


Figura B.5: Curva de aprendizagem e valor médio do ERLE.

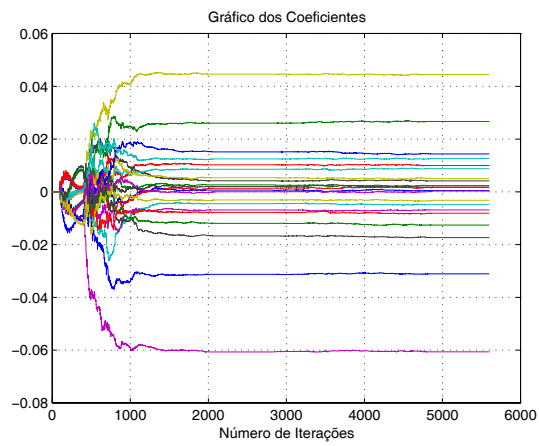


Figura B.6: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

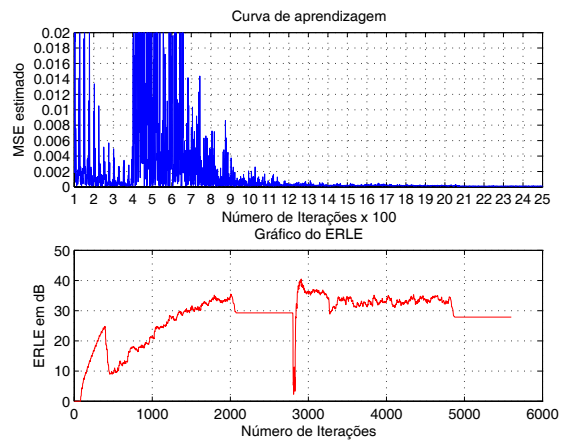


Figura B.7: Curva de aprendizagem e valor médio do ERLE.

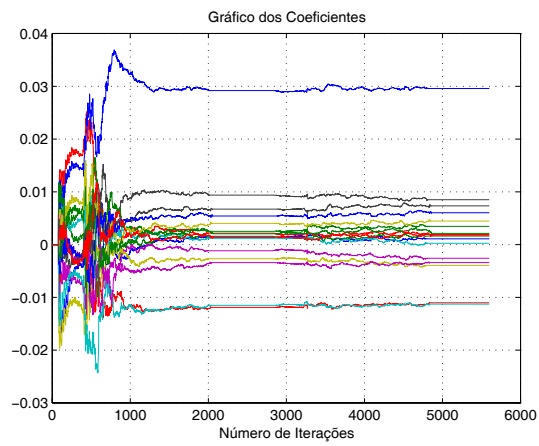


Figura B.8: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

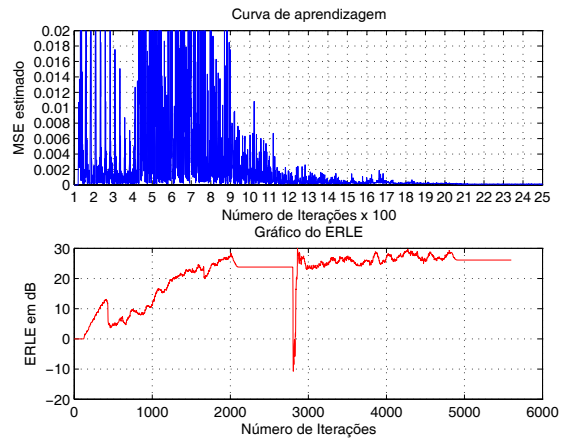


Figura B.9: Curva de aprendizagem e valor médio do ERLE.

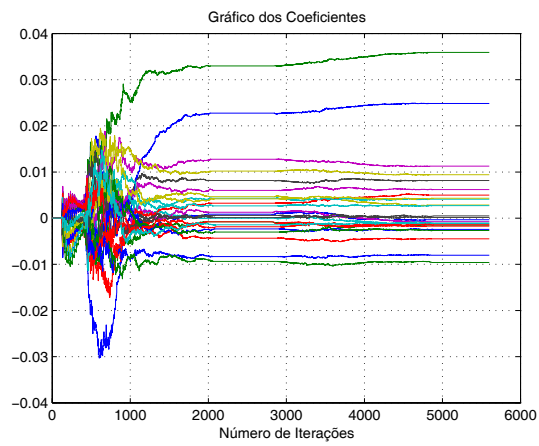


Figura B.10: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

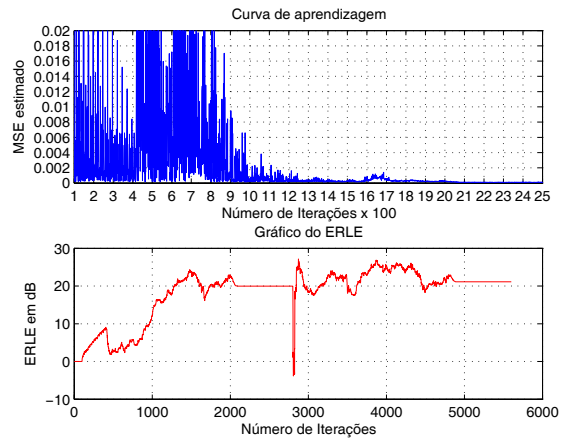


Figura B.11: Curva de aprendizagem e valor médio do ERLE.

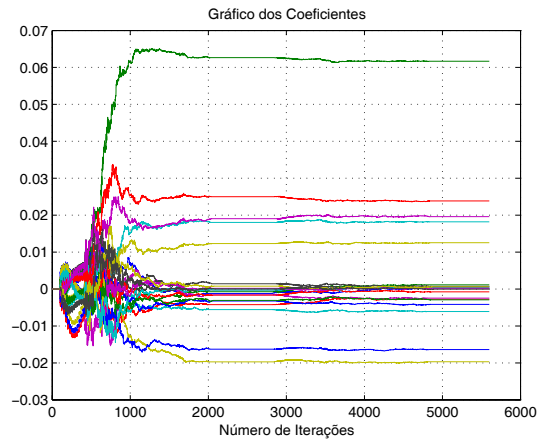


Figura B.12: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

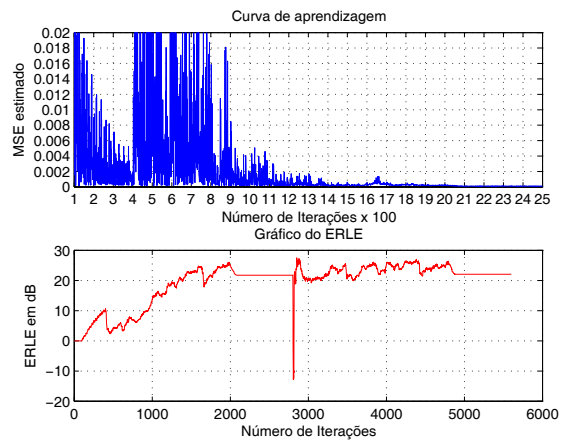


Figura B.13: Curva de aprendizagem e valor médio do ERLE.

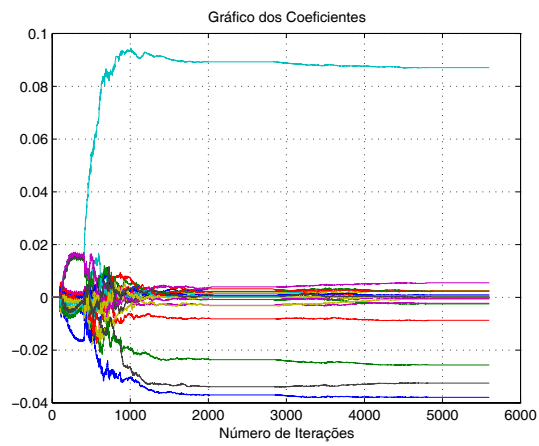


Figura B.14: Gráfico de convergência dos coeficientes.

2 - Gráficos de desempenho do algoritmo NLMS correspondentes aos modelos de canais 2 a 8 (G.168)

Gráficos correspondentes ao modelo 2 de canal.

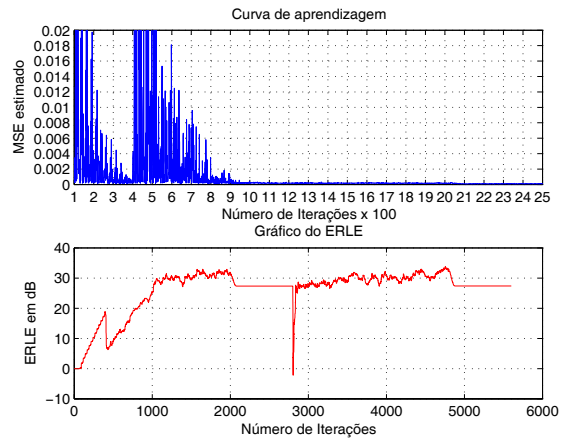


Figura B.15: Curva de aprendizagem e valor médio do ERLE.

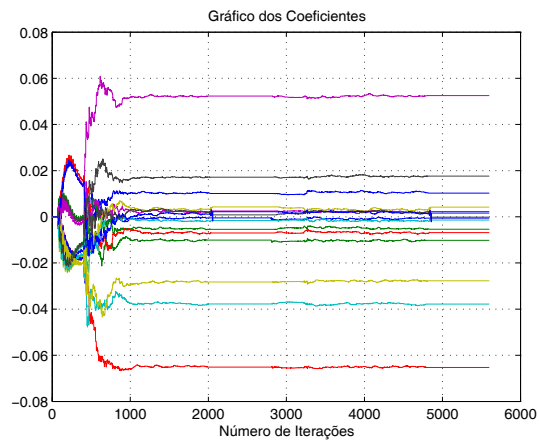


Figura B.16: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

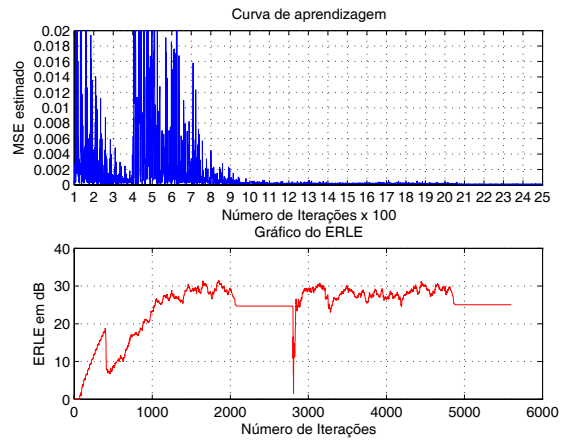


Figura B.17: Curva de aprendizagem e valor médio do ERLE.

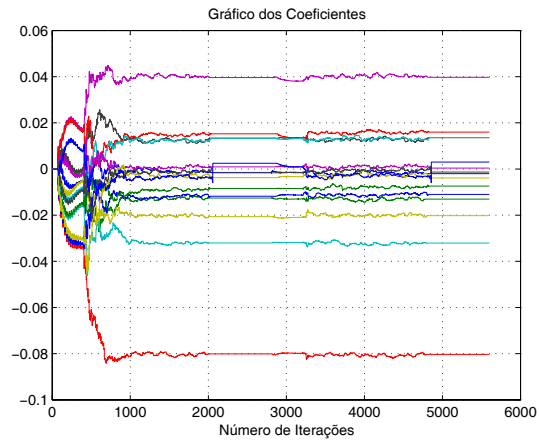


Figura B.18: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

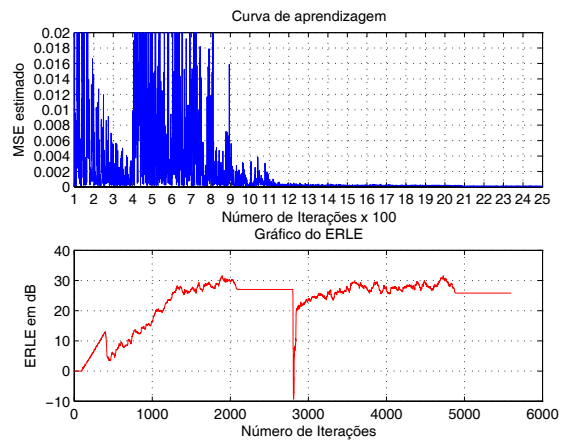


Figura B.19: Curva de aprendizagem e valor médio do ERLE.

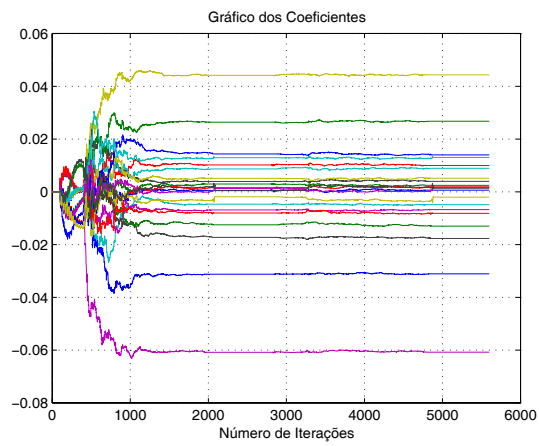


Figura B.20: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

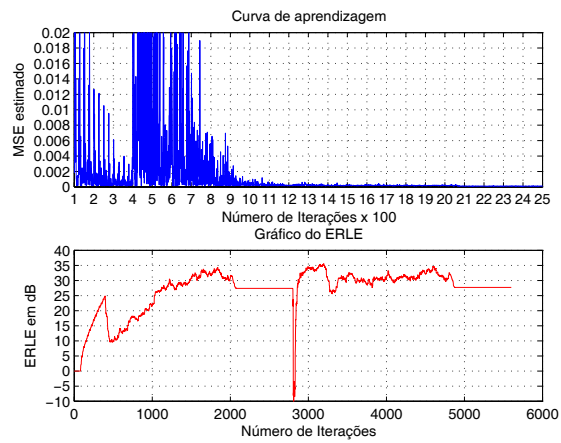


Figura B.21: Curva de aprendizagem e valor médio do ERLE.

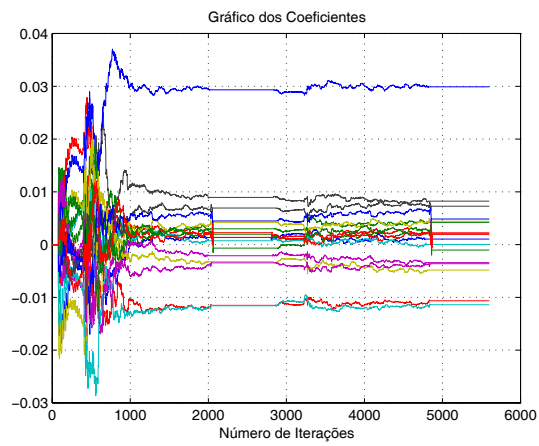


Figura B.22: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

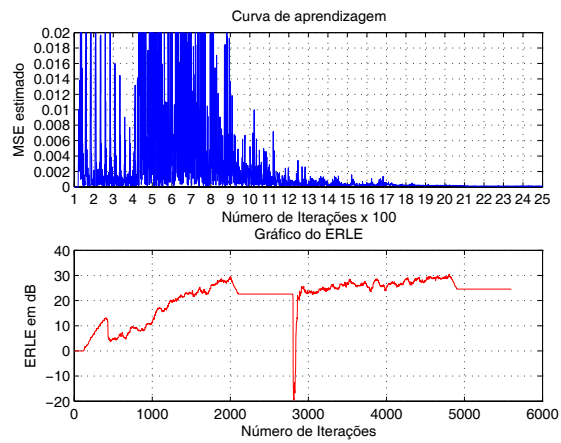


Figura B.23: Curva de aprendizagem e valor médio do ERLE.

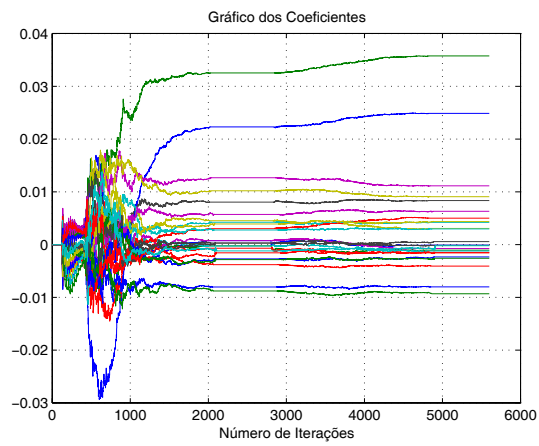


Figura B.24: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

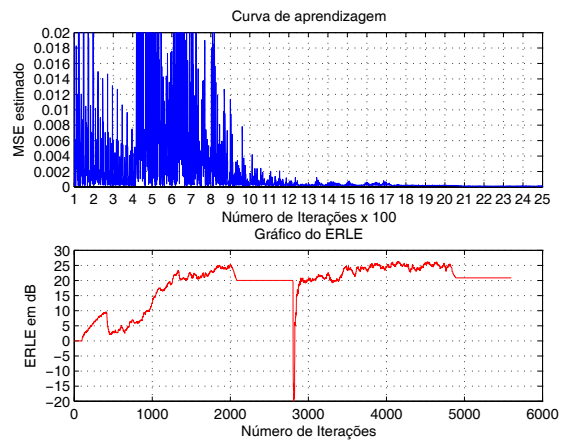


Figura B.25: Curva de aprendizagem e valor médio do ERLE.

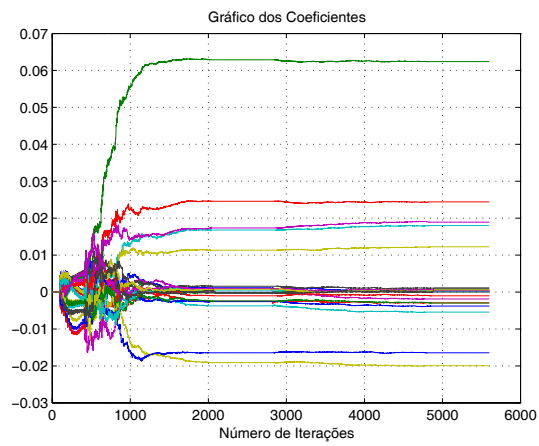


Figura B.26: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

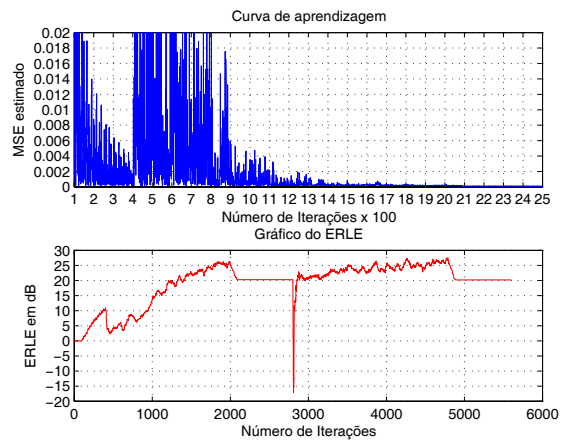


Figura B.27: Curva de aprendizagem e valor médio do ERLE.

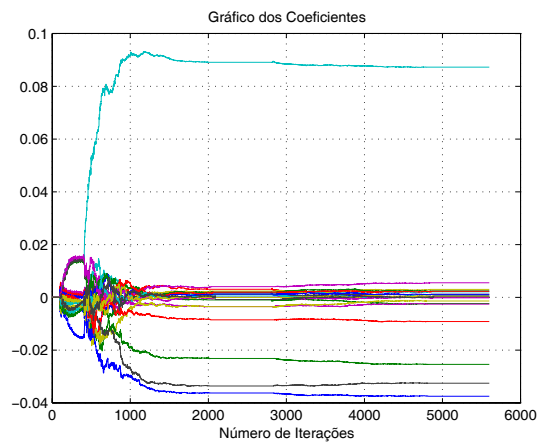


Figura B.28: Gráfico de convergência dos coeficientes.

3 - Gráficos de desempenho do algoritmo RLS correspondentes aos modelos de canais 2 a 8 (G.168)

Gráficos correspondentes ao modelo 2 de canal.

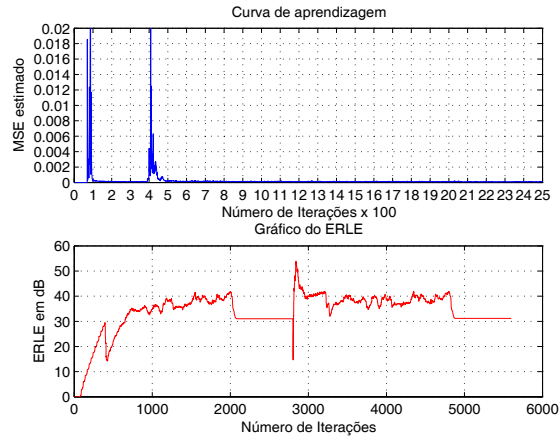


Figura B.29: Curva de aprendizagem e valor médio do ERLE.

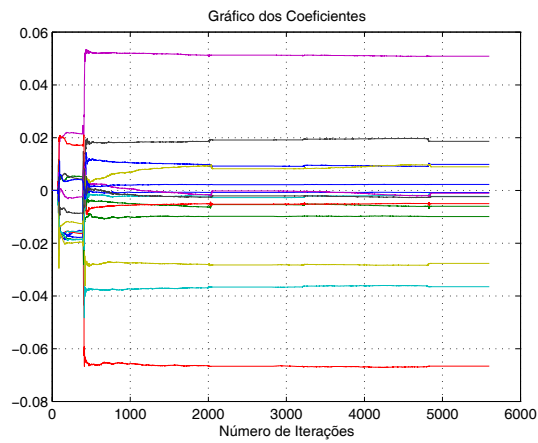


Figura B.30: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

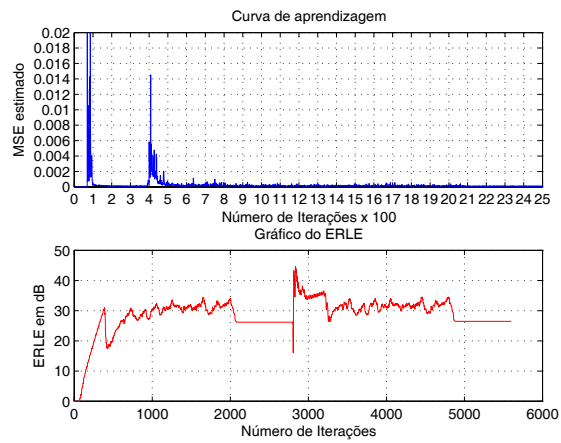


Figura B.31: Curva de aprendizagem e valor médio do ERLE.

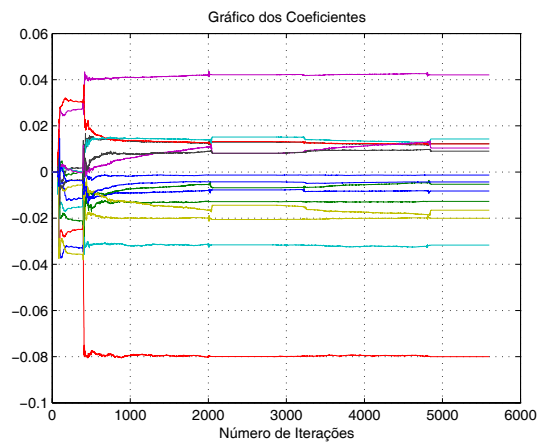


Figura B.32: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

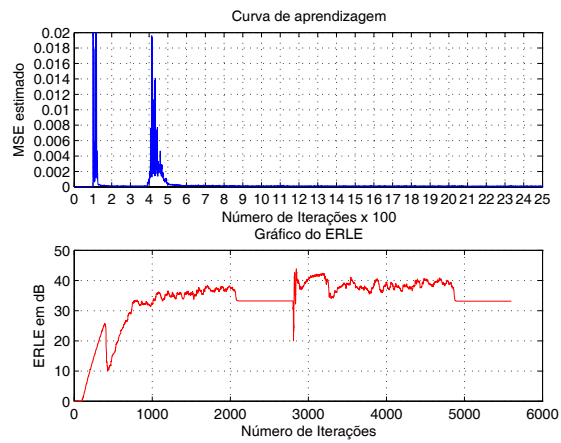


Figura B.33: Curva de aprendizagem e valor médio do ERLE.

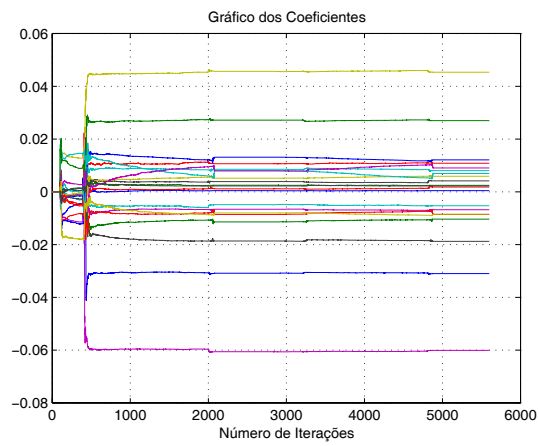


Figura B.34: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

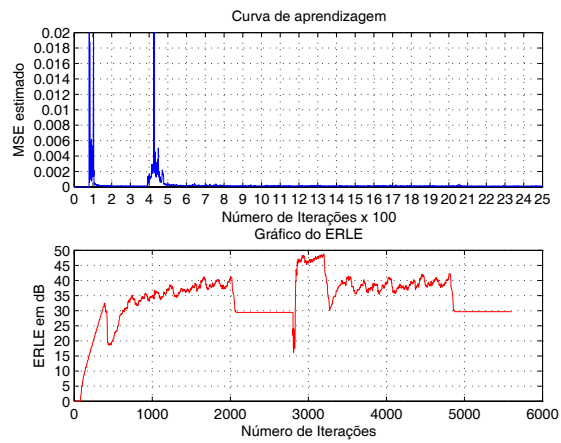


Figura B.35: Curva de aprendizagem e valor médio do ERLE.

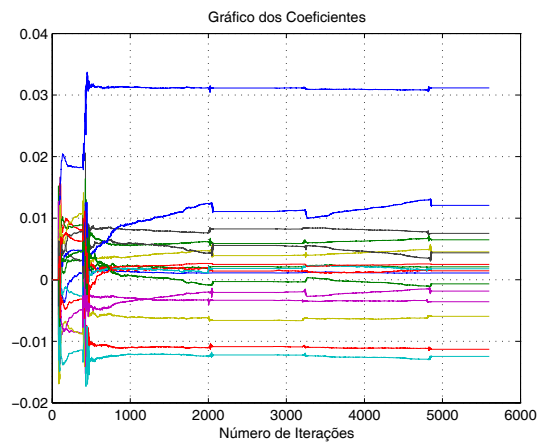


Figura B.36: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

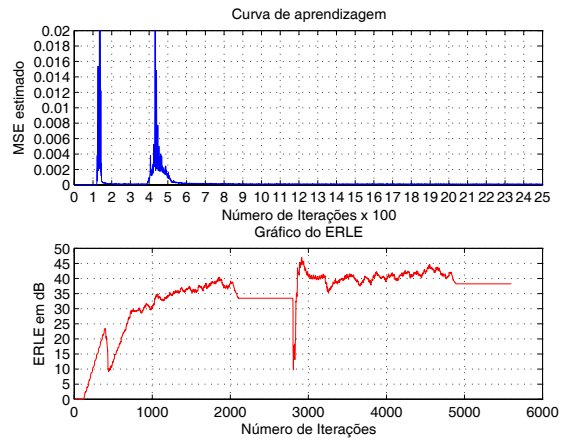


Figura B.37: Curva de aprendizagem e valor médio do ERLE.

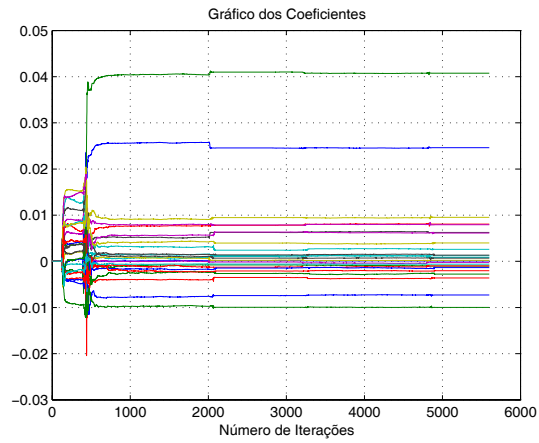


Figura B.38: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

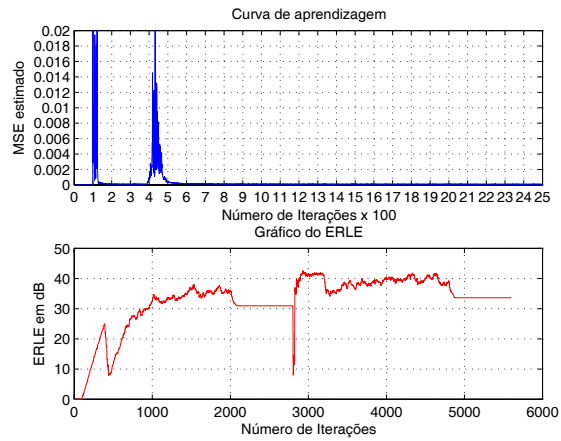


Figura B.39: Curva de aprendizagem e valor médio do ERLE.

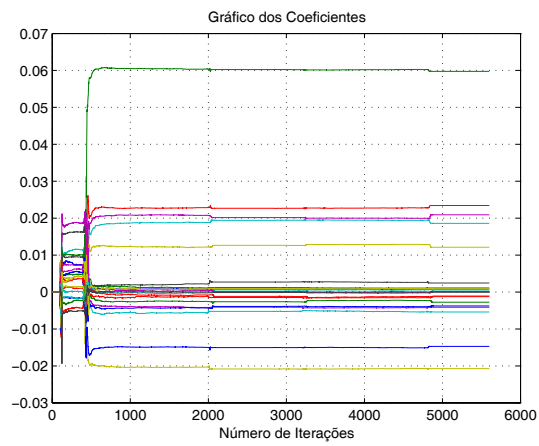


Figura B.40: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

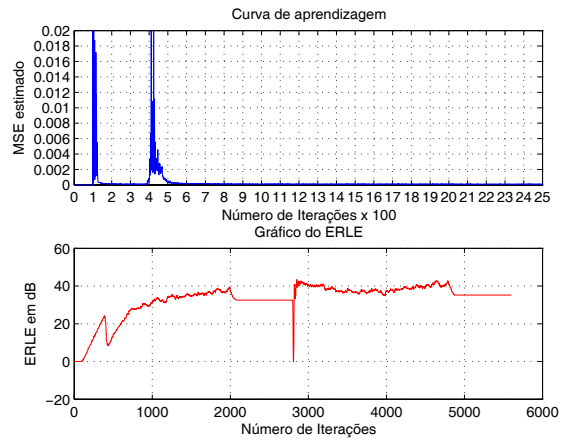


Figura B.41: Curva de aprendizagem e valor médio do ERLE.

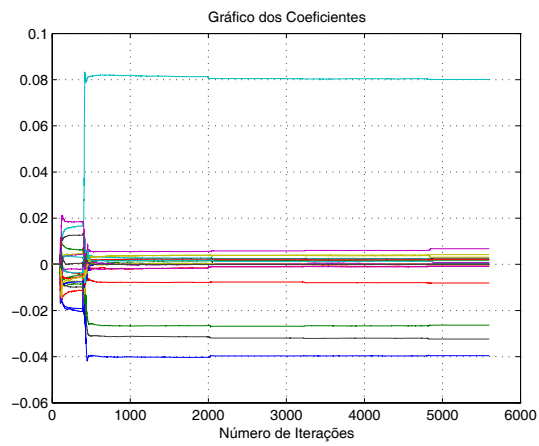


Figura B.42: Gráfico de convergência dos coeficientes.

4 - Gráficos de desempenho do algoritmo AP correspondentes aos modelos de canais 2 a 8 (G.168)

Gráficos correspondentes ao modelo 2 de canal.

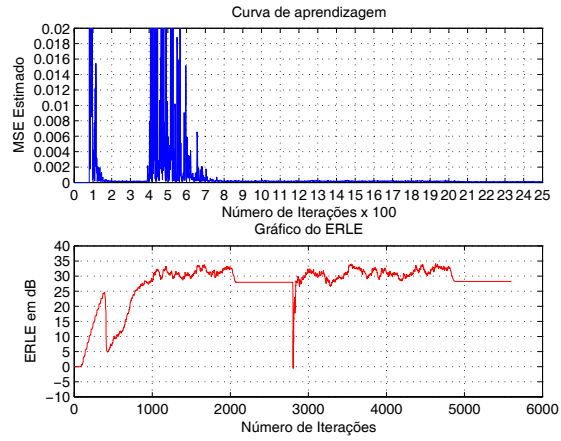


Figura B.43: Curva de aprendizagem e valor médio do ERLE.

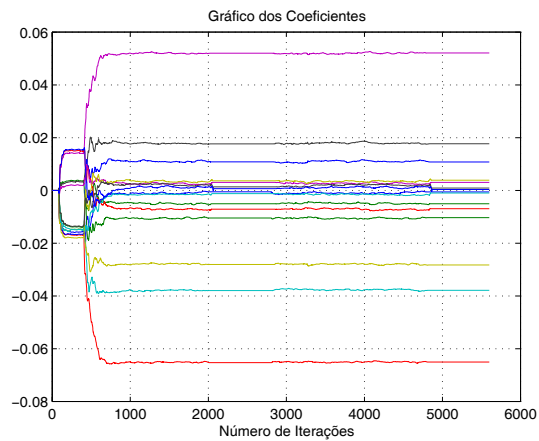


Figura B.44: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

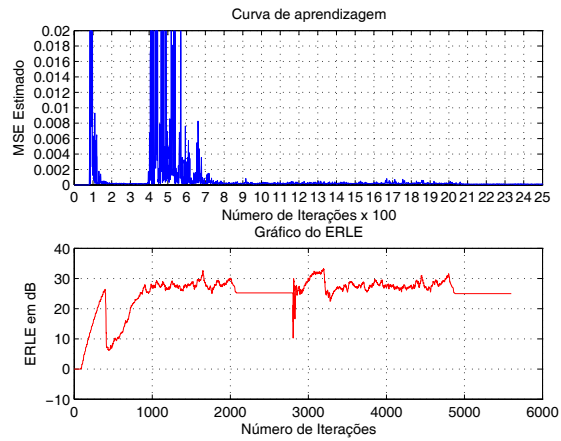


Figura B.45: Curva de aprendizagem e valor médio do ERLE.

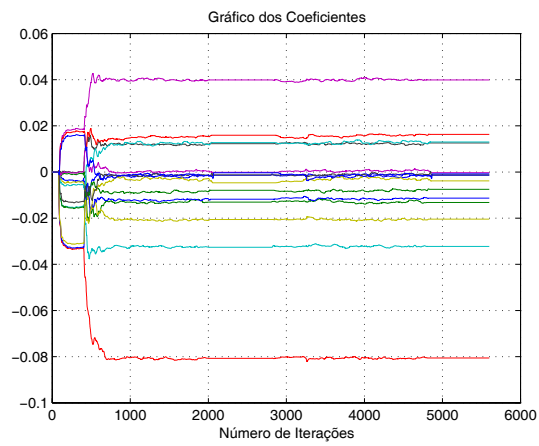


Figura B.46: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

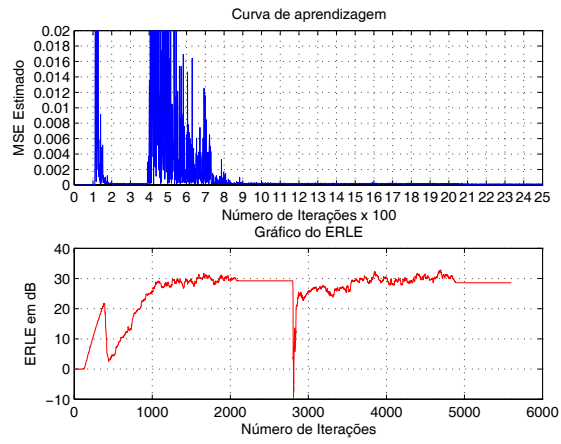


Figura B.47: Curva de aprendizagem e valor médio do ERLE.

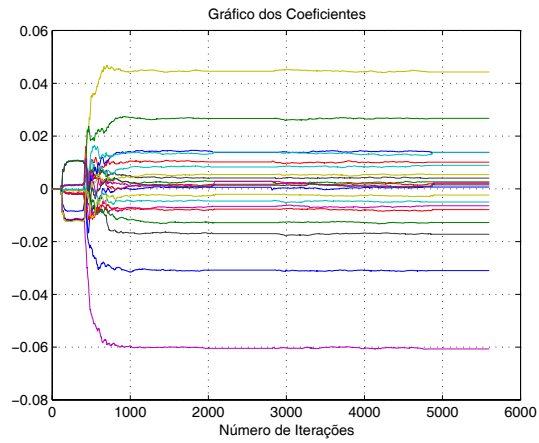


Figura B.48: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

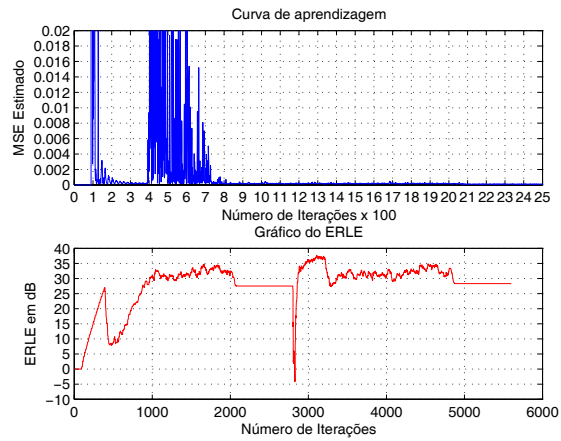


Figura B.49: Curva de aprendizagem e valor médio do ERLE.

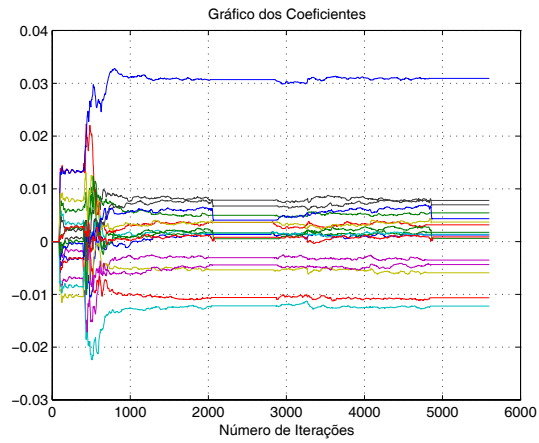


Figura B.50: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

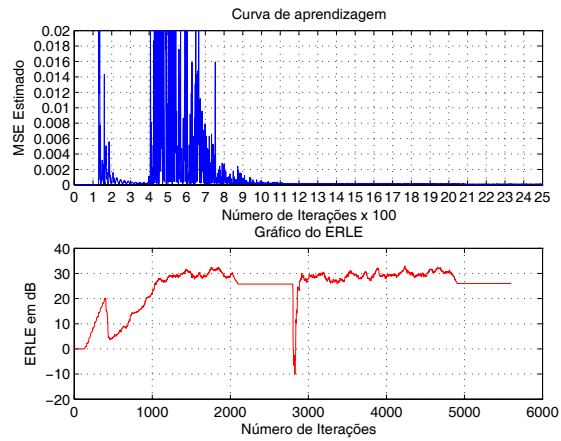


Figura B.51: Curva de aprendizagem e valor médio do ERLE.

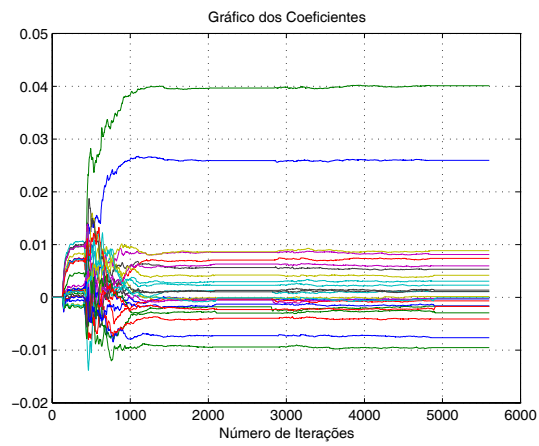


Figura B.52: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

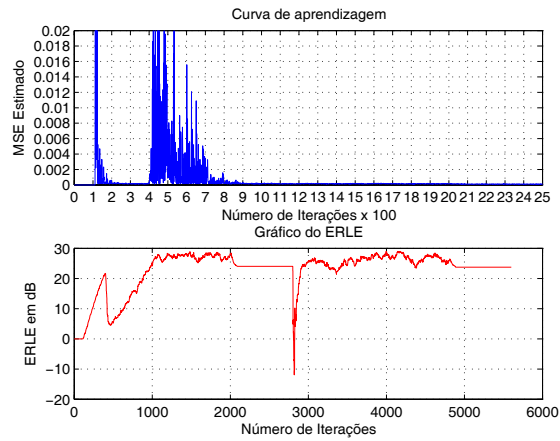


Figura B.53: Curva de aprendizagem e valor médio do ERLE.

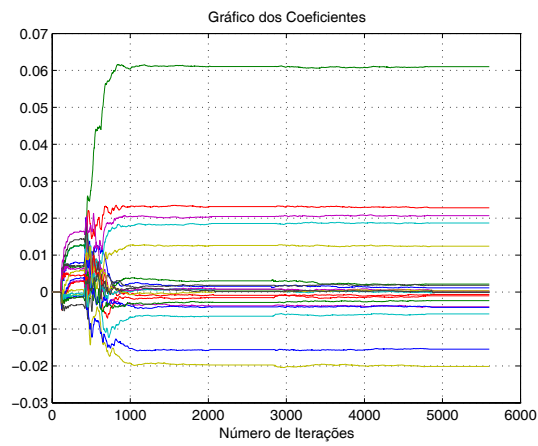


Figura B.54: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

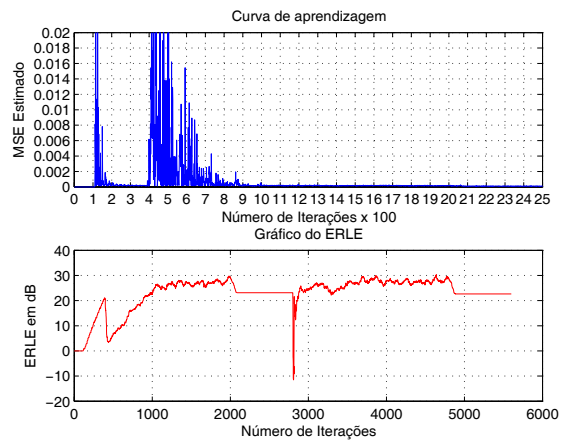


Figura B.55: Curva de aprendizagem e valor médio do ERLE.

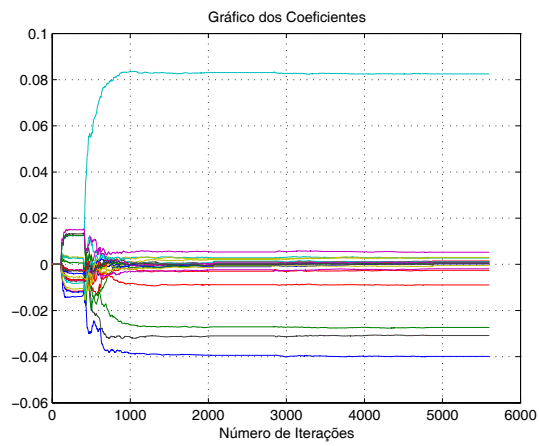


Figura B.56: Gráfico de convergência dos coeficientes.

5 - Gráficos de desempenho do algoritmo SM-BNDRLMS correspondentes aos modelos de canais 2 a 8 (G.168)
 Gráficos correspondentes ao modelo 2 de canal.

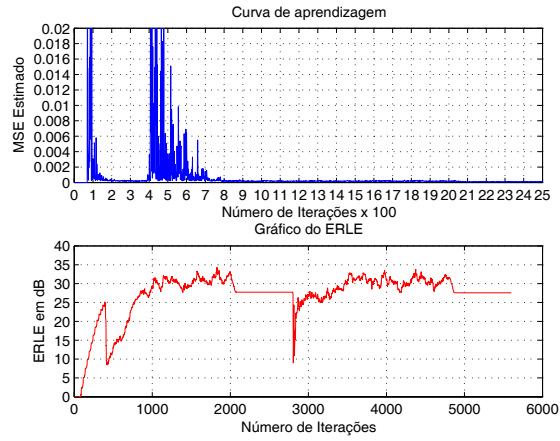


Figura B.57: Curva de aprendizagem e valor médio do ERLE.

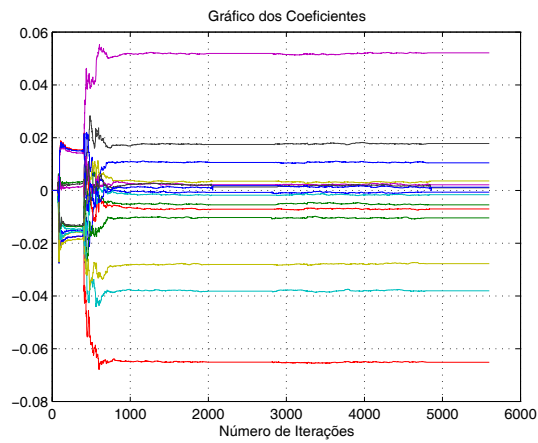


Figura B.58: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

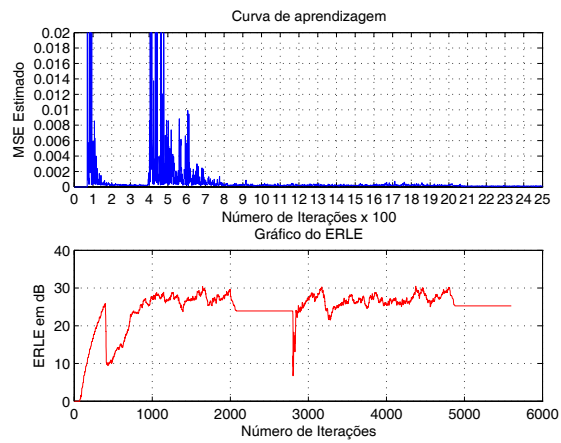


Figura B.59: Curva de aprendizagem e valor médio do ERLE.

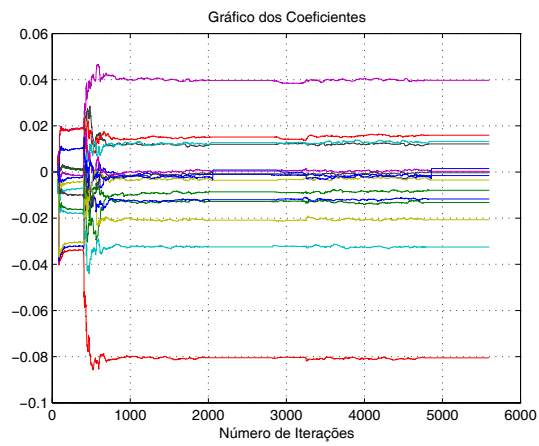


Figura B.60: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

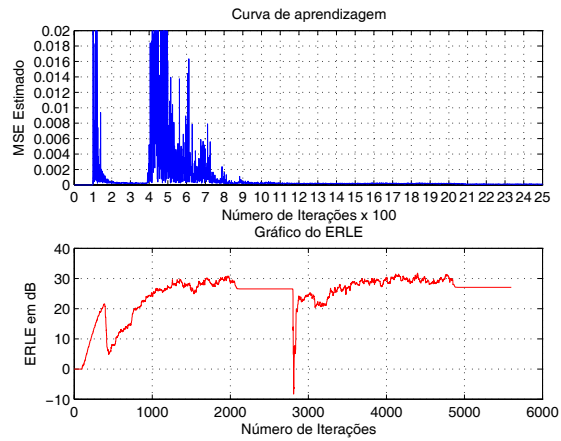


Figura B.61: Curva de aprendizagem e valor médio do ERLE.

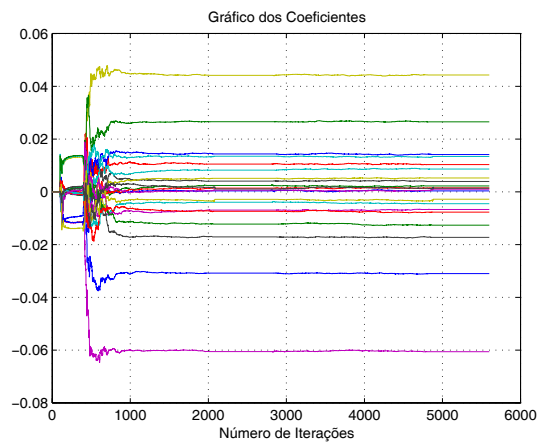


Figura B.62: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

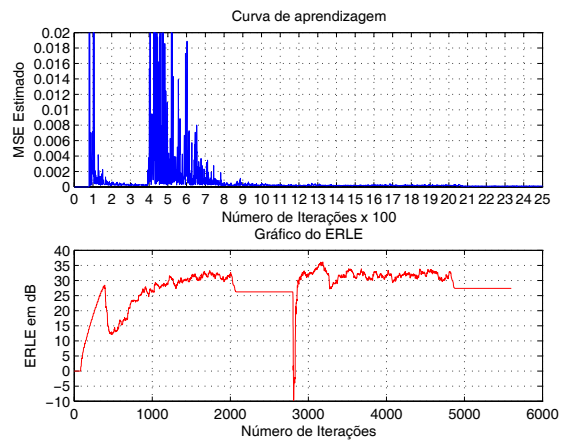


Figura B.63: Curva de aprendizagem e valor médio do ERLE.

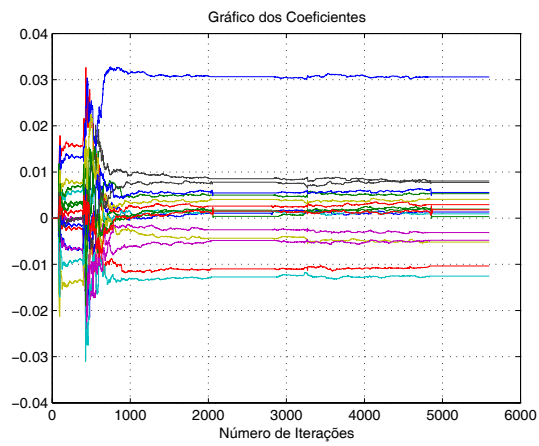


Figura B.64: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

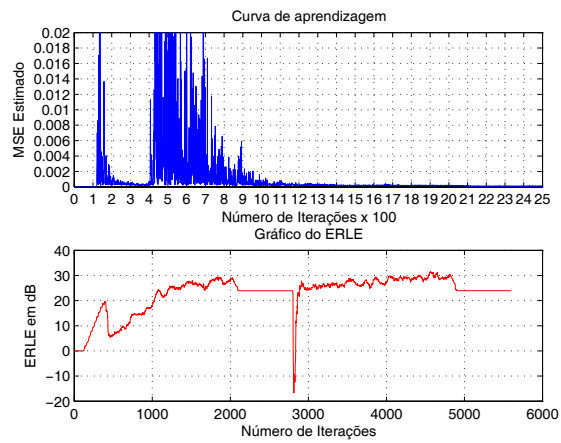


Figura B.65: Curva de aprendizagem e valor médio do ERLE.

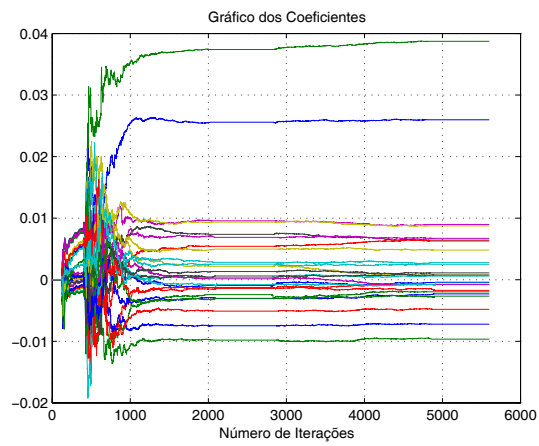


Figura B.66: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

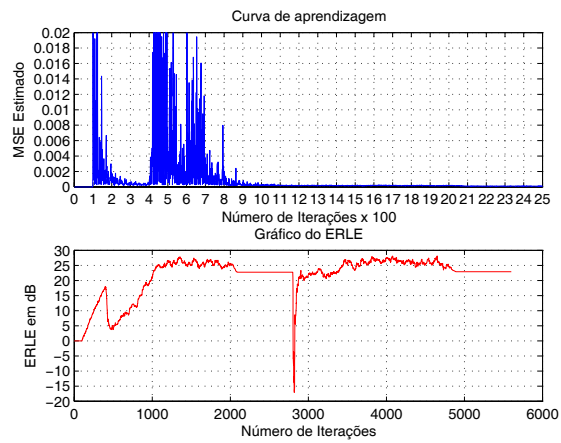


Figura B.67: Curva de aprendizagem e valor médio do ERLE.

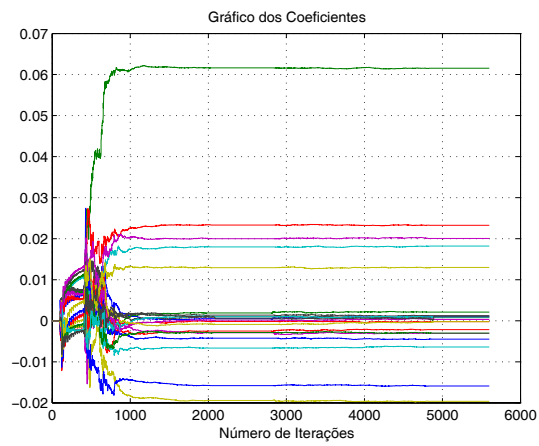


Figura B.68: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

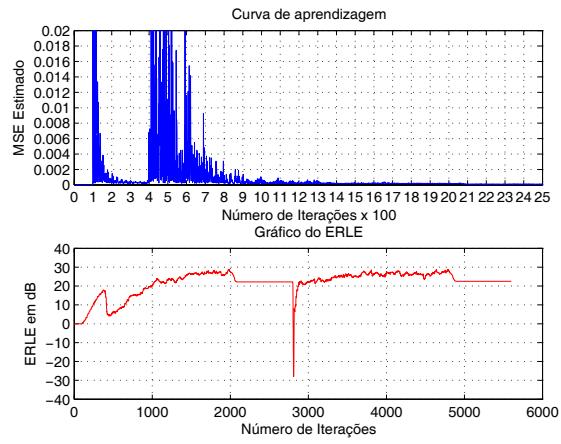


Figura B.69: Curva de aprendizagem e valor médio do ERLE.

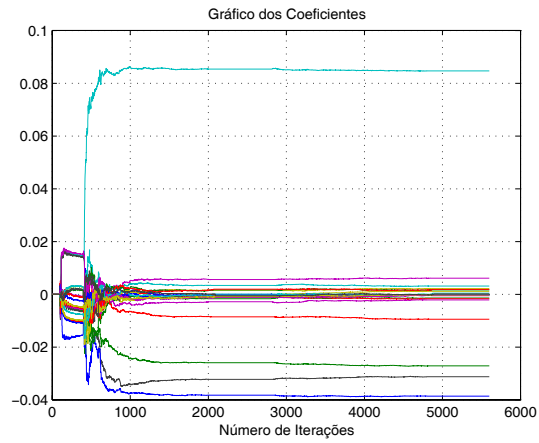


Figura B.70: Gráfico de convergência dos coeficientes.

6 - Gráficos de desempenho do algoritmo SM-AP correspondentes aos modelos de canais 2 a 8 (G.168)

Gráficos correspondentes ao modelo 2 de canal.

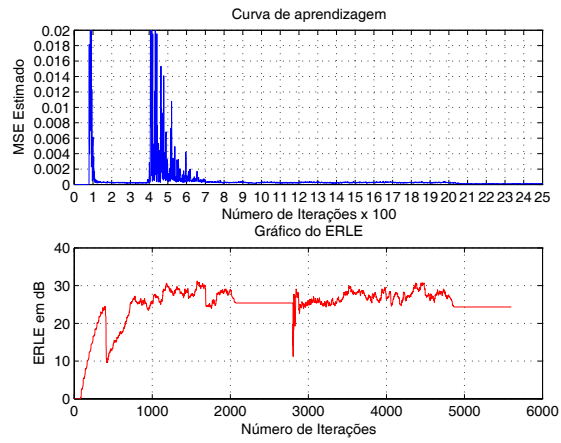


Figura B.71: Curva de aprendizagem e valor médio do ERLE.

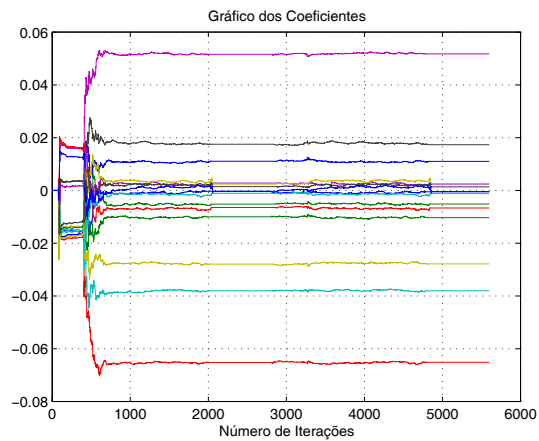


Figura B.72: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

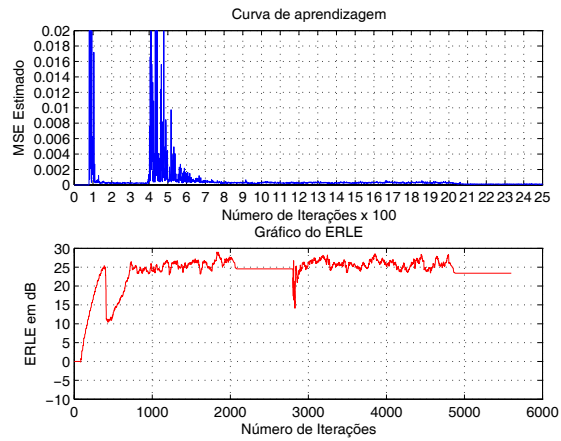


Figura B.73: Curva de aprendizagem e valor médio do ERLE.

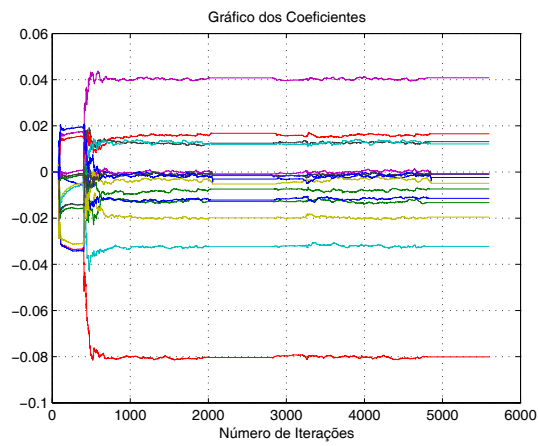


Figura B.74: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

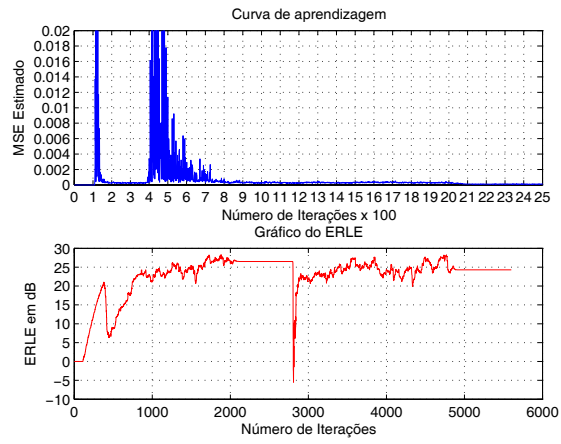


Figura B.75: Curva de aprendizagem e valor médio do ERLE.

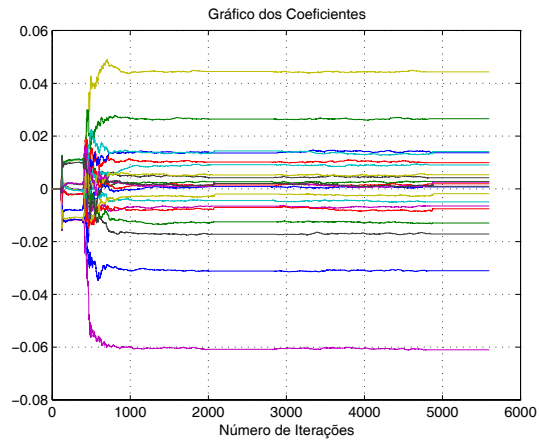


Figura B.76: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

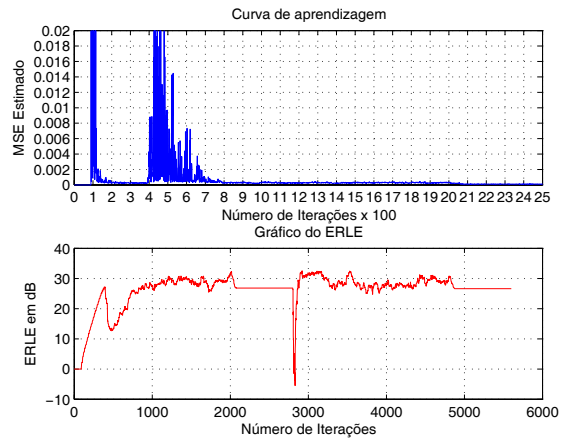


Figura B.77: Curva de aprendizagem e valor médio do ERLE.

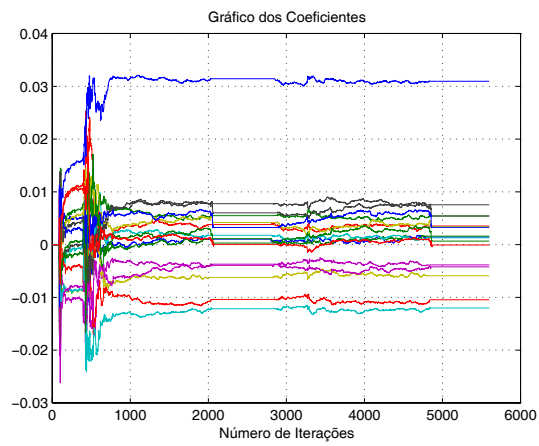


Figura B.78: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

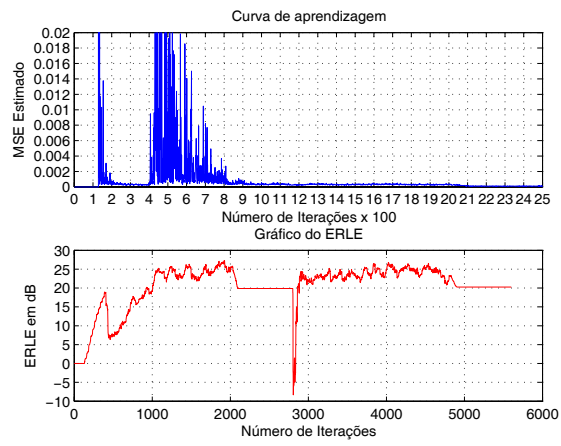


Figura B.79: Curva de aprendizagem e valor médio do ERLE.

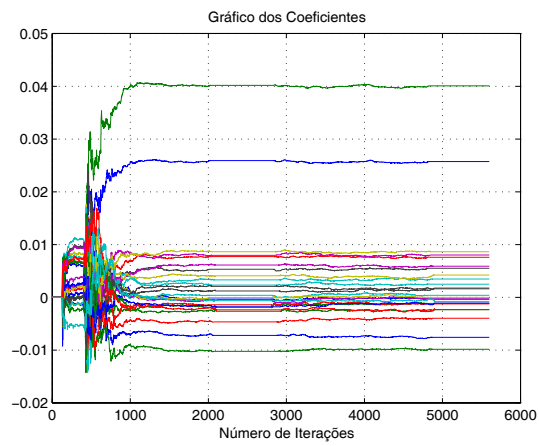


Figura B.80: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

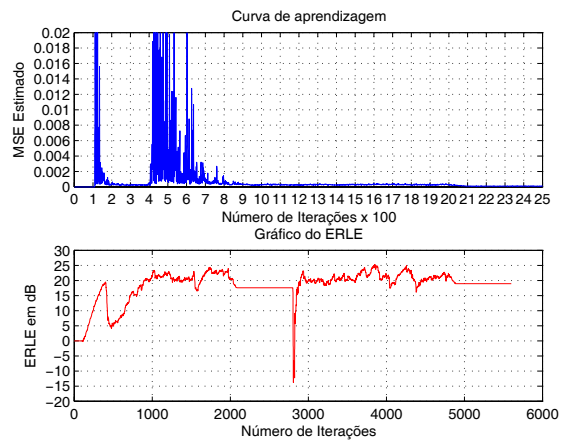


Figura B.81: Curva de aprendizagem e valor médio do ERLE.

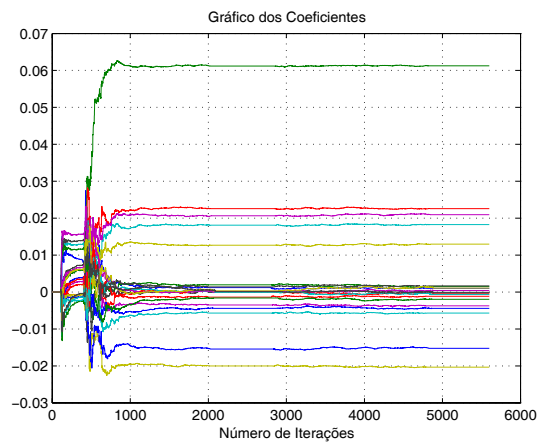


Figura B.82: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

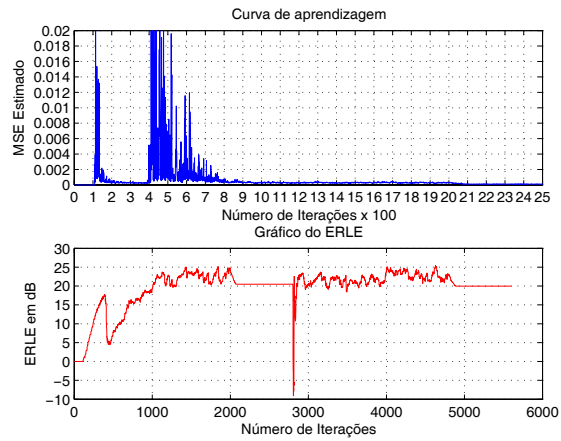


Figura B.83: Curva de aprendizagem e valor médio do ERLE.

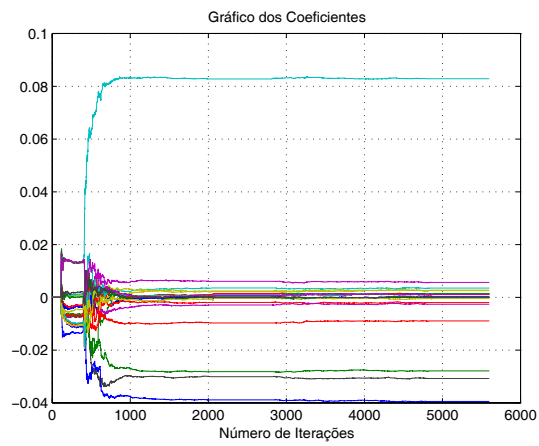


Figura B.84: Gráfico de convergência dos coeficientes.

7 - Gráficos de desempenho do algoritmo SM-AP Otimizado correspondentes aos modelos de canais 2 a 8 (G.168)
Gráficos correspondentes ao modelo 2 de canal.

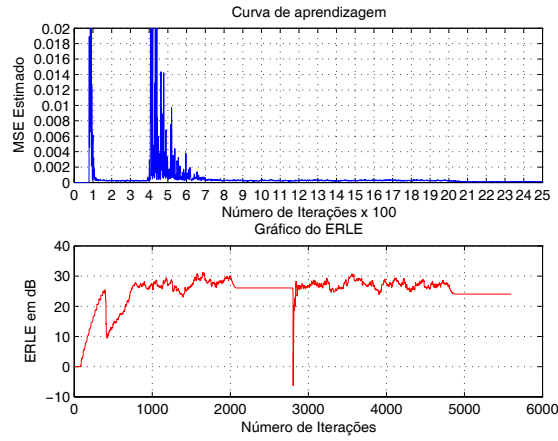


Figura B.85: Curva de aprendizagem e valor médio do ERLE.

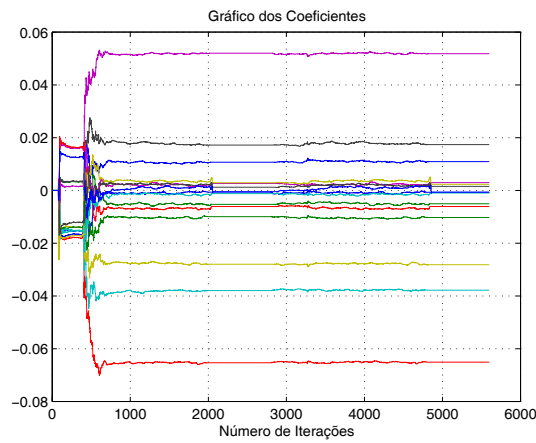


Figura B.86: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 3 de canal.

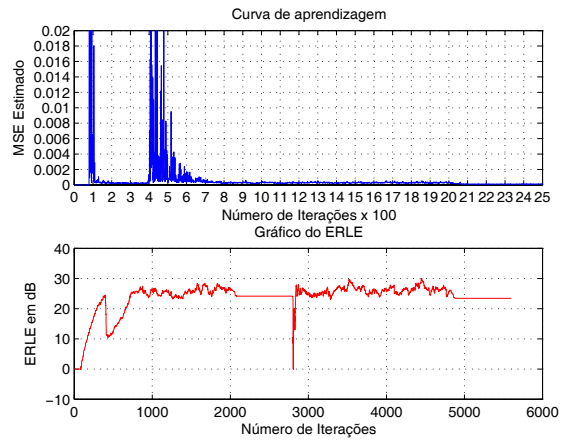


Figura B.87: Curva de aprendizagem e valor médio do ERLE.

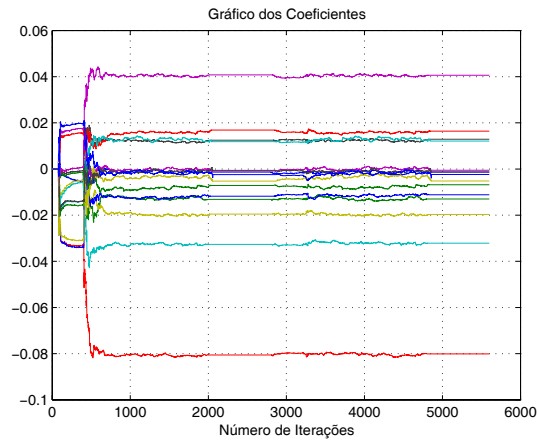


Figura B.88: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 4 de canal.

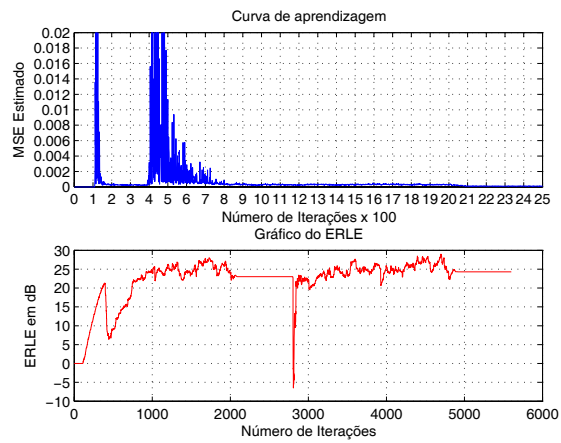


Figura B.89: Curva de aprendizagem e valor médio do ERLE.

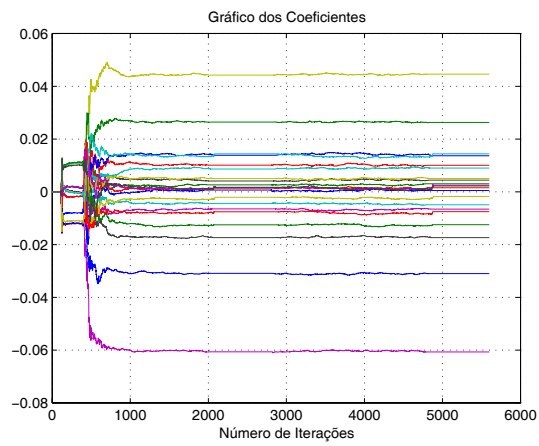


Figura B.90: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 5 de canal.

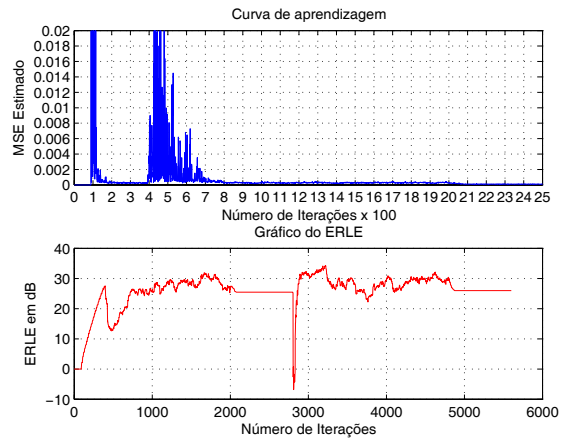


Figura B.91: Curva de aprendizagem e valor médio do ERLE.

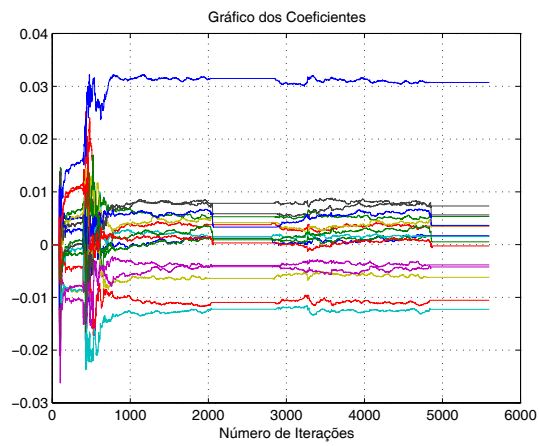


Figura B.92: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 6 de canal.

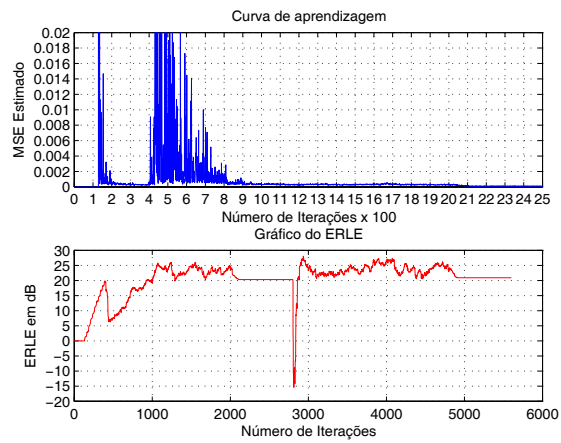


Figura B.93: Curva de aprendizagem e valor médio do ERLE.

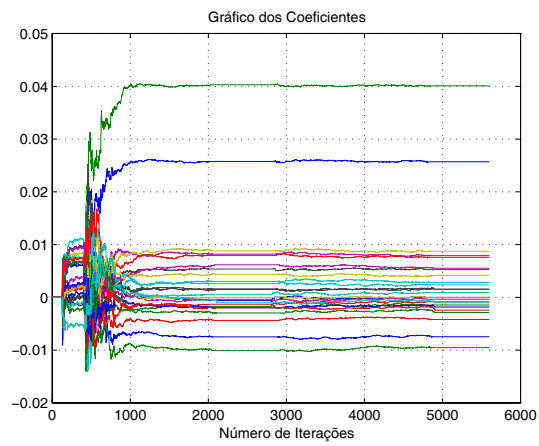


Figura B.94: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 7 de canal.

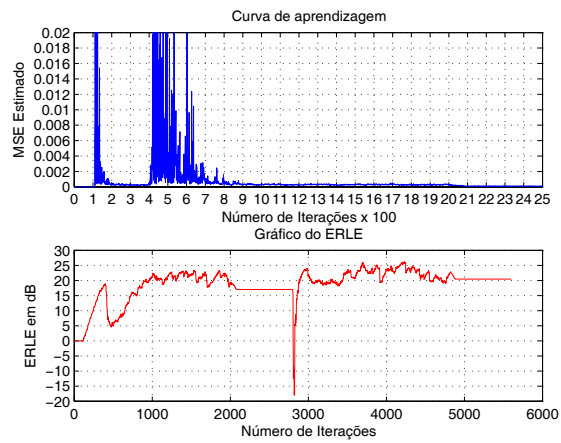


Figura B.95: Curva de aprendizagem e valor médio do ERLE.

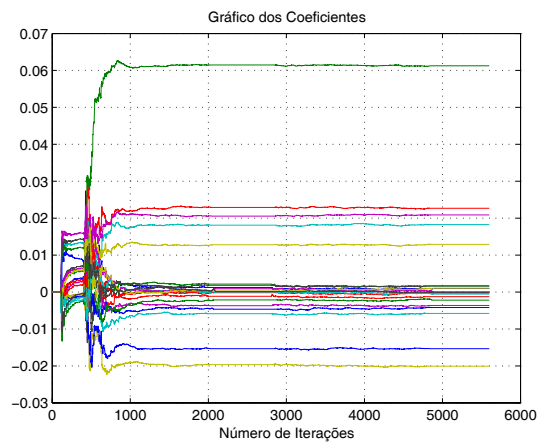


Figura B.96: Gráfico de convergência dos coeficientes.

Gráficos correspondentes ao modelo 8 de canal.

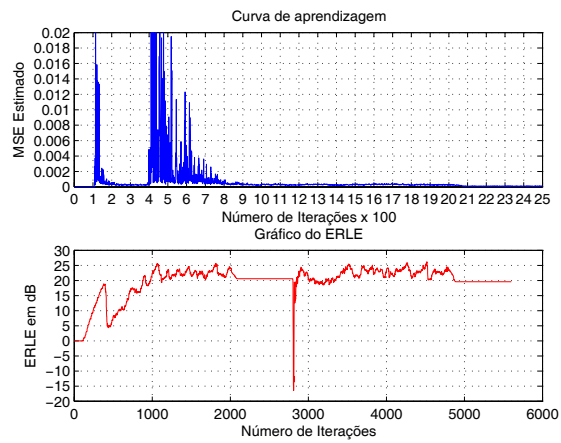


Figura B.97: Curva de aprendizagem e valor médio do ERLE.

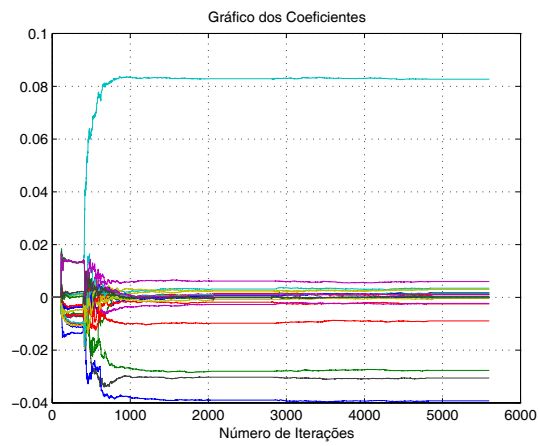


Figura B.98: Gráfico de convergência dos coeficientes.

Apêndice C

Programas em MATLAB utilizados nas simulações

A seguir estão listados os programas em MATLAB referentes aos algoritmos que foram testados neste trabalho.

```
1 - Cancelador de Eco LMS - G168
clear all;
load EchoPath1; %Modelo de Canal G.168
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
load css_8_single; %Sinal de teste G.168
An=css_8_single; %Sinal de entrada
N=100; %Numero de coeficientes do filtro
Mu=.007;
T=max(size(An));
% Realiza Nexp experimentos
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco=filter(wline_path,5,An');
    d=Eco+Nn;
    w=zeros(N,1);
    W=zeros(T,N);
    E=zeros(T,1);
```

```

for k=N:T,
    x=An(k:-1:k-N+1)';
    y=w'*x;
    e=d(k)-y;
    w=w+Mu*e*x;
    E(k,1)=e;
    W(k,:)=w';
end
Wm=Wm+W;
Em=Em+E;
end
Wm=Wm/Nexp;
Em=Em/Nexp;
Pd=filter(1,[1,-0.98],d.^2);
Pe=filter(1,[1,-0.98],Em.^2);
ERLE=10*log10(Pd./Pe);
figure(1);plot(Wm);grid
title('Gráfico dos Coeficientes');
xlabel('Número de Interações');
figure(2);
subplot(2,1,1),plot((abs(Em)), 'b');grid;hold on
title('Curva de aprendizagem');
ylabel('Saida estimada do erro');
subplot(2,1,2),plot(ERLE, 'r');grid;
title('Gráfico do ERLE');
ylabel('ERLE em dB');
xlabel('Número de Interações');

```

```

2 - Cancelador de Eco NLMS - G168

clear all;

load EchoPath1;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
load css_8_single;
An=css_8_single;
T=max(size(An));
N = 100; % Numero de coeficientes do filtro
Mu = 0.9;
alfa = 0.001;
% Realiza Nexp experimentos
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco = filter(wline_path,5,An');
    d=Eco+Nn;
    w = zeros(N,1);
    W = zeros(T,N);
    E = zeros(T,1);
    Y = zeros(T,1);
    for k=N:T,
        x=An(k:-1:k-N+1)';
        y=w'*x;
        e=d(k)-y;
        w = w + Mu*e*x/(alfa + x' * x);
        W(k,:)=w';
        E(k,1) = e;
        Y(k,1) = y;
    end
end

```



```

end
Wm=Wm+W;
Em=Em+E;
Ym=Ym+Y;
end
Wm=Wm/Nexp;
Em=Em/Nexp;
Pd=filter(1,[1,-0.98],d.^2);
Pe=filter(1,[1,-0.98],Em.^2);
ERLE=10*log10(Pd./Pe);
figure(1);plot(Wm);grid ;
title('Gráfico dos Coeficientes');
xlabel('Número de Interações');
figure(2);
subplot(2,1,1),plot((abs(Em)));grid;
title('Curva de aprendizagem');
ylabel('Saida estimada do erro');
subplot(2,1,2),plot(ERLE,'r');grid;
title('Gráfico do ERLE');
ylabel('ERLE em dB');
xlabel('Número de Interações');

```

```

3 - Cancelador de Eco RLS - G168

clear all;

load EchoPath1;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
load css_8_single;
An=css_8_single;
T=max(size(An));
N=100;
lambda=1.0;
delta=1.0;
% Realiza Nexp experimentos
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco = filter(wline_path,5,An');
    d = Eco+Nn;
    w = zeros(N,1);
    W=zeros(T,N);
    E=zeros(T,1);
    Y=zeros(T,1);
    R_1 = eye(N,N)/delta;
    for k=N:T,
        x=An(k:-1:k-N+1)';
        K_k = R_1*x;
        K_linha = K_k/(lambda+x'*K_k);
        R_1 = (R_1 - (K_k*K_k'))/(lambda+x'*K_k)/lambda;
        y = w'*x;
        e = d(k) - y;
    end
end

```

```

    w = w + e*K_linha;

    W(k,:)=w';

    E(k,1) = e;

    Y(k,1) = y;

end

Wm=Wm+W;

Em=Em+E;

Ym=Ym+Y;

end

Wm=Wm/Nexp;

Em=Em/Nexp;

Pd=filter(1,[1,-0.98],d.^2);

Pe=filter(1,[1,-0.98],Em.^2);

ERLE=10*log10(Pd./Pe);

figure(1);plot(Wm);grid;

title('Gráfico dos Coeficientes');

xlabel('Número de Iterações');

figure(2);

subplot(2,1,1),plot((abs(Em)));grid;

title('Curva de aprendizagem');

ylabel('Saida estimada do erro');

subplot(2,1,2),plot(ERLE,'r');grid;

title('Gráfico do ERLE');

ylabel('ERLE em dB');

xlabel('Número de Iterações');

```

4 - Cancelador de Eco AP (Affine-Projection) - G168

```
clear all;
load EchoPath1;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
load css_8_single;
An=css_8_single;
T=max(size(An));
N = 100;
Mu = 0.1;
L = 20;
delta = 1.0;
% Realiza Nexp experimentos
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco=filter(wline_path,5,An');
    d=Eco+Nn;
    w = zeros(N,1);
    W = zeros(T,N);
    E = zeros(T,1);
    Y = zeros(T,1);
    Xap = zeros(N,L+1);
    dap = zeros(L+1,1);
    for k=L+1:L+N,
        NewLine = An(k:-1:k-L);
        Xap = [NewLine;Xap(1:N-1,:)];
    end
    for k=L+N+1:T,
```

```

NewLine = An(k:-1:k-L);
Xap = [NewLine;Xap(1:N-1,:)];
R=R+Xap*Xap';
dap = d(k:-1:k-L);
y = Xap'*w;
eap = dap - y;
t = inv(Xap'*Xap +delta*eye(size(Xap'*Xap)))*eap;
w = w + Mu * Xap * t;
W(k,:) = w';
E(k,1) = eap(1);
Y(k,1) = y(1);

end

Wm=Wm+W;
Em=Em+E;
Ym=Ym+Y;

end

Wm=Wm/Nexp;
Em=Em/Nexp;
Pd=filter(1,[1,-0.98],d.^2);
Pe=filter(1,[1,-0.98],Em.^2);
ERLE=10*log10(Pd./Pe);
figure(1);plot(Wm);grid;
title('Gráfico dos Coeficientes');
xlabel('Número de Interações');
figure(2);
subplot(2,1,1),plot((abs(Em)));grid;
title('Curva de aprendizagem');
ylabel('Saida estimada do erro');
subplot(2,1,2),plot(ERLE,'r');grid;
title('Gráfico do ERLE');
ylabel('ERLE em dB');
xlabel('Número de Interações');

```

```

clear all;
load EchoPath1;
load css_8_single;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
An=css_8_single';
T=max(size(An));
N = 100;
threshold = 1e-5;
delta1 = 1.0;
delta2 = 0.1;
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco=filter(wline_path,5,An);
    Dn=Eco+Nn;
    w = zeros(N,1);
    W = zeros(T,N);
    W2=zeros(T,N);
    E = zeros(T,1);
    Y = zeros(T,1);
    Xk_1 = zeros(N,1);
    Dk_1 = 0;
    for m=N:T,
        x=An(m:-1:m-N+1);
        eco_est = w'*x;
        e = Dn(m) - eco_est;
        if abs(e) > threshold

```

```

alpha_k = (1-threshold./abs(e));
a = x'*x+delta1;
w_linha = w + (alpha_k*e*x)/a;
Ek_1 = Dk_1 - Xk_1'*w_linha;
if abs(Ek_1) > threshold
    b = Xk_1'*Xk_1+delta1;
    c = x'*Xk_1+delta2;
    beta_k = 1 - threshold/abs(Ek_1);
    den = a*b - c^2;
    lambda1 = -Ek_1*c/den;
    lambda2 = Ek_1*a/den;
    w = w_linha + beta_k*(lambda1*x + lambda2*Xk_1);
    W2(m,:)=w';
end
end
W(m,:)=w';
E(m,1) = e(1);
Y(m,1) = eco_est(1);
Xk_1 = x;
Dk_1 = Dn(m);
end
Wm=Wm+W;
Em=Em+E;
Ym=Ym+Y;
end
Wm=Wm/Nexp;
Em=Em/Nexp;
Pd=filter(1,[1,-0.98],Dn.^2);
Pe=filter(1,[1,-0.98],Em.^2);
ERLE=10*log10(Pd./Pe);
figure(1);plot(Wm);grid;
title('Gráfico dos Coeficientes');

```

```
xlabel('Número de Iterações');  
figure(2);  
subplot(2,1,1),plot((abs(Em)));grid;  
title('Curva de aprendizagem');  
ylabel('Saida estimada do erro');  
subplot(2,1,2),plot(ERLE,'r');grid;  
title('Gráfico do ERLE');  
ylabel('ERLE em dB');  
xlabel('Número de Iterações');
```


6 - Cancelador de Eco SM-AP (Set-Membership Affine-Projection) - G168

```

clear all;
load EchoPath1;
load css_8_single;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
An=css_8_single;
T=max(size(An));
N = 100;
P = 20;
threshold = 1e-5;
delta = 1.0;
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
for p=1:Nexp,
    Nn=.01*randn(T,1);
    Eco=filter(wline_path,5,An');
    d=Eco+Nn;
    w = zeros(N,1);
    W = zeros(T,N);
    E = zeros(T,1);
    Y = zeros(T,1);
    Xap = zeros(N,P+1);
    dap = zeros(P+1,1);
    for k=P+1:P+N
        NewLine = An(k:-1:k-P);
        Xap = [NewLine;Xap(1:N-1,:)];
    end;
    for k=P+N+1:T,
        NewLine = An(k:-1:k-P);

```

```

Xap = [NewLine;Xap(1:N-1,:)];
x=An(k:-1:k-N+1)';
R=R+x*x';
dk = d(k);
y = x'*w;
e = dk - y;
if abs(e) > threshold
    alpha_k = (1-threshold./abs(e))*eye(P+1,1);
    e_ = alpha_k.*e;
    t = inv(Xap'*Xap + delta*eye(size(Xap'*Xap)))*e_;
    w = w + Xap*t;
end
W(k,:)=w';
E(k,1) = e(1);
Y(k,1) = y(1);
end
Wm=Wm+W;
Em=Em+E;
Ym=Ym+Y;
end
Wm=Wm/Nexp;
Em=Em/Nexp;
Pd=filter(1,[1,-0.98],d.^2);
Pe=filter(1,[1,-0.98],Em.^2);
ERLE=10*log10(Pd./Pe);
figure(1);plot(Wm);grid;
title('Gráfico dos Coeficientes');
xlabel('Número de Iterações');
figure(2);
subplot(2,1,1),plot((abs(Em)));grid;
title('Curva de aprendizagem');
ylabel('Saida estimada do erro');

```

```
subplot(2,1,2),plot(ERLE,'r');grid;  
title('Gráfico do ERLE');  
ylabel('ERLE em dB');  
xlabel('Número de Iterações');
```

7 - Cancelador de Eco SM-AP (Set-Membership Affine-Projection) Otimizado - G168

```

clear all;
load EchoPath1;
load css_8_single;
wline_path=line_path1/norm(line_path1);
n=length(wline_path);
An = css_8_single;
T=max(size(An));
N = 70;
P = 10;
gama_1 = 1.0e-3;
gama_2 = 1.0e-5;
delta = 1.0;
mu=0.5;
Nexp=100;
Wm=zeros(T,N);
Em=zeros(T,1);
Ym=zeros(T,1);
MSE=zeros(T,1);
for p=1:Nexp,
    Nn=.001*randn(T,1);
    Eco=filter(wline_path,5,An);
    d = Eco+Nn;
    w = zeros(N,1);
    W = zeros(T,N);
    E = zeros(T,1);
    Y = zeros(T,1);
    Xap = zeros(N,P+1);
    for k=P+1:P+N-1,
        Xap = [An(k:-1:k-P);Xap(1:N-1,:)];
    end;
    for k=P+N:T,

```

```

Xap = [An(k:-1:k-P);Xap(1:N-1,:)];
x=An(k:-1:k-N+1)';
dk = d(k);
y = x.'*w;
ek = dk - y;
if (abs(ek) > gama_1)|(abs(ek)<gama_2)
    if abs(ek) > gama_1
        gama_k = (1-gama_1./abs(ek))*eye(P+1,1);
    end
    if abs(ek) < gama_2
        gama_k = (1-gama_2./abs(ek))*eye(P+1,1);
    end
    e_=gama_k.*ek;
    t = inv(Xap'*Xap + delta*eye(size(Xap'*Xap)))*e_;
    w = w + mu*Xap*t;
end
MSE(k) = MSE(k)+ek^2;
W(k,:) = w';
E(k,1) = ek;
Y(k,1) = y;
end
Wm=Wm+W;
Em=Em+E;
Ym=Ym+Y;
end
MSE=MSE/Nexp;
Wm=Wm/Nexp;
Em=Em/Nexp;
P_Eco=filter(1,[1,-0.99],Eco.^2);
P_Residual=filter(1,[1,-0.99],(Eco-Y').^2);
ERLE=10*log10(P_Eco./P_Residual);
figure(1);subplot(2,1,1),plot(Eco,'b');grid;

```

```

subplot(2,1,2),plot(Y,'r');grid;
figure(2);subplot(2,1,1),plot(Nn,'b');grid;
subplot(2,1,2),plot(Em,'b');grid
axis([0 6000 -0.02 0.02])
figure(3);plot(Wm);grid;
title('Gráfico dos Coeficientes');
xlabel('Número de Iterações');
figure(4);subplot(2,1,1),plot(MSE);grid;hold on
title('Curva de aprendizagem');
ylabel('Saida estimada do erro');
subplot(2,1,2),plot(ERLE,'r');grid;
title('Gráfico do ERLE');
ylabel('ERLE em dB');
xlabel('Número de Iterações');

```