



UNIVERSIDADE ESTADUAL DO MARANHÃO - UEMA  
CENTRO DE CIÊNCIAS E TECNOLOGIAS  
MESTRADO EM ENGENHARIA DA COMPUTAÇÃO E SISTEMAS

*Estudo e desenvolvimento de uma família de algoritmos  
não lineares para filtragem adaptativa*

**ELIZANGELA NUNES NEVES CERQUEIRA**

Orientador: Dr Ewaldo Eder Carvalho Santana  
Universidade Estadual do Maranhão - UEMA

Coorientador: Dr Lucio Flávio A. Campos  
Universidade Estadual do Maranhão - UEMA

SÃO LUÍS  
JUNHO DE 2016

ELIZANGELA NUNES NEVES CERQUEIRA

*Estudo e desenvolvimento de uma família de algoritmos  
não lineares para filtragem adaptativa*

Dissertação apresentada ao Programa de Pós Graduação  
em Engenharia da Computação e Sistemas, como  
requisito parcial para obtenção do título de Mestre em  
Engenharia da Computação e Sistemas.

Orientador: Dr Ewaldo Eder Carvalho Santana  
Universidade Estadual do Maranhão -  
UEMA

Coorientador: Dr Lucio Flávio A. Campos  
Universidade Estadual do Maranhão -  
UEMA

SÃO LUÍS  
JUNHO DE 2016

ELIZANGELA NUNES NEVES CERQUEIRA

*Estudo e desenvolvimento de uma família de algoritmos  
não lineares para filtragem adaptativa*

Dissertação apresentada ao Programa de Pós Graduação  
em Engenharia da Computação e Sistemas, como  
requisito parcial para obtenção do título de Mestre em  
Engenharia da Computação e Sistemas.

Trabalho aprovado. São Luís, 21 de Junho de 2016

---

**Dr Ewaldo Eder Carvalho Santana**  
Orientador

---

**Dr Lucio Flávio A. Campos**  
Co-orientador

---

**Dra. Cristiane Cristina Sousa Silva**  
Universidade Federal do Maranhão -  
UFMA

---

**Dra. Áurea Celeste da Costa Ribeiro**  
Universidade Estadual do Maranhão -  
UEMA

SÃO LUÍS  
JUNHO DE 2016

Aos meus pais, João Francisco (*in memoriam*)  
e Maria Erundina, e ao meu esposo Sidney  
Cerqueira. Meu muito obrigado!!

## Agradecimentos

Inicio meus agradecimentos por DEUS, por ter colocado pessoas tão especiais a meu lado, sem as quais certamente não teria dado conta.

Ao Prof. Dr. Ewaldo Eder , meu orientador e exemplo profissional, por não ter permitido que eu interrompesse o processo, pela confiança e principalmente pela paciência.

Ao Prof. Dr. Lucio Flávio A. Campos, meu coorientador, pelo apoio e confiança. Obrigada!!!

Aos meus pais, meu infinito agradecimento. Obrigada por sempre acreditaram em minha capacidade, por me terem dado educação, valores e incentivo sempre, renunciando muitas vezes aos seus sonhos para que eu pudesse realizar o meu, partilho a alegria deste momento. Pai (in memoriam) e mãe, amor eterno e incondicional, vocês são a grande razão de todas as minhas conquistas e de todas as outras que irei alcançar, pois seus ensinamentos são a base de que preciso.

A meu querido esposo, Sidney, por ser tão importante na minha vida. Meu porto seguro, sempre ao meu lado, me fazendo acreditar que posso mais que imagino. Graças a seu companheirismo, amizade, paciência, compreensão, apoio, alegria e amor, este trabalho pôde ser concretizado. Obrigada por ter feito do meu sonho o nosso sonho!

A meus filhos, João Guilherme e Larissa, vocês são minha inspiração para ser sempre melhor. Amo vocês!!!

A todos os meus familiares, irmãos, cunhados e sobrinhos. Não citarei todos os nomes, para não me esquecer de ninguém. Mas há aquelas pessoas especiais que diretamente me incentivaram. A meus irmãos, Maria, Dulcilene, Elivânia, Cristiane, Givanildo e João meu agradecimento especial, pois a seu modo, sempre se orgulharam e confiaram em mim. Em especial a minha irmã Maria Raimunda e meu cunhado Joaquim, por estarem sempre presentes e dispostos a ajudar quando precisei, cuidando tão bem do meu pequeno príncipe Guilherme na minha ausência, vocês foram essenciais para essa conquista.

A meus amigos do mestrado, especialmente à Jardiel, Linhares, Marcio e Idovaldo, vocês se mostraram verdadeiros amigos. Obrigada por dividirem comigo as angústias e alegrias, sempre me incentivando a não desistir.

A todos os professores do Programa de Pós-Graduação em Engenharia da Computação e Sistema, pelo incentivo, apoio e voto de confiança. Meus sinceros agradecimentos.

Todos vocês me proporcionaram mais que a busca de conhecimento técnico e científico, mas uma LIÇÃO DE VIDA. Ninguém vence sozinho... OBRIGADA!!!

*“Desistir... Eu já pensei seriamente nisso, mas nunca me levei realmente a sério. É que tem mais chão nos meus olhos do que cansaço nas minhas pernas, mais esperança nos meus passos do que tristeza nos meus ombros, mais estrada no meu coração do que medo na minha cabeça.” (Cora Coralina)*

## Resumo

Neste trabalho objetiva-se desenvolver uma família de algoritmos adaptativos baseados em funções não lineares, utilizando-se a função  $1 - e^{-\alpha\epsilon^2}$  como critério aplicado sobre o erro, o qual deseja-se minimizar. Essa função gera uma superfície que proporciona convergência mais rápida do algoritmo e menor desajuste quando comparado ao LMS. O desenvolvimento de tal algoritmo baseia-se na utilização de estatísticas de alta ordem para obtenção de mais informações sobre os sinais envolvidos no processo, com o objetivo de melhorar a performance de um filtro adaptativo. Simulações utilizando o Matlab são apresentadas para validar a teoria apresentada, e também o aplicamos na eliminação de ruído em sinal de eletrocardiograma. Os resultados mostraram que o algoritmo teve um desempenho melhor do que o LMS quando o ruído tinha uma distribuição Gaussiana.

**Palavras-chave:** filtragem adaptativa, cancelamento de ruído, LMS, Eletrocardiograma.

## Abstract

This work aimed to develop a family of adaptive algorithms based on non-linear functions, using the function  $1 - e^{-\alpha\epsilon^2}$  as applied criteria about the error, which it wishes minimizar. Essa function generates a surface that provides faster convergence of the algorithm and smaller misfit compared to the LMS. The development of such an algorithm is based on the use of higher order statistics to obtain more information about the signals involved in the process, in order to improve the performace of an adaptive filter. Simulations using Matlab are presented to validate the theory presented, and also apply in noise elimination in ECG signal. The results showed that the algorithm has a better performance than the LMS when the noise had a Gaussian distribution.

**Keywords:** Adaptive filtering, noise cancellation, LMS, Electrocardiogram.

## Lista de Figuras

Figura 1 – Classificação de filtros digitais . . . . .	5
Figura 2 – Estrutura geral de um filtro de Wiener . . . . .	7
Figura 3 – Estrutura de um filtro FIR transversal . . . . .	7
Figura 4 – Porção de uma superfície quadrática tridimensional . . . . .	9
Figura 5 – Elementos de um filtro adaptativo . . . . .	11
Figura 6 – Estrutura geral de um filtro adaptativo . . . . .	12
Figura 7 – Estrutura de um filtro adaptativo para predição linear . . . . .	13
Figura 8 – Estrutura de um filtro adaptativo para identificação de sistemas . . .	14
Figura 9 – Estrutura de um filtro adaptativo para modelagem inversa . . . . .	14
Figura 10 – Estrutura de um filtro adaptativo para cancelamento de ruídos . . .	15
Figura 11 – Superfície de desempenho do algoritmo LMS e processo de convergência . . . . .	19
Figura 12 – Excesso no erro final em relação ao erro mínimo . . . . .	20
Figura 13 – Gráfico da superfície gerada pela função $1 - e^{-2\varepsilon^2}$ e algumas curvas de níveis. Os pesos $w_0$ e $w_1$ variam de -1 a 1. . . . .	22
Figura 14 – Gráficos das funções $1 - e^{-\varepsilon^2}$ , $1 - e^{-2\varepsilon^2}$ e $1 - e^{-3\varepsilon^2}$ . . . . .	23
Figura 15 – Gráficos das funções $1 - e^{-2\varepsilon^2}$ e $\varepsilon^2$ , onde podemos ver a maior inclinação da primeira, no intervalo [-1,1] . . . . .	24
Figura 16 – Diagrama de blocos do filtro com um vetor de entrada $\mathbf{X}$ , um sinal desejado $d$ , composto por um sinal $s$ e um ruído $n$ e o erro $\varepsilon$ . . . . .	25
Figura 17 – Curva de aprendizagem de um algoritmo adaptativo . . . . .	30
Figura 18 – Curvas de aprendizagem dos algoritmos LMS e SAG com $\alpha = 2$ . . .	31
Figura 19 – Curvas de aprendizagem do algoritmo SAG com $\alpha = 1$ , $\alpha = 2$ , $\alpha = 3$ , $\alpha = 4$ e $\alpha = 5$ . . . . .	31
Figura 20 – Curvas de aprendizagem dos algoritmos LMS e SAG com $\alpha = 4$ . . .	32
Figura 21 – Curvas de aprendizagem dos algoritmos LMS e SAG com $\alpha = 10$ . .	32
Figura 22 – Aquisição de sinal de ECG . . . . .	33
Figura 23 – Diagrama de blocos de um cancelador de ruído adaptativo em sinal de ECG com o algoritmo SAG . . . . .	34
Figura 24 – Fluxograma da algoritmo SAG . . . . .	35
Figura 25 – Sinal de ECG e sinal de ECG com ruído . . . . .	36
Figura 26 – Resultado do processo de filtragem no sinal de ECG com ruído pelos algoritmos LMS e SAG. . . . .	37
Figura 27 – Resultado do processo de filtragem no sinal de ECG pelo algoritmo SAG. . . . .	38

## Lista de Tabelas

Tabela 1 – Resumo do Algoritmo LMS . . . . .	18
Tabela 2 – Resumo do Algoritmo SAG . . . . .	25

## Lista de Abreviaturas e Siglas

ECG	Eletrocardiograma
LMS	Least-Mean-Square
RLS	Recursive-Least-Square
NLMS	Normalized Least-Mean-Square
SOS	Second Order Statistics
HOS	Higher Order Statistics
SA	Sigmoidal Algoritmo
RNL	Recursivo Não Linear
MATLAB	MATrix LABoratory
FIR	Finite Impulse Response
IIR	Infinite Impulse Response
EMQ	Erro Médio Quadrático
EMG	Eletromiograma
ANC	Active Noise Cancellor

# Sumário

<b>1 – Introdução</b>	<b>1</b>
1.1 Estado da arte	1
1.2 Objetivos	3
1.3 Organização do texto	3
<b>2 – Filtros digitais</b>	<b>5</b>
2.1 Filtros fixos <i>versus</i> Filtros adaptativos	5
2.2 Filtro de Wiener	6
2.2.1 Solução de Wiener para Filtros Transversais	7
2.3 Filtro adaptativo	10
2.4 Estrutura de um filtro adaptativo	12
2.5 Aplicações	13
2.6 Algoritmos adaptativos	16
2.6.1 Algoritmo da descida mais íngreme	16
2.6.2 Algoritmo Least-Mean-Square (LMS)	17
2.6.2.1 Derivação do algoritmo LMS	17
2.6.2.2 Convergência do LMS	18
2.6.2.3 Excesso do Erro Médio Quadrático	19
<b>3 – Algoritmo proposto</b>	<b>21</b>
3.1 Superfície de desempenho da função	21
3.2 Método	24
3.3 Comportamento e convergência do vetor peso	25
3.4 Desajuste	28
<b>4 – Simulações e resultados</b>	<b>29</b>
4.1 Análise do desempenho de algoritmos adaptativos	29
4.1.1 Simulações computacionais SAG $\times$ LMS	30
4.2 Eliminação de ruído em sinal de Eletrocardiograma (ECG)	32
4.3 Eliminação de ruído Gaussiano em sinal de ECG por meio do algoritmo SAG	34
4.3.1 Resultados ECG	35
<b>5 – Conclusão e Trabalhos futuros</b>	<b>39</b>
5.1 Conclusões	39
5.2 Trabalhos futuros	39

**Referências . . . . . 41**

# 1 Introdução

## 1.1 Estado da arte

A filtragem adaptativa constitui atualmente uma ferramenta de grande relevância no processamento de sinais, sendo largamente utilizada em diversas aplicações práticas que não podem ser resolvidas através de filtros fixos, especialmente quando precisa-se processar sinais provenientes de ambientes com estatísticas desconhecidas ou que variam no tempo (não estacionário), pois tais filtros possuem a capacidade de adaptar-se automaticamente as alterações no sinal de entrada.

Essa técnica possui um vasto campo de aplicações, sendo explorada com sucesso em problemas de economia, engenharia biomédica, equalização de canais, sistemas de controle e em telecomunicações (WIDROW; STEARNS, 1985). Na área biomédica podem ser encontradas diversas aplicações, dentre as quais pode-se citar: cancelamento de interferências do coração doador no eletrocardiograma (ECG) durante o transplante de coração; cancelamento da influência materna em ECG fetal (SANTANA et al., 2006). Em (HAYKIN, 2001) são apresentadas diversas aplicações do processamento adaptativo de sinais.

A eficiência dos filtros adaptativos depende principalmente do projeto usado e do algoritmo de adaptação. O principal desafio da filtragem adaptativa é desenvolver algoritmos que sejam, simultaneamente, fáceis de implementar, robustos, rápidos, que rejeitem bem o ruído, e que acompanhem eficientemente processos não estacionários de forma contínua.

Diversos algoritmos para a atualização dos parâmetros do filtro adaptativo foram desenvolvidos nos últimos anos. Dentre os algoritmos existentes na literatura podemos citar o algoritmo LMS (Least-Mean-Square) convencional, Normalized LMS (NLMS), o LMS no domínio da frequência e o RLS (Recursive Least Square), cada um com particularidades específicas e desempenhos diferentes.

O LMS é o algoritmo mais utilizado em aplicações práticas de filtragem adaptativa. Tal algoritmo possui baixa complexidade computacional, mas seu comportamento durante a convergência varia de acordo com as características do sinal de entrada, acarretando numa convergência lenta para sinais de entrada correlacionados.

As exigências impostas nas aplicações práticas de filtragem adaptativa, tais como velocidade de convergência, complexidade computacional e robustez do algoritmo exigiram modificações no algoritmo LMS convencional, fazendo surgir novos algoritmos, tais como o NLMS (Normalized Least Mean Square), que possui maior complexidade

computacional do que o convencional, porém, devido à normalização, torna a convergência independente da potência do sinal de entrada e o algoritmo LMS no domínio da frequência, que a medida que o número de coeficientes do filtro adaptativo aumenta, a complexidade computacional diminui e sua convergência para sinais coloridos melhora em relação ao LMS convencional.

O algoritmo RLS, por outro lado, possui alta taxa de convergência, porém apresenta problemas de instabilidade numérica e alto custo computacional, sendo necessário muita cautela ao implementá-lo (SAYED, 2003; REGALIA, 1992). Outra característica de tal algoritmo, é que sua capacidade de acompanhar processos não estacionários nem sempre é adequada, como mostra (EWEDA, 1994), sendo seu desempenho inferior ao do LMS em algumas situações nesse quesito.

A maioria dos algoritmos existentes na literatura utiliza como função de custo o erro médio quadrático, como é o caso dos algoritmos acima citados. Tais algoritmos são baseados em estatísticas de segunda ordem (SOS), método suficiente quando se assume que os sinais envolvidos no processo têm distribuição Gaussiana, obtendo-se um grande número de simplificações na análise do comportamento do algoritmo, assim como proporcionando um menor custo computacional quando comparado a métodos baseados em estatística de alta ordem (HOS) (SANTANA et al., 2006). No entanto, alguns sinais naturais ou artificiais, não são necessariamente gaussianos, longe disso. Assim, o uso da estatística de alta ordem como uma forma de extrair mais informações dos sinais envolvidos no processo de adaptação tem se mostrado útil em sistemas adaptativos (SILVA et al., 2010).

Neste campo, de particular interesse para a abordagem aqui proposta podemos citar os trabalhos de (BARROS et al., 2003), (SANTANA et al., 2006) e (SILVA et al., 2010), ambos desenvolveram algoritmos baseados na utilização de estatísticas de alta ordem para a obtenção de mais informações sobre os sinais envolvidos no processo, com o objetivo de melhorar o desempenho de um filtro adaptativo.

(BARROS et al., 2003) desenvolveram uma família de algoritmos baseados na soma dos momentos pares do erro, inspirados na função de distribuição de probabilidade de uma variável aleatória, os quais podem ser escritos como uma combinação linear de seus momentos. Já (SANTANA et al., 2006), utilizaram uma função não linear, contínua e simétrica, a função  $f = \ln(\cosh(\alpha\epsilon))$ , como critério a ser aplicado sobre o erro, onde  $\alpha$  é um número inteiro, para desenvolver o Algoritmo Sigmoidal (SA) e (SILVA et al., 2010), desenvolveram o algoritmo RNL, utilizando como critério aplicado sobre o erro, uma função não linear, par, contínua e simétrica. Tal algoritmo é baseado em uma potência par do erro, inspirado no algoritmo RLS padrão.

Apesar da grande importância dos algoritmos adaptativos, a solução, obtida pelos algoritmos disponíveis na literatura, ainda não é a ótima. Isto justifica a grande

busca, pelos pesquisadores, de algoritmos que sanem essa deficiência.

## 1.2 Objetivos

Neste trabalho objetiva-se desenvolver uma família de algoritmos adaptativos baseado no método do gradiente descendente que utilize como critério de desempenho uma função de custo não linear para ajuste dos coeficientes do filtro.

No específico, esse trabalho visa contribuir nos seguintes aspectos:

- Um novo modelo de filtro adaptativo com função de custo não linear com rápida convergência;
- Realizar simulações computacionais no intuito de comparar o desempenho do algoritmo proposto com o algoritmo mais utilizado na literatura, nesse caso o LMS;
- Continuar contribuindo com o estudo de filtros digitais, em especial a filtragem adaptativa;
- Realizar aplicação na área de biomedicina, para cancelamento de ruído em sinal de eletrocardiograma.

## 1.3 Organização do texto

Este trabalho encontra-se dividido da seguinte forma: No capítulo 2, apresenta-se o conceito de filtro adaptativo assim como, a estrutura e princípio de funcionamento do mesmo; mostra-se algumas das principais aplicações dos filtros adaptativos; revisa-se filtragem ótima, mostrando a superfície quadrática, gerada pela utilização do método dos mínimos quadrados; faz-se uma breve revisão dos algoritmos da descida mais íngreme e *Least Mean Square* (LMS), mostrando a derivação, convergência do vetor peso e excesso de erro final do LMS.

No capítulo 3 mostra-se o desenvolvimento do algoritmo SAG, que utiliza como critério aplicado sobre o erro a função  $1 - e^{-\alpha\epsilon^2}$ , obtendo equação de atualização dos pesos do filtro, a expressão que garante a convergência e o desajuste.

No capítulo 4 realiza-se simulações no MATLAB, comparando-se o algoritmo SAG com o LMS, através de suas curvas de aprendizagem.

No capítulo 5 aplica-se o algoritmo desenvolvido na eliminação de ruído Gaussiano em sinal de ECG, através de simulações no software MATLAB.

E por fim, no capítulo 6 mostra-se a conclusão e proposta de continuidade do trabalho.

## 2 Filtros digitais

O termo filtro é usado para descrever um sistema de hardware ou software, cuja finalidade pode ser tanto recuperar informação contaminada por ruído ou simplesmente selecionar partes de interesse de um sinal (CORDEIRO, 2008). Filtros usados para tais propósitos podem ser de parâmetros fixos ou adaptativos.

### 2.1 Filtros fixos *versus* Filtros adaptativos

Os filtros digitais são classificados em filtros de parâmetros fixos e filtros adaptativo, como pode ser observado na Figura 1. Ambos os tipos de filtros possuem particularidades bem distintas, que irão definir em que tipo de aplicações poderão ser utilizados.



Figura 1 – Classificação de filtros digitais

Os filtros digitais de parâmetro fixo possuem as seguintes características:

- Podem ser lineares ou não lineares;  
Filtros lineares computam a saída do filtro utilizando uma combinação linear dos sinais de entrada. Caso contrário, diz-se que os filtros são não lineares.
- São invariantes no tempo;
- No processo de projeto, utiliza-se informações sobre a banda de frequência que o filtro atuará;
- Os coeficientes do filtro são escolhidos durante a fase de projeto e permanecem constantes durante toda a operação normal do filtro.

Em algumas aplicações práticas, a utilização de filtros de parâmetros fixos pode se tornar inadequada, seja por falta de informações suficientes para efetuar o projeto do filtro fixo ou devido a alterações ocorridas no projeto durante a operação normal de aplicação (AGOSTINHO, 2011).

Nesses casos, aplica-se os filtros adaptativos, que tem como principal vantagem a capacidade de adaptar-se automaticamente às alterações no sinal de entrada objetivando melhorar seu próprio desempenho durante a operação (TAKAYAMA et al., 2011). Suas principais características estão listadas abaixo:

- São filtros com coeficientes que variam no tempo;
- Adaptam-se automaticamente às alterações nos sinais de entrada;
- Baseiam-se em algoritmo de minimização do erro.

Os filtros adaptativos são comumente classificados como lineares ou não-lineares dependendo da estrutura de filtragem utilizada, isto é, desconsiderando sua característica adaptativa tendo em vista que um filtro adaptativo não é, a rigor, um sistema linear uma vez que não satisfaz o princípio da superposição durante sua adaptação, devido à contínua variação de seus coeficientes (BATISTA, 2009). Dessa forma, considerando coeficientes fixos, se a relação de entrada e saída obedece ao princípio da superposição, o filtro adaptativo é dito linear, caso contrário, é dito não-linear.

Embora a maioria das aplicações sejam restritas ao uso de filtros lineares, os filtros adaptativos não-lineares são indispensáveis em aplicações onde os fenômenos físicos a serem modelados são fortemente não-lineares (FARHANG-BOROUJENY, 2013). Dentre essas aplicações, destacam-se a equalização de sistemas de comunicação por satélite, o processamento de sinais dependente da percepção e o estudo das distorções harmônicas em alto falantes (BATISTA, 2009). No entanto, as estruturas de filtragem não-lineares estão fora do escopo do presente trabalho de pesquisa.

## 2.2 Filtro de Wiener

Para melhor compreender os conceitos relacionados a filtragem adaptativa, torna-se conveniente abordar a teoria de filtragem ótima (filtro de Wiener). O filtro de Wiener é um filtro linear ótimo no sentido do Erro Médio Quadrático (EMQ), como explicitado mais adiante.

A Figura 2 representa a estrutura geral de um filtro de Wiener, onde  $x_k$  e  $d_k$  representam respectivamente a entrada e o sinal desejado do filtro a cada instante  $k$ . O

filtro produz uma saída  $y_k$  que é comparada com o sinal desejado  $d_k$ , gerando um erro  $\varepsilon_k$ , dado por:

$$\varepsilon_k = d_k - y_k \quad (1)$$

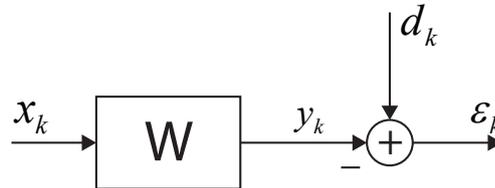


Figura 2 – Estrutura geral de um filtro de Wiener

Pode-se notar em 1 que quanto menor for o sinal do erro, mais próxima a saída será da resposta desejada. Dessa forma, o objetivo é ajustar os parâmetros do filtro de modo a minimizar o erro. Dentre os vários critérios que podem ser usados para orientar esse ajuste o critério adotado, nos filtros de Wiener, é a minimização da função do EMQ, onde o filtro busca minimizar o valor esperado  $E[\varepsilon_k^2]$  (NASCIMENTO, 2005). O Erro Médio Quadrático é dado por:

$$\xi = E[\varepsilon_k^2] \quad (2)$$

### 2.2.1 Solução de Wiener para Filtros Transversais

Existem diversas estruturas para a implementação de filtros lineares. Tais estruturas podem ser classificadas em dois grupos (BATISTA, 2009): estruturas com resposta ao impulso finita (*Finite Impulse Response - FIR*) e estruturas com resposta ao impulso infinita (*Infinite Impulse Response - IIR*). Dentre as estruturas FIR, os filtros transversais ou filtros de linha de atrasos são os mais utilizados, sendo o filtro de Wiener baseado nessa estrutura. Na Figura 3 pode-se ver o diagrama de blocos de um filtro FIR transversal.

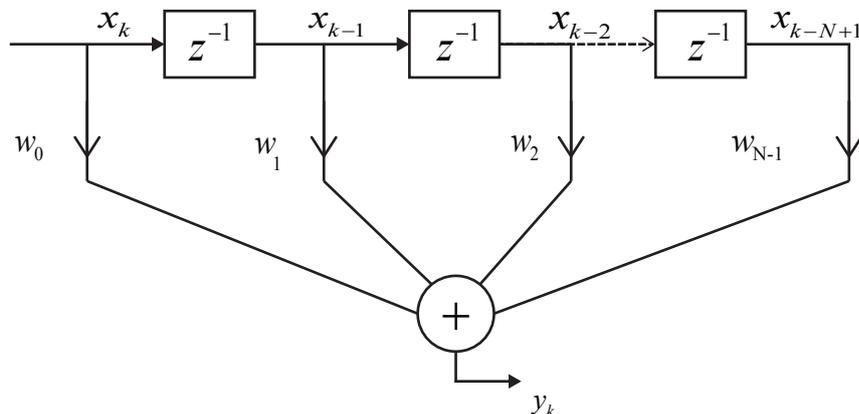


Figura 3 – Estrutura de um filtro FIR transversal

onde:

- $\mathbf{W} = [w_0 \ w_1 \ w_2 \ \dots \ w_{N-1}]$  e  $\mathbf{X}_k = [x_k \ x_{k-1} \ x_{k-2} \ \dots \ x_{k-N+1}]$ , representam, respectivamente, o vetor de coeficientes e o vetor de entrada do filtro. Sendo o subscrito  $k$  o índice de tempo.
- $y_k$  é a saída do filtro, dada por:

$$y_k = w_0x_k + w_1x_{k-1} + w_2x_{k-2} + \dots + w_{N-1}x_{k-N+1} = \sum_{i=0}^{N-1} w_i x(k-i) \quad (3)$$

$$y_k = \mathbf{X}_k^T \mathbf{W} \quad (4)$$

Considerando  $N - 1$  a ordem do filtro FIR, a sua saída é calculada utilizando somente o valor da entrada no instante  $k$  e nas  $N - 1$  amostras anteriores. Dessa forma, o filtro não é realimentado e por isso também é chamado de filtro não recursivo ou filtro de média movél.

Utilizando a Expressão 4, tem-se que o erro pode ser reescrito como:

$$\varepsilon_k = d_k - \mathbf{X}_k^T \mathbf{W} \quad (5)$$

onde  $T$  é a operação de transposição de matrizes.

Substituindo-se a Equação 5 em 2 e sabendo que  $\mathbf{X}_k^T \mathbf{W} = \mathbf{W}^T \mathbf{X}_k$ , tem-se :

$$E[\varepsilon_k^2] = E[d_k^2 - 2d_k \mathbf{X}_k^T \mathbf{W} + \mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W}] \quad (6)$$

Dessa forma, o EMQ é representado por:

$$\xi = E[d_k^2] - 2E[d_k \mathbf{X}_k^T] \mathbf{W} + \mathbf{W}^T E[\mathbf{X}_k \mathbf{X}_k^T] \mathbf{W} \quad (7)$$

Seja:

$$\bullet \mathbf{R} = E[\mathbf{X}_k \mathbf{X}_k^T] = E \begin{bmatrix} x_k^2 & x_k x_{k-1} & x_k x_{k-2} & \dots & x_k x_{k-N+1} \\ x_{k-1} x_k & x_{k-1}^2 & x_{k-1} x_{k-2} & \dots & x_{k-1} x_{k-N+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{k-N+1} x_k & x_{k-N+1} x_{k-1} & x_{k-N+1} x_{k-2} & \dots & x_{k-N+1}^2 \end{bmatrix}, \text{ a}$$

matriz de correlação de  $\mathbf{X}_k$ ;

- $\mathbf{P} = E[d_k \mathbf{X}_k] = E[d_k x_k \ d_k x_{k-1} \ d_k x_{k-2} \ \dots \ d_k x_{k-N+1}]^T$ , o vetor de correlação cruzada entre  $d_k$  e  $\mathbf{X}_k$ ;

- e  $E[d_k^2] = \sigma_d^2$ , a variância da resposta desejada

Na matriz  $\mathbf{R}$ , os termos da diagonal principal indicam as médias quadradas das componentes do sinal de entrada e os demais termos representam a correlação entre os elementos do sinal de entrada. Tal matriz indica a dependência entre os diversos termos do vetor  $\mathbf{X}_k$ , enquanto que o vetor de correlação cruzada,  $\mathbf{P}$ , mensura a dependência entre o sinal desejado e cada componente do vetor de sinal de entrada.

Substituindo  $\mathbf{R}$ ,  $\mathbf{P}$  e  $\sigma_d^2$  na Equação 7, podemos reescrevê-la da seguinte forma:

$$\xi = \sigma_d^2 - 2\mathbf{P}^T\mathbf{W} + \mathbf{W}^T\mathbf{R}\mathbf{W} \quad (8)$$

Observa-se na Equação 8 que o Erro Médio Quadrático é uma função quadrática em  $\mathbf{W}$ , com um único mínimo, já que possui forma de hiperparabolóide convexa. Tal superfície é conhecida como **superfície de desempenho** (ALMEIDA, 2004)

O gráfico da superfície de desempenho do Erro Médio Quadrático pode ser observado na Figura 4, sendo o Erro Médio Quadrático plotado na vertical e os pesos  $w_0$  e  $w_1$  variando de -1 a 1, supondo que estamos trabalhando em um filtro com apenas dois pesos.

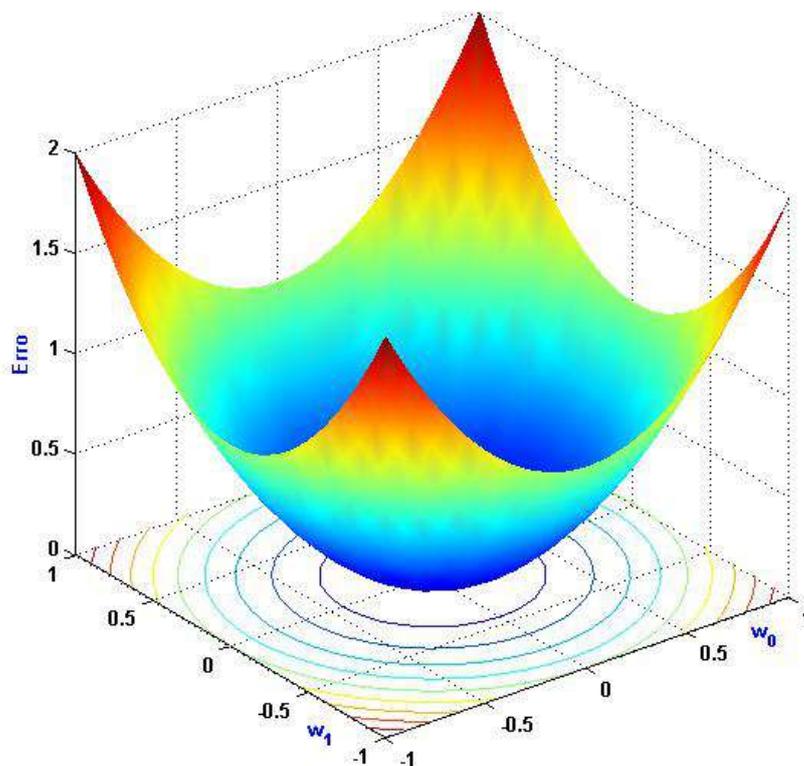


Figura 4 – Porção de uma superfície quadrática tridimensional

O ponto mínimo da superfície de desempenho do erro médio quadrático pode ser obtido, calculando-se o gradiente em relação ao vetor de pesos do filtro adaptativo e igualando esse resultado a zero. O gradiente de tal superfície é obtido, derivando-se a Equação 8 em relação a cada peso:

$$\nabla \xi = \frac{\partial \xi}{\partial \mathbf{W}} = \left[ \frac{\partial \xi}{\partial w_0} \frac{\partial \xi}{\partial w_1} \dots \frac{\partial \xi}{\partial w_N} \right]^T \quad (9)$$

$$\nabla \xi = \frac{\partial(-2\mathbf{P}^T \mathbf{W})}{\partial \mathbf{W}} + \frac{\partial(\mathbf{W}^T \mathbf{R} \mathbf{W})}{\partial \mathbf{W}} \quad (10)$$

Assim,

$$\nabla \xi = -2\mathbf{P} + 2\mathbf{R}\mathbf{W} \quad (11)$$

Igualando a Expressão 11 a zero para encontrar o vetor de pesos ótimos,  $\mathbf{W}_*$ , temos:

$$-2\mathbf{P} + 2\mathbf{R}\mathbf{W} = 0 \quad (12)$$

Considerando que a matriz de autocorrelação,  $\mathbf{R}$ , seja definida positiva e que  $\mathbf{R}^{-1}$  exista, podemos pré-multiplicar a Equação 12 pela inversa da matriz de autocorrelação,  $\mathbf{R}$ , resultando na expressão 13, conhecida como **solução de Wiener** no sentido médio quadrático.

$$\mathbf{W}_* = \mathbf{R}^{-1}\mathbf{P} \quad (13)$$

Para encontrar os parâmetros do filtro de Wiener deve-se, a princípio, resolver o sistema de equações de Wiener-Hopf, dado pela Equação 13. A resolução de tal sistema requer a matriz de autocorrelação,  $\mathbf{R}$ , e o vetor de correlação cruzada,  $\mathbf{P}$ , que geralmente não são conhecidos a priori, além de necessitar de uma inversão matricial que é computacionalmente dispendiosa. Além do mais, se os sinais não forem estacionários,  $\mathbf{R}$  e  $\mathbf{P}$  variam no tempo e  $\mathbf{W}$  precisa ser repetidamente calculado.

Através de um algoritmo adaptativo, o filtro deve ajustar os seus pesos de modo a aproximá-los o máximo possível do vetor de pesos ótimos, representado pela solução de Wiener.

## 2.3 Filtro adaptativo

Filtros adaptativos são filtros que ajustam seus próprios parâmetros conforme as características de um sinal que se deseja obter. Tais filtros são formados por três módulos, que podem ser observados na Figura 5 (NUNES, 2014).

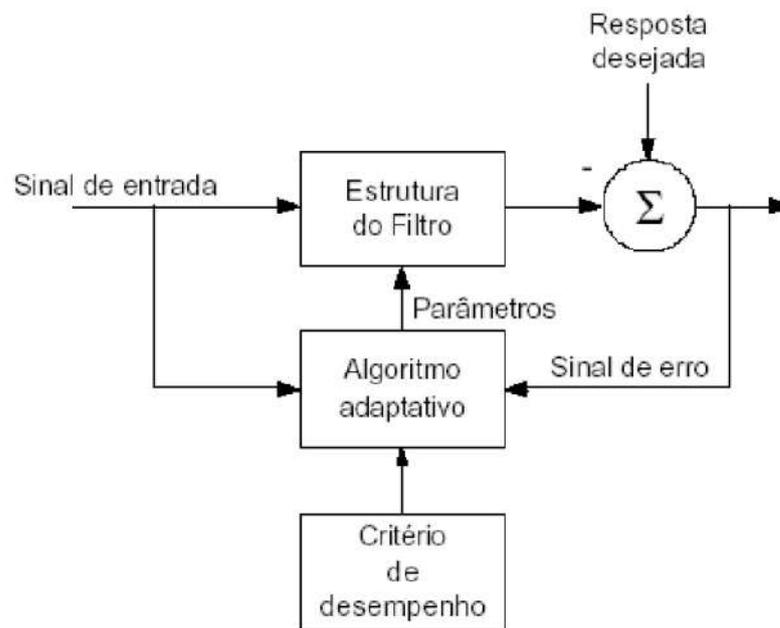


Figura 5 – Elementos de um filtro adaptativo

- **Estrutura do filtro:** Esse módulo produz a saída do filtro a partir de medições do sinal de entrada. A estrutura é fixa e seus parâmetros são ajustados pelo algoritmo adaptativo.
- **Critério de desempenho:** A saída do filtro adaptativo e a resposta desejada são processadas pelo módulo de critério de desempenho para avaliar sua qualidade em relação aos requisitos da aplicação.
- **Algoritmo adaptativo:** O algoritmo adaptativo tem por objetivo ajustar os pesos do filtro, satisfazendo um determinado critério de desempenho, que normalmente é a minimização de uma função do erro do processo de adaptação.

Na escolha do algoritmo de adaptação deve-se levar em consideração: a velocidade de convergência e precisão do algoritmo, a complexidade numérica do algoritmo, a estabilidade da estrutura do algoritmo (HAYKIN, 2001)

- A velocidade de convergência: número de iterações necessárias para que o algoritmo, perante sinais estacionários, se aproxime o suficiente da solução ótima de Wiener. Quanto maior a velocidade de convergência, mais rapidamente o algoritmo se adaptará a um ambiente estacionário de características desconhecidas.
- A complexidade numérica: refere-se ao número de operações a efetuar para completar uma iteração do algoritmo.
- Dependendo do algoritmo de adaptação poderá haver instabilidade numérica ao ser concretizado num processador

## 2.4 Estrutura de um filtro adaptativo

Os filtros adaptativos são compostos basicamente por um filtro digital e um algoritmo adaptativo, sendo o algoritmo adaptativo responsável por ajustar os coeficientes do filtro digital de modo que o erro do processo de adaptação seja minimizado progressivamente, fazendo com que o sinal de saída do filtro seja o mais próximo possível do sinal desejado. A estrutura geral de um filtro adaptativo é apresentada na Figura 6.

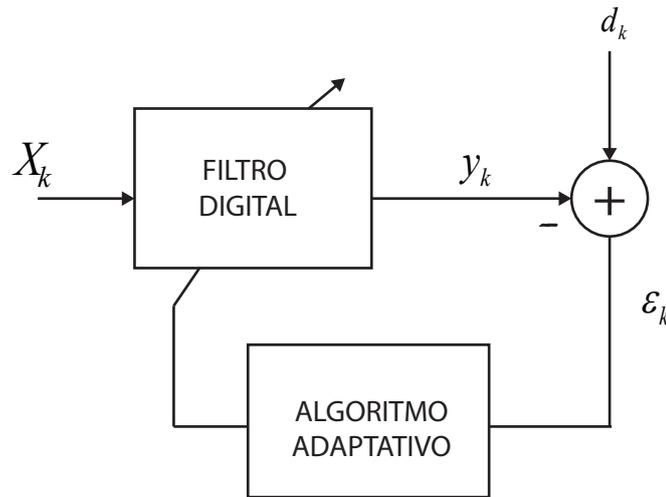


Figura 6 – Estrutura geral de um filtro adaptativo

onde  $\mathbf{X}_k = [x_k \ x_{k-1} \ x_{k-2} \ \dots \ x_{k-N+1}]$  é o vetor de entrada do filtro adaptativo e  $\mathbf{W}_k = [w_{0k} \ w_{1k} \ w_{2k} \ \dots \ w_{(N-1)k}]$  são os  $N$  coeficientes do filtro que podem se alterar a cada instante  $k$ .

Filtros adaptativos normalmente usam filtros FIR, devido à sua simplicidade e garantia de estabilidade. Portanto a saída pode ser calculada por:

$$y_k = w_{0k}x_k + w_{1k}x_{k-1} + w_{2k}x_{k-2} + \dots + w_{(N-1)k}x_{k-N+1} = \sum_{i=0}^{N-1} w_{ik}x(k-i) \quad (14)$$

$$y_k = \mathbf{X}_k^T \mathbf{W}_k \quad (15)$$

É interessante observar que a única diferença entre a Equação 14 e 3 é que os coeficientes do filtro podem variar a cada instante de amostragem.

Pode-se interpretar a filtragem adaptativa estruturada na Figura 6 da seguinte forma: o sistema fornece o sinal de entrada do filtro,  $\mathbf{X}_k$ , que é ponderado pelos

coeficientes do filtro adaptativo,  $W_k$ , gerando a saída  $y_k$ . A saída é subtraída do sinal desejado,  $d_k$ , resultando num erro, dado por:

$$\varepsilon_k = d_k - y_k \quad (16)$$

O processo de adaptação dos coeficientes do filtro consiste na otimização de uma função, chamada função de custo, que em geral, depende do sinal de erro (GONTIJO, 2006).

## 2.5 Aplicações

Como já mencionado anteriormente, a filtragem adaptativa possui um vasto campo de aplicações, sendo que a maioria dessas aplicações recai em uma das seguintes classes de problemas: predição, identificação de sistemas, modelagem inversa e cancelamento de ruídos. Tais classes de problemas são descritas como segue:

### – Predição linear:

O filtro adaptativo, neste caso, é utilizado para prever valores futuros de um sinal de entrada, veja Figura 7. O sinal de entrada  $s$  é o próprio sinal desejado  $d$  e o sinal  $x$  é o sinal  $s$  com atraso.

Os valores passados do sinal  $s$  fornecem a entrada aplicada ao filtro adaptativo, e dependendo da aplicação de interesse, a saída do filtro adaptativo ou o erro estimado (predito) pode servir como saída do sistema. No caso de a saída do filtro ser a resposta do sistema, diz-se que o filtro adaptativo opera como um preditor, enquanto que no caso em que se usa o erro estimado como saída do sistema diz-se que o filtro adaptativo opera como um preditor de erro (CASTOLDI, 2007).

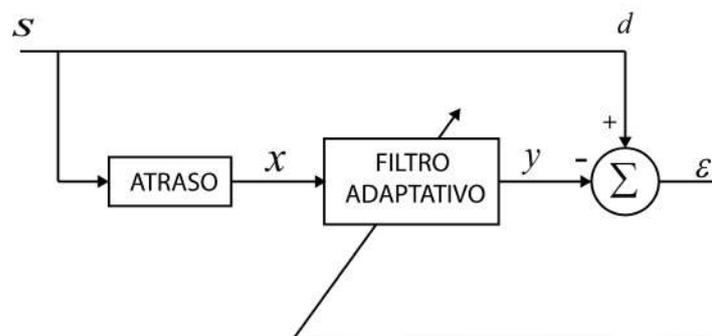


Figura 7 – Estrutura de um filtro adaptativo para predição linear

### – Identificação de sistemas:

Essa classe de aplicação é também conhecida como modelagem de sistemas, o objetivo do filtro adaptativo aqui é fornecer um modelo linear que melhor represente uma planta desconhecida, veja a Figura 8. O sinal  $s$ , alimenta tanto a planta quanto o filtro adaptativo e a saída da planta é a resposta desejado do filtro adaptativo.

A identificação adaptativa é frequentemente utilizada em sistemas de comunicações, sistemas de controle e em identificação de estruturas. Nos sistemas de controle, por exemplo, pode ser utilizada para modelar uma planta com a finalidade de projetar um controlador (MALUENDA et al., 2005).

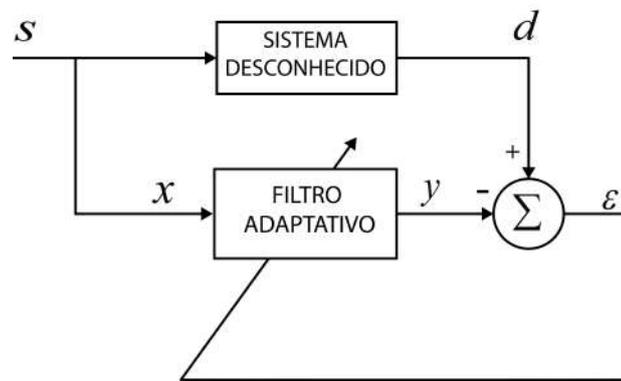


Figura 8 – Estrutura de um filtro adaptativo para identificação de sistemas

– Modelagem inversa:

Nessa aplicação a função do filtro adaptativo é encontrar o melhor modelo inverso para reduzir ou anular a distorção provocada por um canal, veja a Figura 9. O sinal que foi transmitido ou distorcido por um canal é utilizado como entrada do filtro adaptativo e uma versão atrasada do sinal original  $s$  é utilizado como sinal de referência.

Uma aplicação da modelagem inversa é a equalização de canal, cujo objetivo é reduzir a interferência inter simbólica através de um filtro adaptativo que inverta a resposta do canal (MALUENDA et al., 2005).

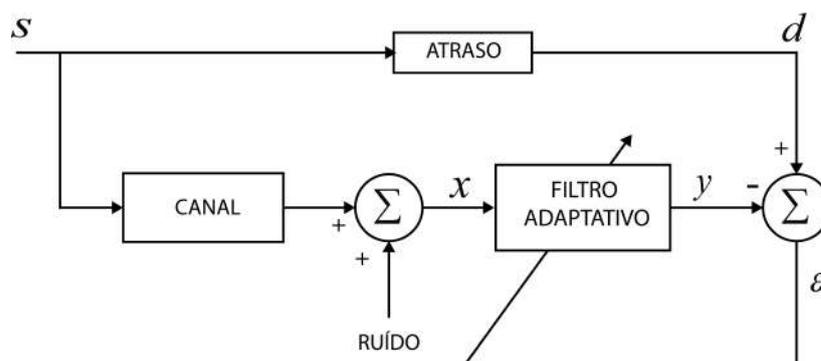


Figura 9 – Estrutura de um filtro adaptativo para modelagem inversa

– Cancelamento de ruído:

O filtro adaptativo é utilizado para cancelar um ruído desconhecido contido em um sinal primário  $s$ , conforme Figura 10. Essa aplicação, cujo dispositivo é conhecido como controlador ativo de ruído (ANC – *Active Noise Canceller*), consiste na captação de um sinal corrompido por um ruído. O filtro adaptativo procura minimizar, ou cancelar completamente em um cenário ideal este ruído, resultando em um sinal similar ao original antes da ação desse ruído. O sinal primário  $s$  corrompido por um ruído  $n$ , é utilizado como sinal de desejado do filtro adaptativo, ou  $d = s + n$  e um sinal de ruído  $n'$  é utilizado como entrada do filtro (fornece uma referência sobre as características do ruído  $n$ ). Os sinais  $n$  e  $n'$  são gerados pela mesma fonte e são correlacionados entre si e não são correlacionados com o sinal  $s$ .

O filtro adaptativo processa o sinal de referência  $n'$ , produzindo a saída  $y$  que é comparada ao sinal desejado  $d$ , gerando um erro dado por:

$$\begin{aligned} e &= d - y \\ &= s + n - y \end{aligned} \quad (17)$$

O sinal de erro constitui a saída do sistema de cancelamento de ruído e é utilizado para recalibrar os pesos internos do filtro adaptativo. Note que na Equação 17, o componente do ruído na saída do filtro é dado por  $n - y$ . A função do filtro adaptativo é minimizar o valor médio quadrático do sinal de erro sem afetar o sinal  $s$ . Portanto, minimizar o Erro Médio Quadrático equivale em minimizar o valor médio quadrático do ruído na saída,  $n - y$ . Dessa forma, a função do filtro adaptativo é filtrar o ruído  $n'$  de tal forma que a sua resposta  $y$  fique o mais próximo possível do ruído  $n$ . Quanto mais a saída do filtro  $y$  se aproximar do ruído  $n$  mais perfeita é a filtragem.

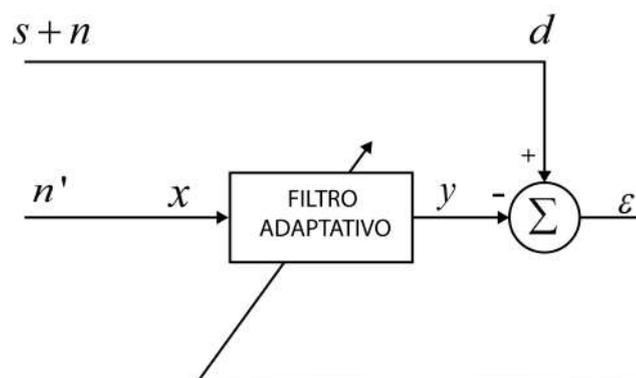


Figura 10 – Estrutura de um filtro adaptativo para cancelamento de ruídos

## 2.6 Algoritmos adaptativos

Os algoritmos em filtragem adaptativa dividem-se em duas classes: algoritmos de gradiente e algoritmos de mínimos quadrados.

- Algoritmos de gradiente: baseiam-se no filtro de Wiener, estimando o gradiente da superfície das funções de custo. Tem como principal vantagem a baixa complexidade. São bons na exploração de superfícies pouco complicadas.
- Algoritmos de mínimos quadrados: são baseados no filtro de Kalman. De forma determinística minimizam a soma dos quadrados dos erros parciais. Tem como vantagem a baixa sensibilidade a mínimos locais da superfície da função de custo e a maior velocidade de convergência comparativamente com os algoritmos de gradiente.

### 2.6.1 Algoritmo da descida mais íngreme

Ajustar os coeficientes do filtro, significa descer a superfície de desempenho de forma a encontrar o mínimo da função de custo. Uma forma de atingir essa finalidade é utilizando o algoritmo da descida mais íngreme. Tal algoritmo é uma técnica recursiva baseada no método do gradiente descendente utilizada para a determinar o conjunto de pesos  $\mathbf{W}$  que minimiza a função de custo  $\xi$ .

Dado que a função de custo a ser minimizada é convexa, o conjunto de pesos ótimos será obtido iterativamente através de pequenos passos na direção para a qual a função decresça mais rapidamente, ou seja, no sentido contrário da projeção do gradiente da função de custo considerada ( $\nabla \xi_k$ ), já que o mesmo corresponde à direção de maior crescimento da função (GONTIJO, 2006)

O método da descida mais íngreme é formalmente descrito através da seguinte equação recursiva:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \frac{1}{2} \mu \nabla \xi_k \quad (18)$$

De acordo com este método, o “próximo” vetor peso,  $\mathbf{W}_{k+1}$ , é igual ao vetor peso “atual”,  $\mathbf{W}_k$ , mais uma quantidade proporcional ao negativo do gradiente. O filtro é inicializado com um vetor de pesos qualquer e o mesmo é ajustado recursivamente, caminhando em direção ao conjunto de pesos ótimos, dado pela solução de Wiener.

Para determinar uma solução recursiva, o vetor de coeficientes passou a ser variante no tempo. Portanto:

$$\mathbf{W} \Rightarrow \mathbf{W}_k \quad (19)$$

$$\xi \Rightarrow \xi_k \quad (20)$$

Dessa forma o gradiente da Expressão 11 pode ser reescrito como:

$$\nabla \xi_k = -2\mathbf{P} + 2\mathbf{R}\mathbf{W}_k \quad (21)$$

A Equação 21 representa o gradiente na iteração  $k$ .

Substituindo a Equação 21 em 18, pode-se reescrever a equação de atualização dos pesos da seguinte forma:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu[\mathbf{P} + \mathbf{R}\mathbf{W}_k] \quad (22)$$

## 2.6.2 Algoritmo Least-Mean-Square (LMS)

Apesar do processo iterativo utilizado pelo algoritmo da descida mais íngreme eliminar a necessidade da inversão da matriz  $\mathbf{R}$ , ainda torna-se necessário ter conhecimento das estatísticas de segunda ordem ( $\mathbf{R}$  e  $\mathbf{P}$ ) dos dados a processar. Como visto anteriormente, na maioria das aplicações práticas de filtragem adaptativa essas estatísticas não são conhecidas a priori, sendo dessa forma impossível obter em tempo real, o vetor gradiente exato a cada iteração.

Uma maneira de contornar essa situação é utilizando o algoritmo LMS. Tal algoritmo utiliza o método do gradiente e é uma implementação estocástica do algoritmo da descida mais íngreme. O algoritmo LMS proposto por Widrow e Hoff em 1960, é um dos algoritmos mais utilizados na implementação de filtros adaptativos. Esse amplo campo de aplicações pode ser atribuída a sua simplicidade e baixa complexidade computacional (HAYKIN, 2001)

### 2.6.2.1 Derivação do algoritmo LMS

O algoritmo LMS utiliza uma aproximação estocástica do vetor gradiente para atualizar o vetor de coeficientes  $\mathbf{W}_k$ , usando o erro quadrático instantâneo  $\varepsilon_k^2$  como uma aproximação para o erro médio quadrático  $E[\varepsilon_k^2]$ . Portanto, a cada iteração no processo adaptativo uma estimativa para o gradiente pode ser obtida por:

$$\begin{aligned} \hat{\nabla} \xi_k &= \frac{\partial \varepsilon_k^2}{\partial \mathbf{W}} = \left[ \frac{\partial \varepsilon_k^2}{\partial w_0} \quad \frac{\partial \varepsilon_k^2}{\partial w_1} \quad \cdots \quad \frac{\partial \varepsilon_k^2}{\partial w_N} \right]^T \\ &= 2\varepsilon_k \left[ \frac{\partial \varepsilon_k}{\partial w_0} \quad \frac{\partial \varepsilon_k}{\partial w_1} \quad \cdots \quad \frac{\partial \varepsilon_k}{\partial w_N} \right]^T \end{aligned} \quad (23)$$

Derivando-se então a Expressão 5 em relação ao vetor de pesos  $\mathbf{W}$ , tem-se:

$$\frac{\partial \varepsilon_k}{\partial \mathbf{W}} = \left[ \frac{\partial \varepsilon_k}{\partial w_0} \quad \frac{\partial \varepsilon_k}{\partial w_1} \quad \cdots \quad \frac{\partial \varepsilon_k}{\partial w_N} \right]^T = -\mathbf{X}_k \quad (24)$$

Substituindo-se a Expressão 24 em 23, tem-se a seguinte estimativa do gradiente:

$$\hat{\nabla}\xi_k = -2\varepsilon_k\mathbf{X}_k \quad (25)$$

Usando a Equação 25 em 18, resulta na seguinte equação de atualização de pesos do LMS :

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu\varepsilon_k\mathbf{X}_k \quad (26)$$

Os passos do algoritmo LMS são resumidos na Tabela 1.

Tabela 1 – Resumo do Algoritmo LMS

Entradas	Vetor de pesos ( $\mathbf{W}_k$ ), vetor de entradas ( $\mathbf{X}_k$ ) e saída desejada ( $d_k$ )
Saídas	Saída do filtro ( $y_k$ ) e vetor peso atualizado ( $\mathbf{W}_{k+1}$ )
Parâmetros	Tamanho do passo de atualização ( $\mu$ )
Inicialização	Normalmente inicia-se $\mathbf{W}_0$ com 0
Passo 1	$y_k = \mathbf{W}_k^T \mathbf{X}_k$
Passo 2	$\varepsilon_k = d_k - y_k$
Passo 3	$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu\varepsilon_k\mathbf{X}_k$

A expressão do gradiente instantaneo, pode ser obtido substituindo-se a Equação 5 em 25, assim:

$$\hat{\nabla}\xi_k = -2[d_k\mathbf{X}_k - \mathbf{X}_k\mathbf{X}_k^T\mathbf{W}_k] \quad (27)$$

Tomando-se o valor esperado da Equação 27 para determinar o comportamento médio do gradiente, tem-se:

$$\begin{aligned} E[\hat{\nabla}\xi_k] &= -2\{E[d_k\mathbf{X}_k - \mathbf{X}_k\mathbf{X}_k^T\mathbf{W}_k]\} \\ &= -2\{E[d_k\mathbf{X}_k] - E[\mathbf{X}_k\mathbf{X}_k^T\mathbf{W}_k]\} \\ &= -2\mathbf{P} + 2\mathbf{R}\mathbf{W}_k \end{aligned} \quad (28)$$

Com isso, verifica-se que a Expressão 28 representa o gradiente exato de  $\xi_k$ , usado pelo algoritmo da descida mais íngreme para atualização dos pesos do filtro adaptativo. Portanto, na média, o algoritmo LMS produz o mesmo resultado do algoritmo da descida mais íngreme.

### 2.6.2.2 Convergência do LMS

A Equação 26 representa o processo iterativo do algoritmo LMS. A cada iteração, o algoritmo calcula uma nova estimativa da solução ótima na tentativa de minimizar o Erro Médio Quadrático do sistema.

O parâmetro  $\mu$  é um fator que controla a estabilidade e a velocidade de convergência do algoritmo (SANTANA, 2006). Um  $\mu$  muito alto resultará em instabilidade

e oscilações. Por outro lado, em situações de não-estacionariedade, um  $\mu$  muito baixo resultará em baixa capacidade de adaptação às mudanças nas estatísticas do sinal de entrada.

Portanto para garantir a estabilidade e convergência do algoritmo LMS, tal parâmetro deve obedecer a seguinte condição:

$$0 < \mu < \frac{1}{\lambda_{max}} \quad (29)$$

onde  $\lambda_{max}$  é o maior autovalor da matriz de autocorrelação  $\mathbf{R}$ .

A Figura 11 apresenta dois processos de busca do mínimo de uma função de custo ao longo de diversas iterações, sendo que o processo da esquerda tem maior parâmetro  $\mu$ , portanto necessita de menos iterações para se aproximar do mínimo, no entanto possui maior ruído (ZIPF et al., 2011).

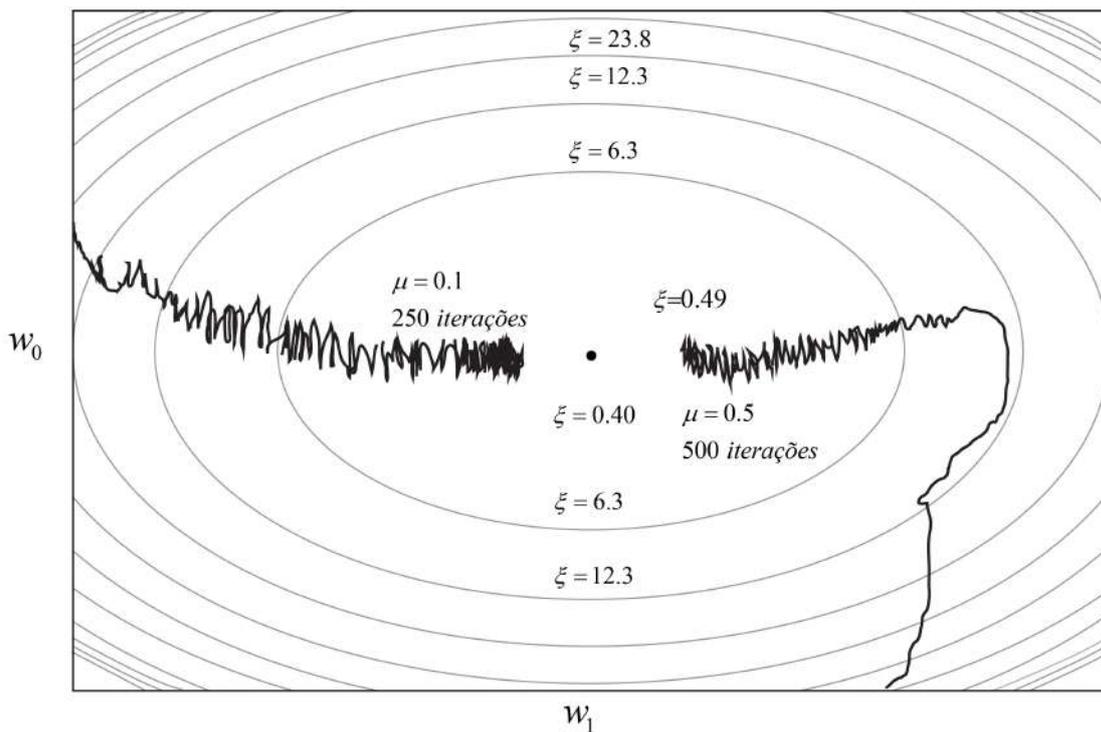


Figura 11 – Superfície de desempenho do algoritmo LMS e processo de convergência

### 2.6.2.3 Excesso do Erro Médio Quadrático

No processo adaptativo, quando os pesos finais estão próximos, mais não iguais a  $\mathbf{W}_*$ , o Erro Médio Quadrático ( $\xi$ ) será maior que o Erro Médio Quadrático mínimo ( $\xi_{min}$ ). Temos, assim, um excesso no erro final. Define-se como excesso do Erro Médio Quadrático, ExcessoMSE, como sendo a diferença entre o Erro Médio Quadrático

atual ( $\xi_k$ ) e o Erro Médio Quadrático mínimo ( $\xi_{\min}$ ), como pode ser visto na Figura 12 (WIDROW; STEARNS, 1985)

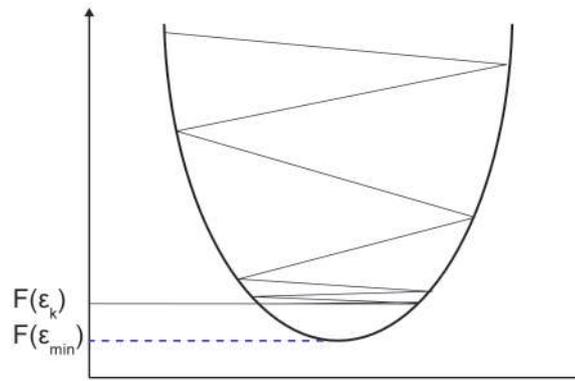


Figura 12 – Excesso no erro final em relação ao erro mínimo

Assim, o excesso de erro final é definido por:

$$\begin{aligned} \text{ExcessoMSE} &= E[\xi_k - \xi_{\min}] \\ &= \mu E[n_k^2] \text{tr}[\mathbf{R}] \end{aligned} \quad (30)$$

sendo  $\text{tr}[\mathbf{R}]$  o traço da matriz  $\mathbf{R}$  e  $n_k$  um sinal de ruído.

Define-se, como o desajuste ( $M$ ), a diferença entre o Erro Médio Quadrático atual e o Erro Médio Quadrático mínimo, normalizado pelo Erro Médio Quadrático mínimo, como segue:

$$M = \frac{E[\xi_k - \xi_{\min}]}{\xi_{\min}} \quad (31)$$

Assim, temos que:

$$M_{LMS} = \mu \text{tr}[\mathbf{R}] \quad (32)$$

## 3 Algoritmo proposto

Para projetar novos algoritmos, com características interessantes, basta alterar adequadamente a função-custo (DINIZ, 1997). Neste capítulo, mostraremos o desenvolvimento do algoritmo adaptativo chamado SAG. Tal algoritmo é baseado no método do gradiente descendente e utiliza a função  $1 - e^{-\alpha\epsilon^2}$  como critério a ser aplicado sobre o erro.

### 3.1 Superfície de desempenho da função

A superfície de desempenho gerada pela função de custo utilizada por um algoritmo nos fornece importantes informações sobre o comportamento desse algoritmo, portanto torna-se necessário um breve estudo da superfície de desempenho do algoritmo SAG. A idéia básica é mostrar que a superfície de desempenho obtida por essa função oferece maior velocidade de convergência, bem como um menor desajuste na busca do peso mínimo, quando comparada a superfície de desempenho do LMS.

A função  $1 - e^{-\alpha\epsilon^2}$  é uma não linearidade, contínua e simétrica, cujo gráfico está representado na Figura 13. Como podemos observar, esta função não possui mínimo local, apenas o mínimo global.

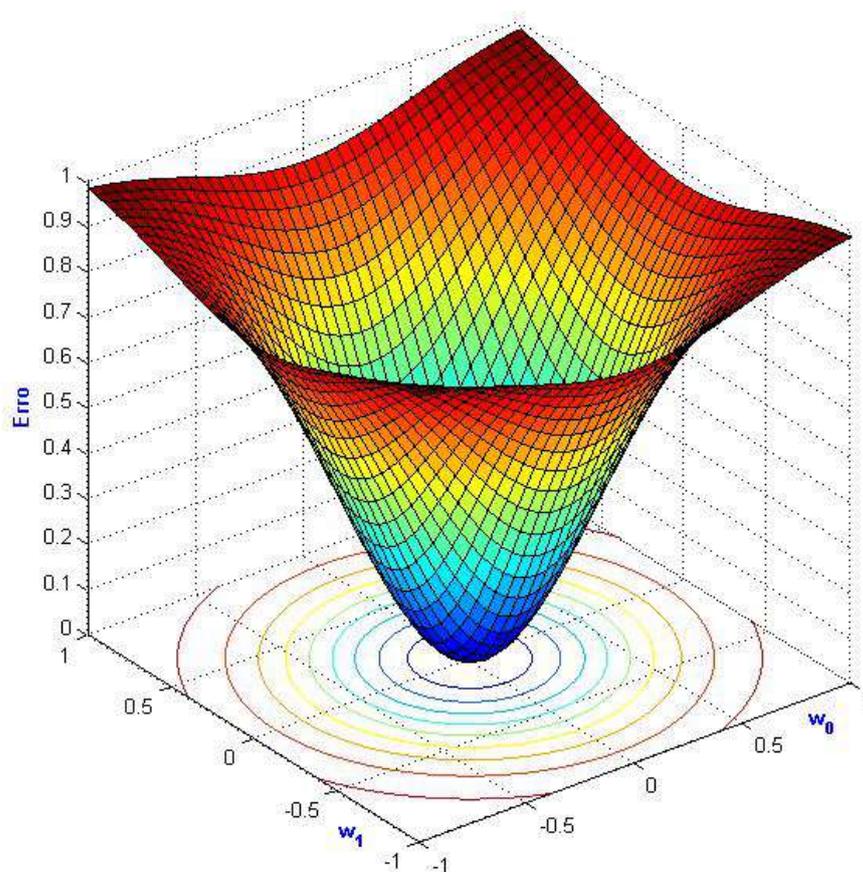


Figura 13 – Gráfico da superfície gerada pela função  $1 - e^{-2\epsilon^2}$  e algumas curvas de níveis. Os pesos  $w_0$  e  $w_1$  variam de -1 a 1.

A partir desta função é possível obter uma família de funções,  $1 - e^{-\alpha\epsilon^2}$ , multiplicando o argumento  $\epsilon$  por um inteiro positivo  $\alpha$ , conforme observamos na Figura 14.

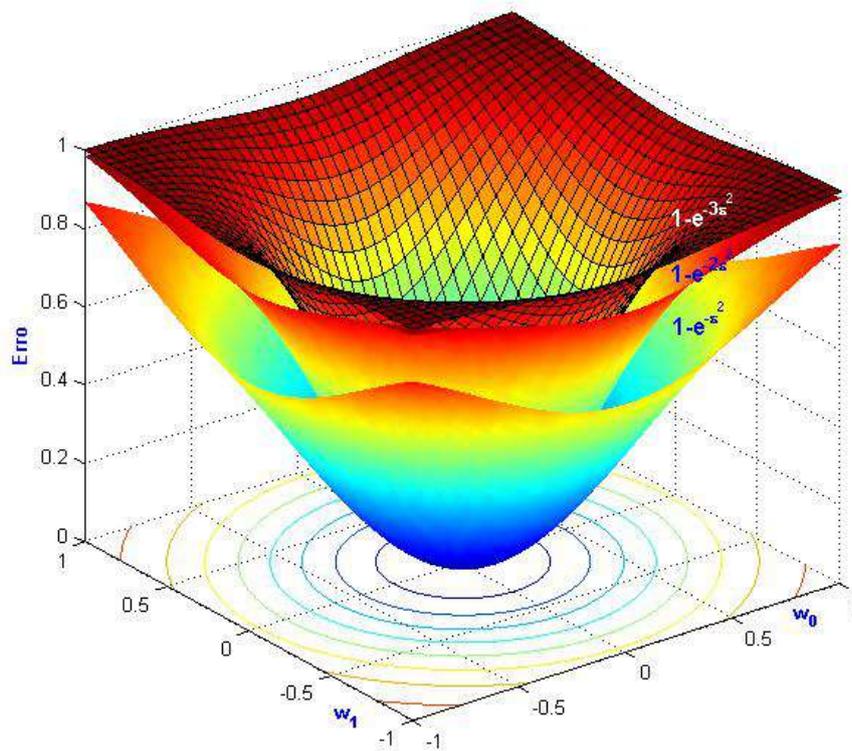


Figura 14 – Gráficos das funções  $1 - e^{-\epsilon^2}$ ,  $1 - e^{-2\epsilon^2}$  e  $1 - e^{-3\epsilon^2}$

É possível observar na Figura 14 que a superfície de desempenho da função torna-se mais inclinada a medida que tomamos valores maiores de  $\alpha$

Outra particularidade dos elementos deste conjunto de funções é que, para um valor fixo de  $\alpha$ , pode-se determinar intervalos  $[-\delta, \delta]$  onde as curvas destas funções tem inclinações maiores do que a curva da função  $\epsilon^2$ . Conforme pode-se observar na Figura 15

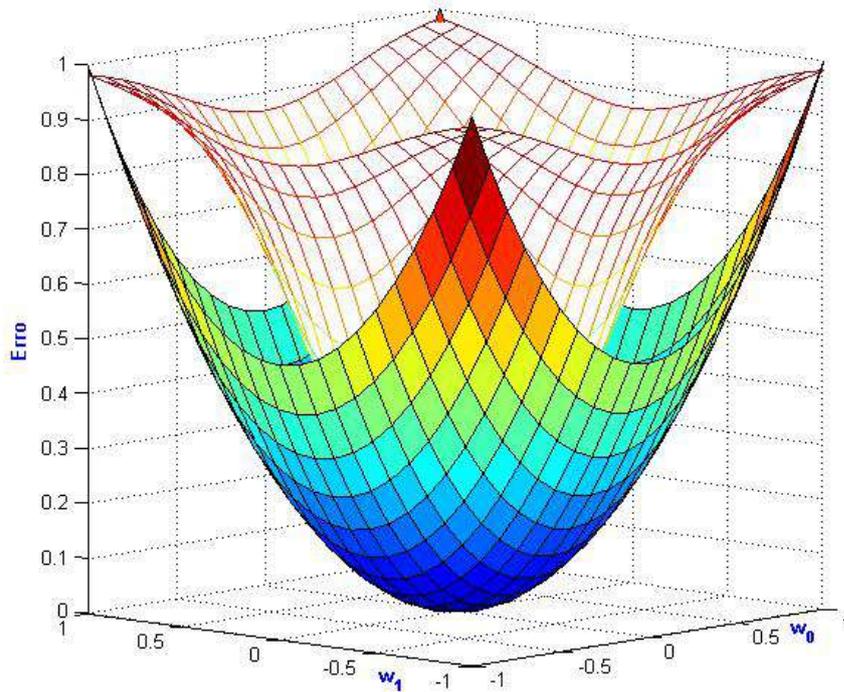


Figura 15 – Gráficos das funções  $1 - e^{-2\epsilon^2}$  e  $\epsilon^2$ , onde podemos ver a maior inclinação da primeira, no intervalo  $[-1,1]$

## 3.2 Método

Considera-se a Figura 16, onde  $d_j$  é um sinal desejado,  $\mathbf{X}_j$  é a entrada do filtro e  $y_j$  é a saída do filtro, sendo  $d_j = s_j + n_j$ , onde  $s_j$  é o sinal que se deseja extrair e  $n_j$  é um ruído. Supondo-se que  $n_j$  seja estatisticamente independente de  $s_j$  e  $\mathbf{X}_j$  e que todas as variáveis têm distribuição de probabilidade que não são necessariamente Gaussiana. Deseja-se recuperar  $s_j$ , estimando-se um sinal de saída  $y_j = \mathbf{W}_j^T \mathbf{X}_j$ , para depois calcular  $\epsilon_j = d_j - y_j$ .

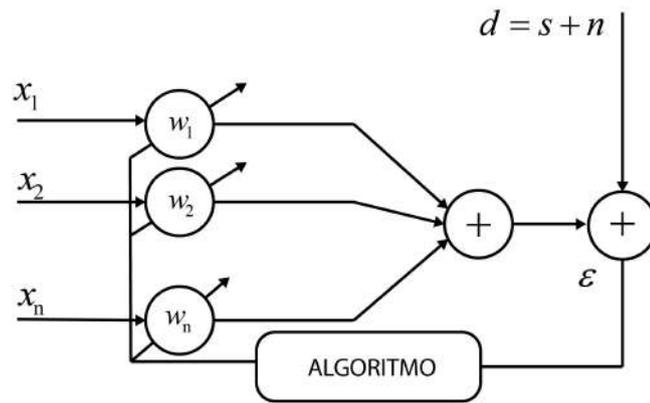


Figura 16 – Diagrama de blocos do filtro com um vetor de entrada  $\mathbf{X}$ , um sinal desejado  $d$ , composto por um sinal  $s$  e um ruído  $n$  e o erro  $\varepsilon$

A cada iteração, no processo adaptativo, o algoritmo SAG utiliza uma estimativa instantânea do gradiente da função  $F_j = 1 - e^{-\alpha \varepsilon_j^2}$ , dada por:

$$\hat{\nabla} F_j = -2\alpha \varepsilon_j e^{-\alpha \varepsilon_j^2} \mathbf{X}_j \quad (33)$$

A partir desta simples estimação do gradiente, podemos especificar a equação de atualização dos pesos do filtro dada pelo algoritmo SAG, como segue:

$$\mathbf{W}_{j+1} = \mathbf{W}_j - \mu \hat{\nabla}_j \quad (34)$$

$$= \mathbf{W}_j + 2\mu\alpha \varepsilon_j e^{-\alpha \varepsilon_j^2} \mathbf{X}_j \quad (35)$$

Os passos do algoritmo SAG são resumidos na Tabela 2.

Tabela 2 – Resumo do Algoritmo SAG

Entradas	Vetor de pesos ( $\mathbf{W}_j$ ), vetor de entradas ( $\mathbf{X}_j$ ) e saída desejada ( $d_j$ )
Saídas	Saída do filtro ( $y_j$ ) e vetor peso atualizado ( $\mathbf{W}_{j+1}$ )
Parâmetros	Tamanho do passo de atualização ( $\mu$ ) e de $\alpha$
Inicialização	Normalmente inicia-se $\mathbf{W}_0 = 0$
Passo 1	$y_j = \mathbf{W}_j^T \mathbf{X}_j$
Passo 2	$\varepsilon_j = d_j - y_j$
Passo 3	$\mathbf{W}_{j+1} = \mathbf{W}_j + 2\mu\alpha \varepsilon_j e^{-\alpha \varepsilon_j^2} \mathbf{X}_j$

Como antes,  $\mu$  é uma constante que controla a velocidade e a estabilidade da adaptação.

### 3.3 Comportamento e convergência do vetor peso

O comportamento do vetor peso pode ser analisado verificando-se as condições sobre as quais o algoritmo converge para a solução desejada, e seu comportamento até

atingir o estado estacionário. Isso é possível através do estudo do desajuste e do tempo de convergência (NAHUZ, 2016). Definindo-se  $\mathbf{V}_j = \mathbf{W}_j - \mathbf{W}_*$ , onde  $\mathbf{W}_*$  é o vetor de peso ótimo, ou seja,  $s_j = \mathbf{W}_*^T \mathbf{X}_j$ . Pode-se reescrever a equação  $d_j = s_j + n_j$  da seguinte forma:

$$d_j = \mathbf{W}_*^T \mathbf{X}_j + n_j \quad (36)$$

Fazendo-se a substituição da Expressão 36 em  $\varepsilon_j = d_j - y_j$  e lembrando que  $y_j = \mathbf{W}_j^T \mathbf{X}_j$ , tem-se:

$$\begin{aligned} \varepsilon_j &= \mathbf{W}_*^T \mathbf{X}_j + n_j - \mathbf{W}_j^T \mathbf{X}_j \\ &= n_j - (\mathbf{W}_j^T - \mathbf{W}_*^T) \mathbf{X}_j \end{aligned} \quad (37)$$

Usando  $\mathbf{V}_j^T = \mathbf{W}_j^T - \mathbf{W}_*^T$  e  $\mathbf{V}_j^T \mathbf{X}_j = \mathbf{X}_j^T \mathbf{V}_j$ , é possível reescrever a Equação 37, como segue:

$$\begin{aligned} \varepsilon_j &= n_j - \mathbf{V}_j^T \mathbf{X}_j \\ &= n_j - \mathbf{X}_j^T \mathbf{V}_j \end{aligned} \quad (38)$$

Retomando a equação de atualização do algoritmo SAG:

$$\mathbf{W}_{j+1} = \mathbf{W}_j + 2\mu\alpha\varepsilon_j e^{-\alpha\varepsilon_j} \mathbf{X}_j$$

$$\mathbf{W}_{j+1} - \mathbf{W}_* = \mathbf{W}_j + 2\mu\alpha\varepsilon_j e^{-\alpha\varepsilon_j} \mathbf{X}_j - \mathbf{W}_* \quad (39)$$

E substituindo-se  $\mathbf{V}_j = \mathbf{W}_j - \mathbf{W}_*$  na Expressão 39, resulta:

$$\mathbf{V}_{j+1} = \mathbf{V}_j + 2\mu\alpha\varepsilon_j e^{-\alpha\varepsilon_j^2} \mathbf{X}_j \quad (40)$$

Utilizando-se a expansão de Taylor da expressão  $\varepsilon_j e^{-\alpha\varepsilon_j^2}$  e substituindo a Equação 38, tem-se:

$$\begin{aligned} \varepsilon_j e^{-\alpha\varepsilon_j^2} &= \sum_{i=1}^{\infty} a_i \varepsilon_j^{2i-1} \\ &= \sum_{i=1}^{\infty} a_i (n_j - \mathbf{X}_j^T \mathbf{V}_j)^{2i-1} \end{aligned} \quad (41)$$

Fazendo a expansão binomial da expressão entre parênteses, tem-se:

$$(n_j - \mathbf{V}_j^T \mathbf{X}_j)^{2j-1} = \sum_{p=1}^{2i-1} \binom{2i-1}{p} n_j^p (-\mathbf{V}_j^T \mathbf{X}_j)^{2i-1-p} \quad (42)$$

Negligenciando as altas potências de  $\mathbf{V}_j$ , tem-se:

$$(n_j - \mathbf{V}_j^T \mathbf{X}_j)^{2j} = n_j^{2i-1} - (2i-1)n_j^{2i-2} \mathbf{V}_j^T \mathbf{X}_j \quad (43)$$

Substituindo a Equação 43 em 41, resulta:

$$\varepsilon_j e^{-\alpha \varepsilon_j^2} = \sum_{i=1}^{\infty} a_i [n_j^{2i-1} - (2i-1)n_j^{2i-2} \mathbf{V}_j^T \mathbf{X}_j] \quad (44)$$

Usando a Equação 44 em 40, podemos escrever:

$$\mathbf{V}_{j+1} \approx \mathbf{V}_j + 2\mu\alpha \sum_{i=1}^{\infty} a_i [n_j^{2i-1} - (2i-1)n_j^{2i-2} \mathbf{V}_j^T \mathbf{X}_j] \mathbf{X}_j \quad (45)$$

Realizando algumas manipulações matemáticas na Equação 45, obtem-se:

$$\mathbf{V}_{j+1} \approx \mathbf{V}_j + 2\mu\alpha \sum_{i=1}^{\infty} [a_i n_j^{2i-1} \mathbf{X}_j] - 2\mu\alpha \sum_{i=1}^{\infty} [a_i (2i-1) n_j^{2i-2}] \mathbf{V}_j^T \mathbf{X}_j \mathbf{X}_j^T \quad (46)$$

Assumindo que  $\mathbf{X}_j$  e  $s_j$  são mutuamente independentes e aplicando o operador esperança em ambos os lados da Equação 46, tem-se:

$$E\{\mathbf{V}_{j+1}\} \approx E\{\mathbf{V}_j\} + 2\mu\alpha \sum_{i=1}^{\infty} [a_i E\{n_j^{2i-1} \mathbf{X}_j\}] - 2\mu\alpha \sum_{i=1}^{\infty} [a_i (2i-1) E\{n_j^{2i-2}\}] E\{\mathbf{X}_j \mathbf{X}_j^T\} E\{\mathbf{V}_j^T\} \quad (47)$$

$$E\{\mathbf{V}_{j+1}\} \approx E\{\mathbf{V}_j\} + 2\mu\alpha \sum_{i=1}^{\infty} [a_i E\{n_j^{2i-1} \mathbf{X}_j\}] - 2\mu\alpha \sum_{i=1}^{\infty} [a_i (2i-1) E\{n_j^{2i-2}\}] \mathbf{R} E\{\mathbf{V}_j^T\} \quad (48)$$

sendo  $\mathbf{R} = E\{\mathbf{X}_j \mathbf{X}_j^T\}$ .

$$E\{\mathbf{V}_{j+1}\} \approx \left[ I - 2\mu\alpha \sum_{i=1}^{\infty} [a_i (2i-1) E\{n_j^{2i-2}\}] \mathbf{R} \right] E\{\mathbf{V}_j\} \quad (49)$$

Note que:

$$\sum_{i=1}^{\infty} [a_i (2i-1) E\{n_j^{2i-2}\}] = E \left\{ \frac{d}{dn_j} (n_j e^{-\alpha n_j^2}) \right\} = E\{[1 - 2\alpha n_j^2] e^{-\alpha n_j^2}\} \quad (50)$$

Substituindo a Equação 50 em 49, pode-se reescrever:

$$E\{\mathbf{V}_{j+1}\} \approx \left[ I - 2\mu\alpha E\{[1 - 2\alpha n_j^2] e^{-\alpha n_j^2}\} \mathbf{R} \right] E\{\mathbf{V}_j\} \quad (51)$$

A partir da Equação 51 obtem-se a condição de convergência, dada por:

$$0 < \mu < \frac{1}{2\alpha[1 - 2\alpha n_j^2]e^{-\alpha n_j^2}} \quad (52)$$

### 3.4 Desajuste

Na subseção 2.6.2.3, definimos o desajuste para a superfície quadrática. Devido a dificuldade de se derivarem expressões para medir o desajuste de não linearidades, Douglas e Meng (DOUGLAS; MENG, 1994), propõem uma aproximação, a qual utilizaremos aqui, dada por:

$$M = \frac{E\{f^2(n_j)\}tr[\mathbf{R}]}{2E\{f'(n_j)\}E\{n_j^2\}} \quad (53)$$

Dessa forma o desajuste do algoritmo SAG, cuja função de custo utilizada é  $F = 1 - e^{-\alpha \varepsilon_j^2}$ , será:

$$M = \frac{E\{4\alpha^2 n_j^2 e^{-2\alpha n_j^2}\}tr[\mathbf{R}]}{2E\{[2\alpha - 4\alpha^2 n_j^2]e^{-\alpha n_j^2}\}E\{n_j^2\}} \quad (54)$$

onde:

- $f(n_j) = 2\alpha n_j e^{-\alpha n_j^2}$
- $f^2(n_j) = 4\alpha^2 n_j^2 e^{-2\alpha n_j^2}$
- $f'(n_j) = [2\alpha - 4\alpha^2 n_j^2]e^{-\alpha n_j^2}$

## 4 Simulações e resultados

Nesta seção apresenta-se uma análise comparativa dos desempenhos dos algoritmos LMS e SAG, através das suas curvas de aprendizagem. Posteriormente será realizado a aplicação do algoritmo em eliminação de ruído em sinal de ECG.

### 4.1 Análise do desempenho de algoritmos adaptativos

O desempenho de um filtro adaptativo é diretamente dependente do algoritmo adaptativo utilizado no filtro. Algumas características do processo de adaptação são de grande utilidade para a análise e o entendimento do comportamento dos algoritmos adaptativos. A mais importante delas é a curva de aprendizado, que caracteriza o desempenho de um algoritmo adaptativo e é amplamente utilizada em estudos teóricos e experimentais (BORN, 2000). Tal curva relaciona o Erro Médio Quadrático (EMQ) em função do tempo.

A curva de aprendizagem possui dois estágios, chamados adaptação e regime permanente. Quando o filtro adaptativo inicia sua operação, os seus coeficientes podem possuir valores bem diferentes de  $\mathbf{W}_*$  (vetor de coeficientes ótimo), causando um erro grande. Durante a execução do algoritmo, os coeficientes vão se aproximando de  $\mathbf{W}_*$ , causando por consequência, uma diminuição do erro, esta fase é chamada de adaptação. Após esta fase o filtro entra no modo de regime permanente.

Os critérios de desempenho mais importantes de um algoritmo de filtragem adaptativa são a velocidade de convergência (ou taxa de convergência) e o desajuste. A velocidade de convergência descreve o comportamento transitório do algoritmo no estágio de adaptação e é definida como o número de iterações requerida pelo algoritmo para convergir até uma solução ótima. O desajuste descreve o comportamento do algoritmo no estágio de regime permanente e refere-se a uma medida quantitativa pelo qual o valor da média final total do Erro Médio Quadrático excede o mínimo Erro Médio Quadrático produzido pelo filtro ótimo de Wiener. A duração do modo de transição caracteriza a velocidade de adaptação do filtro adaptativo e o regime permanente caracteriza a qualidade da adaptação. Quanto menor for o erro em regime permanente melhor terá sido o cancelamento do ruído (BORN, 2000). Na Figura 17 podemos observar os estágios de uma curva de aprendizagem de um algoritmo adaptativo.

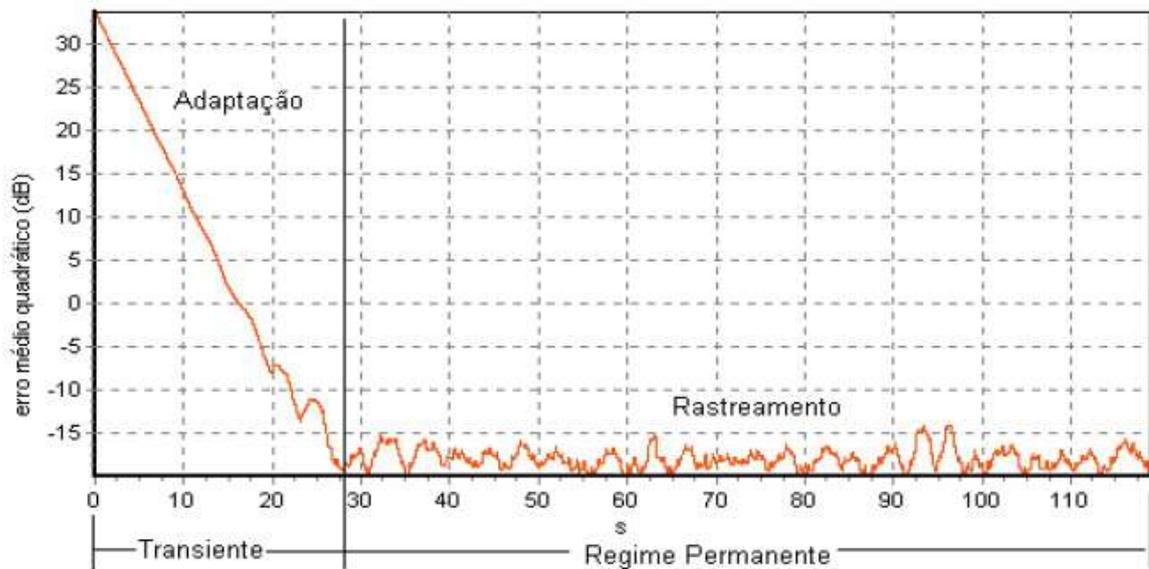


Figura 17 – Curva de aprendizagem de um algoritmo adaptativo

#### 4.1.1 Simulações computacionais SAG $\times$ LMS

Para fazer a análise comparativa entre os algoritmos SAG e LMS foram realizadas 100 simulações de Monte Carlo, para se obter a curva de convergência do Erro Médio Quadrático (EMQ) dos algoritmos. Como uma forma de comparação da efetividade dos algoritmos, utilizou-se um ruído gaussiano como sinal de entrada. O passo de adaptação adotado foi de  $\mu = 10^{-2}$  para ambos os algoritmos, sendo a escolha do  $\mu$  justificada pelo fato de obter-se os melhores resultados para o algoritmo LMS, e assim observar o ganho obtido pelo algoritmo proposto quando comparado com os melhores resultados do LMS. Utiliza-se ainda  $\alpha = 2$  para o algoritmo SAG.

Na Figura 18a é apresentado um gráfico de comparação entre o comportamento dos algoritmos SAG e do LMS. É possível perceber que ambos os algoritmos convergem para o EMQ mínimo, mas com diferentes taxas de convergência para a mesma situação. No LMS, é possível perceber a necessidade de um tempo maior de adaptação até convergir para o regime permanente, necessitando de cerca de 700 iterações para convergir enquanto que o SAG converge com 400 iterações, tendo os dois algoritmos praticamente o mesmo erro em regime permanente, o que pode ser observado na Figura 18b.

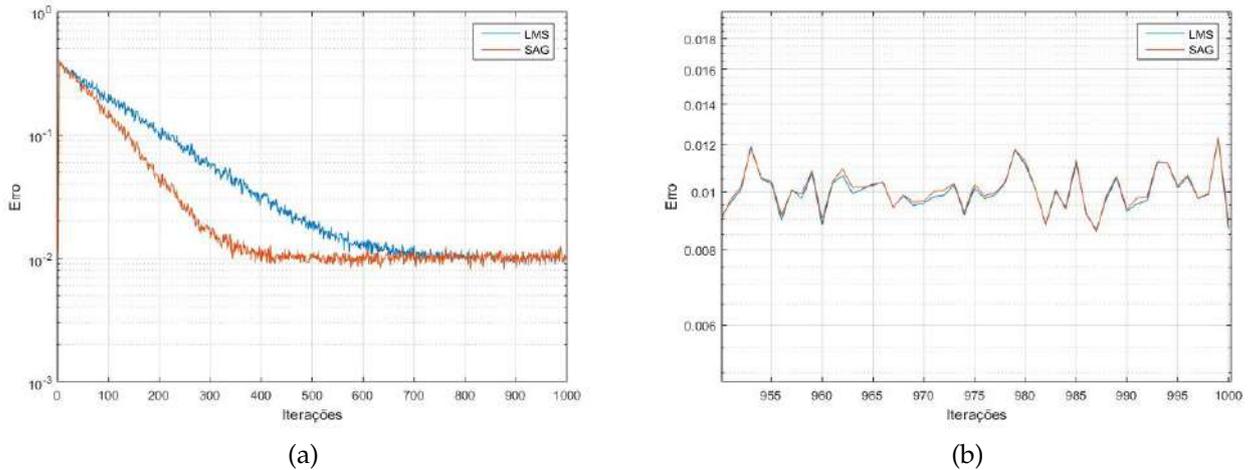


Figura 18 – Curvas de aprendizagem dos algoritmos LMS e SAG com  $\alpha = 2$

Destaca-se ainda que o algoritmo SAG pode ter sua velocidade de convergência melhorada com o aumento do valor de  $\alpha$ , o que pode ser observado na Figura 19.

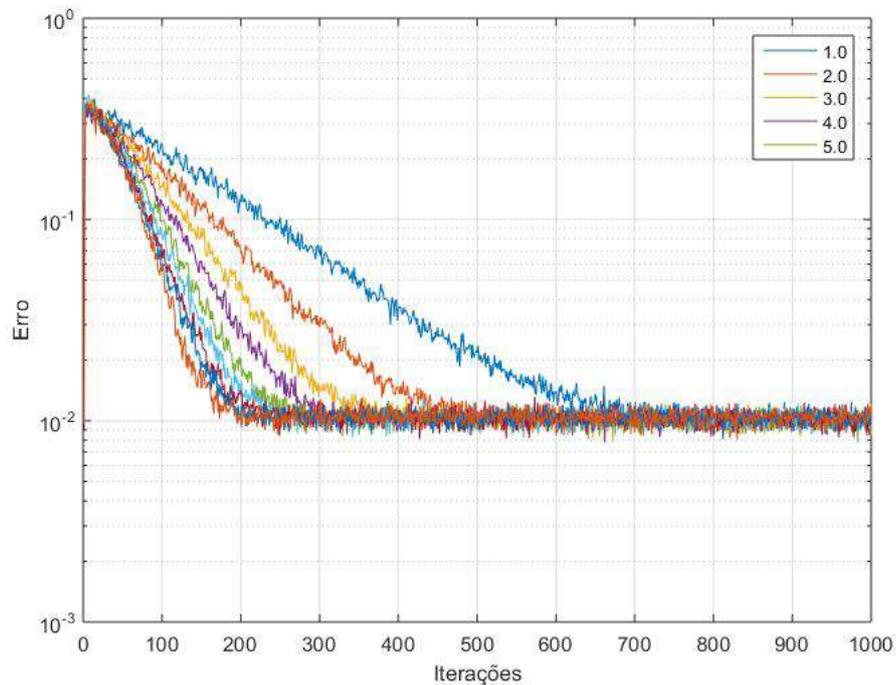


Figura 19 – Curvas de aprendizagem do algoritmo SAG com  $\alpha = 1, \alpha = 2, \alpha = 3, \alpha = 4$  e  $\alpha = 5$

Porém, em contra partida, aumentando-se o parâmetro  $\alpha$  do algoritmo SAG obteve-se um aumento no erro em regime permante quando comparado ao LMS, o que pode ser observado na Figura 20b. Utilizando-se  $\alpha = 4$ , a convergencia do algoritmo SAG ocorre em cerca de 200 iterações, conforme Figura 20a.

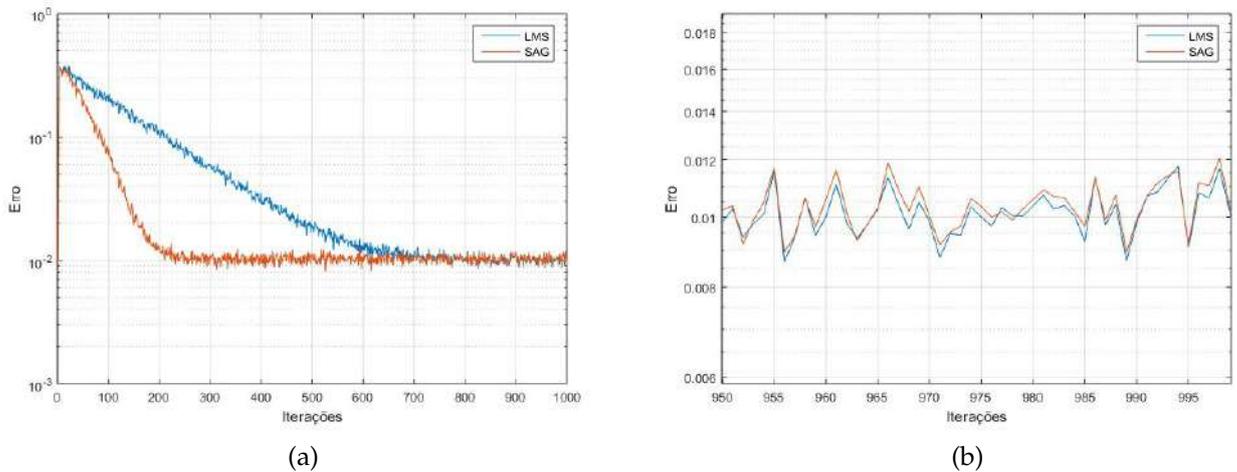


Figura 20 – Curvas de aprendizagem dos algoritmos LMS e SAG com  $\alpha = 4$

Na Figura 21a, para  $\alpha = 10$ , o algoritmo SAG converge em aproximadamente 100 iterações, contudo com um erro final ainda maior em relação aos casos anteriores.

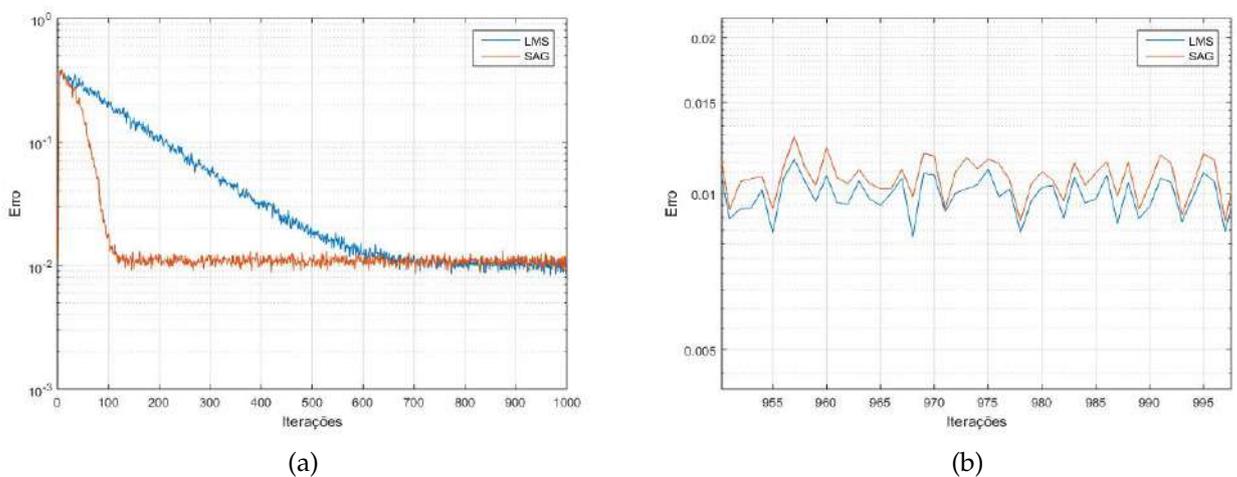


Figura 21 – Curvas de aprendizagem dos algoritmos LMS e SAG com  $\alpha = 10$

## 4.2 Eliminação de ruído em sinal de Eletrocardiograma (ECG)

O Eletrocardiograma, comumente conhecido como ECG, é, sem dúvida, o exame mais empregado em cardiologia, destacando-se por ser rápido, barato e não invasivo (GAWANDE, 2002). Além disso, a disponibilidade do ECG fornece ao médico importantes apontamentos para análises e diagnósticos de anomalias cardíacas. O objetivo do eletrocardiograma é registrar a variação dos potenciais elétricos gerados pela atividade

elétrica do coração, captados por eletrodos dispostos em locais específicos do corpo humano, a qual representa uma sequência de batimentos cardíacos. O sinal de ECG padrão é formado por segmentos consecutivos no tempo, que corresponde a um batimento cardíaco, sendo o mesmo constituído por formas elementares (ou ondas elementares), que se repetem ao longo do tempo. O estudo das amplitudes e da morfologia destas ondas constitui a base da análise do sinal ECG (OLVEIRA, 2010).

A Figura 22 representa a forma de onda característica de um sinal de ECG, considerando condições normais de funcionamento do coração e o processo de aquisição do mesmo.

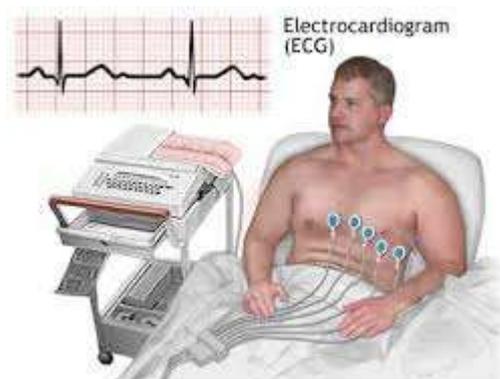


Figura 22 – Aquisição de sinal de ECG

Devido a baixa amplitude do sinal de ECG, por volta de 1mVpp, o mesmo torna-se muito suscetível a interferências, podendo ser contaminado por diversos ruídos durante sua aquisição. Esses ruídos fazem com que a interpretação dos sinais se torne mais difícil (MASSAKI, 2007). Para sinais de ECG a descontaminação se faz necessária pela nitidez e clareza que os diagnósticos exigem, já que a caracterização de cada cardiopatia se manifesta em modificações específicas da forma de onda do sinal (ressalvando-se que algumas anomalias cardíacas não alteram o ECG). Os principais tipos de ruídos, usualmente encontrados, são (D'ALMEIDA et al., 2014):

- Interferência da rede elétrica: ruídos aditivos nos sinais de ECG a partir da indução de sinais senoidais, na frequência de 60 Hz, nos cabos ligados aos eletrodos e à entrada de aquisição de sinais.
- Eletromiogramas (EMG): são distúrbios causados por contrações musculares. Artefatos de movimento: provenientes da movimentação dos eletrodos sob a pele do paciente, usualmente, introduzindo variações bruscas no sinal de ECG, cujas componentes ocupam ampla faixa de frequência.
- Baseline Wander: ruídos provenientes de diversos fatores como a transpiração, respiração, movimento musculares, dentre outros. Esses tipos de ruídos incluem

componentes de baixa frequência, em torno de 0,5 Hz.

A solução para minimizar os efeitos causados por tais interferências, é a utilização do processo de filtragem adaptativa, costumeiramente presente nas técnicas de processamento digital de sinais.

### 4.3 Eliminação de ruído Gaussiano em sinal de ECG por meio do algoritmo SAG

Nesta seção pretende-se mostrar o processo de filtragem adaptativa utilizando-se o algoritmo SAG em eliminação de ruído Gaussiano em um sinal de ECG. Tal processo é representado na Figura 23.

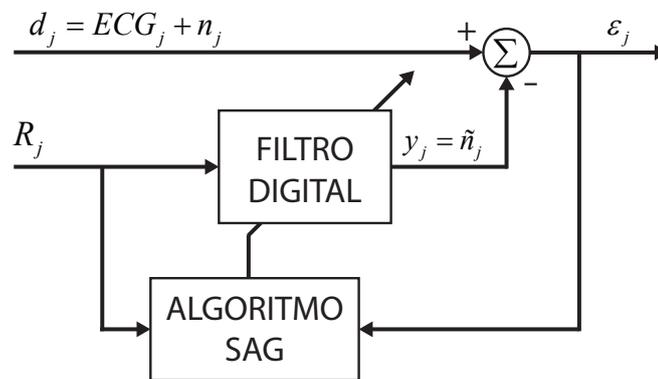


Figura 23 – Diagrama de blocos de um cancelador de ruído adaptativo em sinal de ECG com o algoritmo SAG

Na Figura 23, o sinal de ECG corrompido por um ruído Gaussiano,  $n_j$ , é utilizado como sinal desejado do filtro. O sistema capta uma amostra desse ruído,  $R_j$ , que é utilizada como entrada do filtro, essa amostra é processada pelo filtro, produzindo uma saída,  $\tilde{n}_j$ , o mais parecido possível com o ruído  $n_j$  que contamina o sinal de ECG. O ruído estimado,  $\tilde{n}_j$ , é então subtraído do sinal contaminado,  $d_j$ , gerando um erro  $\varepsilon_j$ , dado por:

$$\begin{aligned}\varepsilon_j &= d_j - \tilde{n}_j \\ &= ECG_j + n_j - \tilde{n}_j\end{aligned}\quad (55)$$

O sinal de erro  $\varepsilon_j$  é utilizado pelo algoritmo SAG para ajustar os pesos do filtro. Quanto melhor for a estimativa do ruído,  $\tilde{n}_j$ , mais parecido com o sinal de ECG será do  $\varepsilon_j$ . Na Figura 24, é apresentado o fluxograma do algoritmo SAG para o problema em estudo.

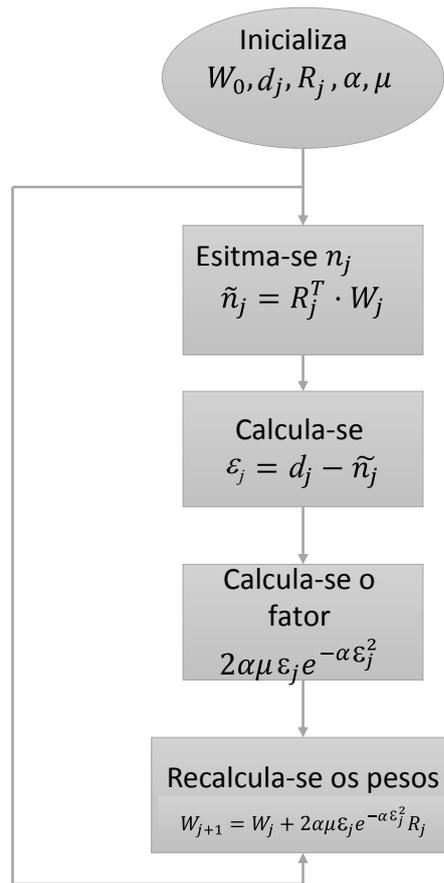


Figura 24 – Fluxograma da algoritmo SAG

O algoritmo é inicializado com  $W_0$ ,  $R_j$ ,  $d_j$ ,  $\alpha$  e  $\mu$ , sendo  $R_j$  composta por  $r_j$  atual e pelas  $N - 1$  amostras anteriores, conforme Expressão 56. Normalmente inicia-se  $W_0$  com zeros. É então encontrado o ruído estimado,  $\tilde{n}_j$ , e o sinal  $\varepsilon_j$ . A partir destes valores é então encontrado o valor de  $\mathbf{W}$  no instante seguinte e assim sucessivamente.

$$R_j = \begin{bmatrix} r_j \\ r_{j-1} \\ \vdots \\ r_{j-N+1} \end{bmatrix} \quad (56)$$

### 4.3.1 Resultados ECG

Nesta seção, visando comparar o desempenho dos algoritmos LMS e SAG, são apresentados resultados de simulação considerando um problema de cancelamento de ruído gaussiano em um sinal de ECG.

Toda a análise foi realizada por meio de simulação computacional através do software MATLAB, utilizando-se um sinal de ECG gerado artificialmente. Em seguida,

utilizou-se um ruído gaussiano, somando-se o mesmo ao sinal de ECG gerado, fazendo com que esse se torne um sinal contaminado e por fim realizou-se o processo de descontaminação através dos algoritmos LMS e SAG.

A seguir, na Figura 25, são apresentados o sinal de ECG original e o sinal de ECG contaminado com ruído gaussiano.

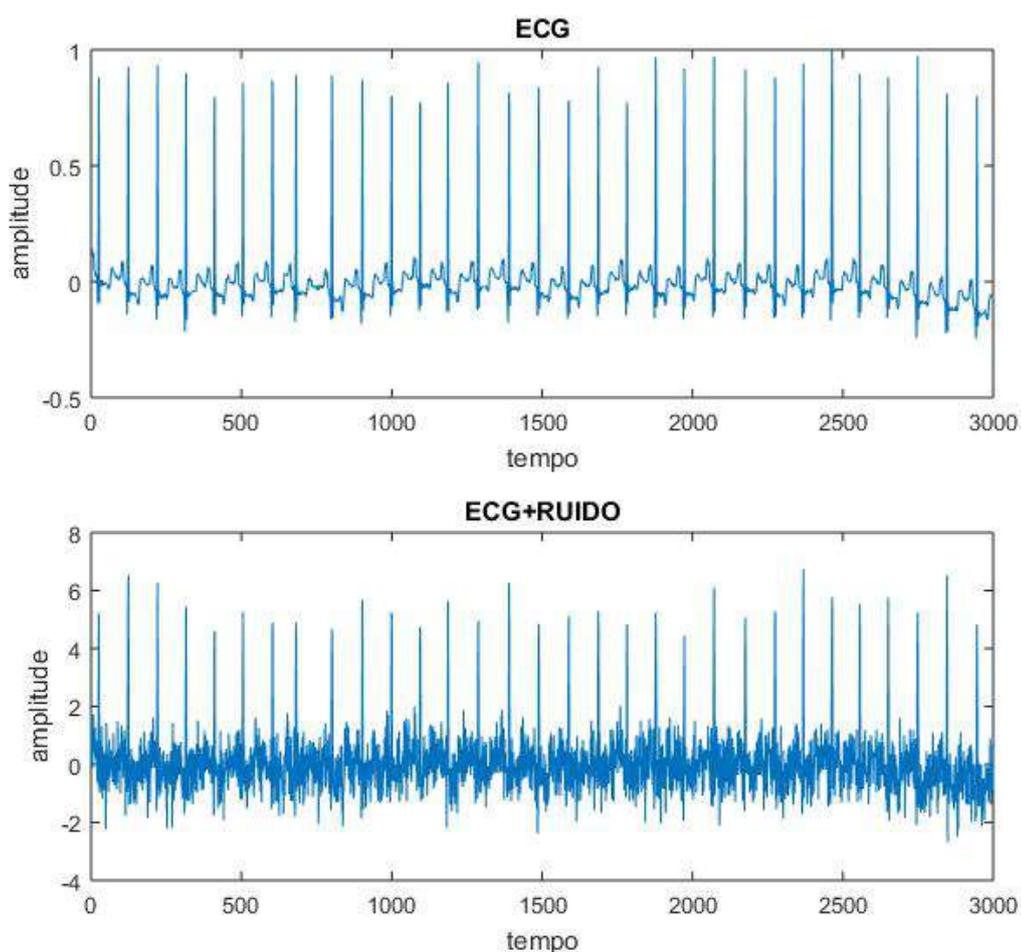


Figura 25 – Sinal de ECG e sinal de ECG com ruído

A Figura 26 mostra o processo de filtragem no sinal de ECG por meio dos algoritmos LMS e SAG. Para realizar tal processo, assumem-se os mesmos parâmetros utilizados na seção anterior.

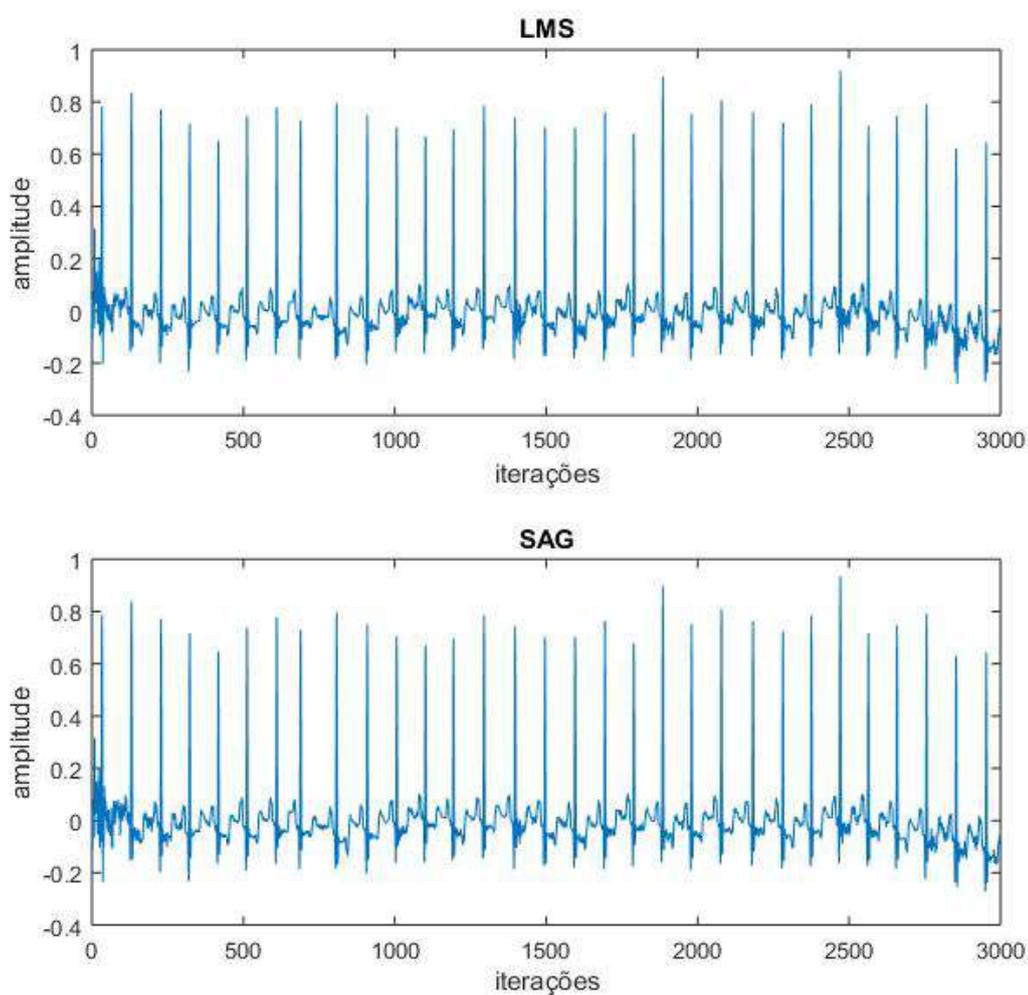


Figura 26 – Resultado do processo de filtragem no sinal de ECG com ruído pelos algoritmos LMS e SAG.

A Figura 27 mostra a evolução do EMQ dos algoritmos LMS e SAG. Aqui, constata-se que o algoritmo SAG apresenta um EMQ em regime permanente menor do que o algoritmo LMS para a aplicação em estudo.

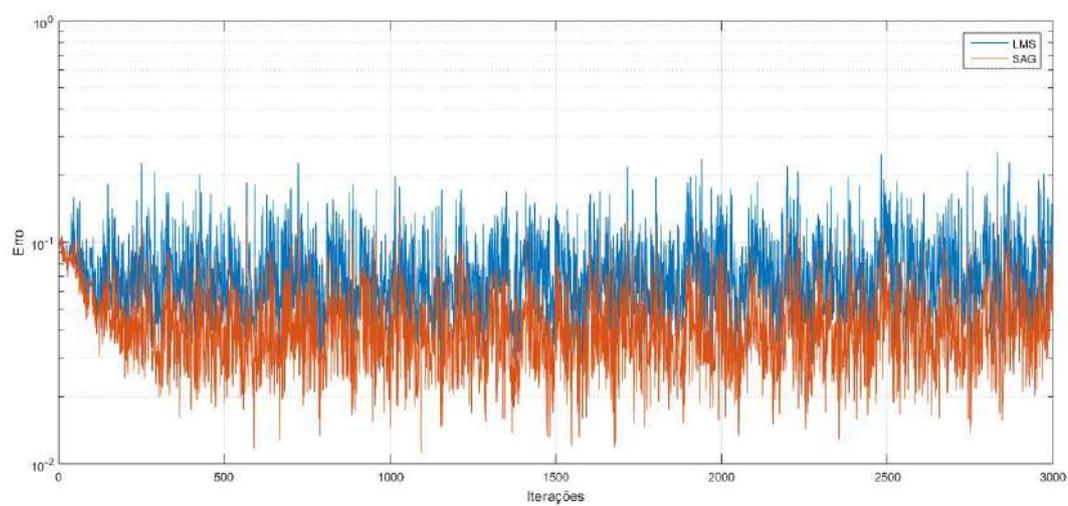


Figura 27 – Resultado do processo de filtragem no sinal de ECG pelo algoritmo SAG.

## 5 Conclusão e Trabalhos futuros

### 5.1 Conclusões

O estudo de filtragem adaptativa tem se tornado relevante nos últimos anos, visto que possui uma gama de aplicações práticas em problemas de comunicações, controle, radar, sonar, sismologia e engenharia biomédica. Os filtros adaptativos utilizam algoritmos adaptativos para ajustar seus coeficientes de forma a acompanhar as alterações dos sinais de entrada, sendo o algoritmo LMS convencional o mais utilizado em aplicações de filtragem adaptativa, pois além de fácil implementação, possui ótimo desempenho e velocidade de convergência. Com isso, neste trabalho desenvolveu-se a teoria do algoritmo LMS, pois devido a sua grande aplicabilidade, esse algoritmo foi utilizado para comparação de desempenho com o algoritmo proposto.

Neste trabalho, desenvolveu-se uma família de algoritmos que utiliza uma função não linear como critério aplicado sobre o erro. O algoritmo proposto nomeado de SAG, possui superfície de desempenho mais íngreme que o algoritmo LMS, o que indica que sua velocidade de convergência é maior. Além disso, a função proposta possui apenas um ótimo global, o que facilita a busca do ponto ótimo. Ao decorrer do trabalho, foram desenvolvidos e apresentadas equações matemáticas relacionadas ao passo de adaptação que garante a convergência do algoritmo proposto, assim como a equação do desajuste.

Realizou-se a aplicação do algoritmo proposto com foco em apenas uma das classes de problemas relacionados a filtragem adaptativa: o cancelamento de ruído. Assim, como forma de analisar o desempenho, utilizando-se uma base de dados com sinais de ECG, aplicou-se um ruído gaussiano, com o propósito de realizar a eliminação desse ruído com o filtro adaptativo, utilizando o algoritmo SAG. Os resultados mostraram que o algoritmo proposto foi superior ao LMS, tanto em termos de convergência quanto em termos de erro no processo de adaptação.

Os resultados mostraram que o algoritmo possui boas características de desempenho e robustez, tornando-o atrativo para utilização em filtragem adaptativa.

### 5.2 Trabalhos futuros

Como proposta para continuação do trabalho, algumas melhorias que podem ser feitas no algoritmo desenvolvido e propostas de trabalhos futuros são listadas a seguir:

- Desenvolvimento de equações mais adequadas para o desajuste;
- Estudos mais aprofundados sobre o parâmetro  $\alpha$ , visto que este parâmetro é definido pelo especialista manualmente. Desse modo, pretende-se desenvolver uma metodologia para que seu valor possa ser calculado de maneira automática para um dado problema, ou seja, o  $\alpha$  específico para aquela aplicação iria ser obtido através de um processo iterativo;
- As aplicações realizadas neste trabalho continham apenas sinais normais de ECG com ruído, e desse modo pode-se obter com bastante precisão, a recuperação da informação original do sinal. No entanto, em propostas futuras pretende-se analisar seu comportamento em sinais de ECG que contenham algum tipo de anormalidade;
- Verificar o comportamento do algoritmo proposto em sinais de ECG corrompidos por outros tipos de ruídos;
- Aplicar o algoritmo em sinais de Eletroencefalograma (EEG) corrompido com ruídos;
- Realizar aplicações e verificações de desempenho do algoritmo proposto em uma gama de problemas de engenharia.

## Referências

- AGOSTINHO, P. **Otimização e aperfeiçoamento de BAC para estudos envolvendo o trato gastrointestinal**. Dissertação (Mestrado) — Universidade Estadual Paulista, Botucatu, 2011.
- ALMEIDA, S. J. M. D. **Análise estatística do comportamento de uma classe de algoritmos de projeções afins**. Tese (Doutorado) — UFSC, 2004.
- BARROS, A. K.; PRINCIPE, J.; TAKEUCHI, Y.; SALES, C. H.; OHNISHI, N. An algorithm based on the even moments of the error. In: IEEE. **Neural Networks for Signal Processing, 2003. NNSP'03. 2003 IEEE 13th Workshop on**. [S.l.], 2003. p. 879–885.
- BATISTA, E. **Filtros Volterra adaptativos: Estruturas interpoladas e modelos estocásticos**. Tese (Tese de Doutorado) — Universidade Federal de Santa Catarina, Florianópolis, 2009.
- BORN, R. S. Filtros adaptativos aplicados a sinais biomédicos. **Monografia, Universidade Federal de Pelotas**, 2000.
- CASTOLDI, F. T. **Desenvolvimento de algoritmos adaptativos usando a descrição de distúrbio mínimo com restrições**. Dissertação (Dissertação de Mestrado) — UFRJ, 2007.
- CORDEIRO, P. **Filtragem adaptativa aplicada em cancelamento de ruído acústico**. Dissertação (Dissertação de Mestrado) — Universidade Federal do Pará, Belém, 2008.
- D'ALMEIDA, A.; GUEDES, L.; CARVALHO, M.; PONZIO, P.; FONSECA, W.; PEIXOTO, Z. Interface multitarefa parametrizável para o tratamento digital de sinais de ecg. In: CBEB. **XXIV Congresso Brasileiro de Engenharia Biomédica**. [S.l.], 2014. p. 2707–2711.
- DINIZ, P. S. **Adaptive filtering**. [S.l.]: Springer, 1997.
- DOUGLAS, S. C.; MENG, T. H. Stochastic gradient adaptation under general error criteria. **Signal Processing, IEEE Transactions on, IEEE**, v. 42, n. 6, p. 1335–1351, 1994.
- EWEDA, E. Comparison of rls, lms, and sign algorithms for tracking randomly time-varying channels. **Signal Processing, IEEE Transactions on, IEEE**, v. 42, n. 11, p. 2937–2944, 1994.
- FARHANG-BOROJENY, B. **Adaptive filters: theory and applications**. [S.l.]: John Wiley & Sons, 2013.
- GAWANDE, A. **Complicações: Dilemas de um cirurgião diante de uma ciência imperfeita**. [S.l.]: Objetiva, 2002.
- GONTIJO, W. A. **Implementação e avaliação de desempenho de algoritmos adaptativos aplicados em controle ativo de vibrações**. Dissertação (Mestrado) — UFSC, 2006.
- HAYKIN, S. **Adaptive Filter Theory**. [S.l.]: Prentice Hall, 2001.
- MALUENDA, Y. R. M.; BERMUDEZ, J. C. M.; NASCIMENTO, V. H. Propriedades do algoritmo lms operando em precisão finita. 2005.

- MASSAKI, O. I. **Utilização de filtros para remoção de interferência de sinais de eletrocardiograma**. Dissertação (Monografia) — Universidade Presbiteriana Mackenzie, São Paulo, 2007.
- NAHUZ, C. S. **Algoritmo adaptativo tipo-LMS com soma do erro**. Dissertação (Dissertação de mestrado) — Universidade Federal do Maranhão, São Luís, 2016.
- NASCIMENTO, V. H. **Análise de algoritmos para filtragem adaptativa baseados em momentos de quarta ordem**. Tese (Doutorado) — USP, 2005.
- NUNES, W. R. B. M. Estudo e implementação de algoritmos com filtragem adaptativa aplicáveis a medição de sinais ruidosos em movimentos periódicos. **Revista de Tecnologia e Ciências do SENAI**, 2014.
- OLVEIRA, L. S. C. de. **Redes Bayesianas dinâmicas com definição de limiar aplicadas ao estudo de caso detecção de extrassístole ventricular**. Tese (Tese de Doutorado) — Universidade Federal do Rio de Janeiro, 2010.
- REGALIA, P. A. Numerical stability issues in fast least-squares adaptation algorithms. **Optical Engineering**, International Society for Optics and Photonics, v. 31, n. 6, p. 1144–1152, 1992.
- SANTANA, E.; PRINCIPE, J.; BARROS, A. K.; FREIRE, R. An adaptive algorithm based on the sigmoidal function. In: IEEE. **Neural Networks, 2006. SBRN'06. Ninth Brazilian Symposium on**. [S.l.], 2006. p. 1–5.
- SANTANA, E. E. C. **Estudo e desenvolvimento de uma família de algoritmos não lineares para filtragem adaptativa**. Dissertação (Mestrado) — UFMA, 2006.
- SAYED, A. H. **Fundamentals of adaptive filtering**. [S.l.]: John Wiley & Sons, 2003.
- SILVA, C. D.; ARAUJO, M. A. F. D.; SANTANA, E.; BARROS, A. K. A recursive algorithm based on non-linear functions of the error. In: IEEE. **Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE**. [S.l.], 2010. p. 596–599.
- TAKAYAMA, D. K.; PEREIRA, R. V.; CAMARGO, V. L. A. de; ABRÃO, T. Implementação eficiente de filtros adaptativos utilizando a plataforma tms320c6713. **Semina: Ciências Exatas e Tecnológicas**, v. 32, n. 1, p. 115–131, 2011.
- WIDROW, B.; STEARNS, S. D. **Adaptive signal processing**. Englewood, NJ, Prentice-Hall Inc, 1985.
- ZIPF, J. G. F. et al. Classificação, análise estatística e novas estratégias de algoritmos lms de passo variável. Florianópolis, SC, 2011.