

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

EWALDO EDER CARVALHO SANTANA

ESTUDO E DESENVOLVIMENTO DE UMA FAMÍLIA DE
ALGORITMOS NÃO LINEARES PARA FILTRAGEM
ADAPTATIVA

São Luís - MA

2006

EWALDO EDER CARVALHO SANTANA

ESTUDO E DESENVOLVIMENTO DE UMA FAMÍLIA DE
ALGORITMOS NÃO LINEARES PARA FILTRAGEM
ADAPTATIVA

Dissertação apresentada ao Programa de Pós Graduação em Engenharia de Eletricidade da UFMA, como requisito para a obtenção do grau de MESTRE em Engenharia de Eletricidade.

Orientador: Allan Kardec Duailibe Barros Filho

Universidade Federal do Maranhão

São Luís - MA

2006

Santana, Ewaldo

ESTUDO E DESENVOLVIMENTO DE UMA FAMÍLIA DE AL-
GORITMOS NÃO LINEARES PARA FILTRAGEM ADAPTA-
TIVA / Ewaldo Santana - 2006

PG.p

1.Engenharia 2. Filtros Adaptativos.. I.Título.

CDU NNN.NN

EWALDO EDER CARVALHO SANTANA

ESTUDO E DESENVOLVIMENTO DE UMA FAMÍLIA DE
ALGORITMOS NÃO LINEARES PARA FILTRAGEM
ADAPTATIVA

Dissertação apresentada ao Programa de Pós Graduação em Engenharia de Eletricidade da UFMA, como requisito para a obtenção parcial do grau de MESTRE em Engenharia de Eletricidade.

Apresentado em 17 de fevereiro de 2006

BANCA EXAMINADORA

Allan Kardec Duailibe Barros Filho

Universidade Federal do Maranhão

Raimundo Carlos Silvério Freire

Universidade Federal de Campina Grande

Sebastian Yuri Cavalcanti Catunda

Universidade Federal do Maranhão

À Éder Júnior, Arthur e Tiago.

Sem vocês, nada seria possível.

Resumo

Neste trabalho é desenvolvida uma família de algoritmos adaptativos baseados em funções não lineares como critério a ser aplicado sobre o erro, o qual deseja-se minimizar. Tal desenvolvimento baseia-se na utilização de estatísticas de alta ordem para a obtenção de mais informações sobre os sinais envolvidos no processo, com o objetivo de melhorar a performance de um filtro adaptativo.

Derivamos equações, baseadas na expansão em séries de Taylor das funções não lineares, para a obtenção de critérios que garantam a convergência. Também fazemos um estudo da covariância do vetor peso em regime estacionário e determinamos equações que mensurem a constante de tempo do processo adaptativo.

Apresentamos o algoritmo sigmoidal, que utiliza como critério a função $\text{Ln}(\cosh \varepsilon)$. Foram feitas simulações com este algoritmo para validar a teoria apresentada, e também o aplicamos para a obtenção das componentes determinísticas de sinais reais de impedância cardiográfica.

Palavras-chave: algoritmos adaptativos, filtragem adaptativa, impedância cardiográfica

Abstract

In this work we develop a family of adaptive algorithms based on nonlinear functions as a criterion to be applied upon the error, that we want to minimize. Such a development is based upon the use of high order statistics to obtain additional information of the signals involved in the process, intending to enhance the adaptive filtering performance.

We derive equations based upon the Taylor's series expansion of the nonlinear functions in order to obtain criterions that guarantee convergence. We also make a study about the covariance of the weight vector on steady state and determine equations that measure the time constant of the adaptive process.

We present the sigmoidal algorithm that uses the function $\text{Ln}(\cosh \varepsilon)$ as criterion. Simulations of this algorithm are performed to validate the theory and it is also applied to obtain the deterministic components of real impedance cardiographic signals.

Keywords: adaptive algorithms, adaptive filtering, cardiographic impedance

Agradecimentos

Ao professor Allan Kardec Duailibe Barros Filho pela amizade, confiança, orientação e dedicação com que encaminhou este trabalho e por nos ajudar a enxergar mais além do nosso olhar.

À professora Maria da Guia da Silva pela oportunidade, incentivo e crédito.

Ao professor Eugênio Medeiros pela amizade e comentários oportunos.

Aos amigos do PIB: Fausto Lucena, Denner Guilhon, André Borges, Lúcio Campos, Ricardo Robson, Carlos Magno, Glenda Salgado, Ivan Júnior, Jaciani Pereira, Raniere Machado e Ranielma Machado.

À toda a minha família.

Aos irmãos Lucas, Jahamagé e toda sua equipe de seareiros.

Aos amigos do Centro Espírita Humberto de Campos pela compreensão.

À CAPES pela bolsa a mim concedida.

Sumário

Lista de Figuras	8
1 Introdução	10
1.1 Motivações	11
1.2 Organização do texto	12
2 A Superfície Quadrática	13
2.1 Introdução	13
2.2 O Combinador Linear Adaptativo	13
2.3 O algoritmo adaptativo LMS	14
2.3.1 Derivação do algoritmo LMS	15
2.3.2 Convergência do vetor peso	18
2.3.3 Excesso do erro quadrático médio	19
2.4 Conclusão do Capítulo	20
3 Uma família de algoritmos baseados em não linearidades do erro	21
3.1 Introdução	21
3.2 Estatística de Segunda Ordem e Estatística de Alta Ordem	21
3.3 Uma função não linear	22

3.4	Derivação do algoritmo	24
3.5	Convergência do vetor peso	25
3.6	Covariância do vetor peso e desajuste	30
3.7	Comparação com o LMS	33
3.8	Conclusão do Capítulo	34
4	O Algoritmo Sigmoidal	35
4.1	Introdução	35
4.2	A função $\text{Ln}(\cosh \varepsilon)$	35
4.3	Derivação do algoritmo Sigmoidal (SA)	37
4.3.1	Convergência do vetor peso, constante de tempo e Desajuste	38
4.3.2	SA versus LMS	38
4.3.3	Simulações com o Algoritmo Sigmoidal	39
4.4	Conclusões do Capítulo	40
5	Estimação Adaptativa Estocástica de Sinal de Impedância Cardiográfica	43
5.1	Introdução	43
5.2	Simulação	45
5.3	Conclusões do capítulo	48
6	Conclusões e Proposta de Continuidade	50
6.1	Conclusões	50

6.2 Proposta de Continuidade	51
--	----

Referências	52
--------------------	-----------

Lista de Figuras

2.1	Combinador Linear Adaptativo	14
2.2	Porção de uma superfície quadrática tridimensional, juntamente com alguns contornos. O erro quadrático médio está plotado na vertical, w_0 e w_1 variam de -1 a 1	16
2.3	A linha pontilhada representa a curva de aprendizagem do algoritmo LMS e a linha cheia representa uma aproximação exponencial dessa curva. . . .	19
3.1	Gráfico da superfície gerada pela função $\cosh(\varepsilon)$ e algumas curvas de níveis. Os pesos w_0 e w_1 variam de -2 a 2.	23
3.2	Excesso no erro final em relação ao erro mínimo	33
4.1	Porção da superfície tridimensional gerada pela função $\text{Ln}(\cosh \varepsilon)$, juntamente com alguns contornos.	36
4.2	Gráficos das funções $\text{Ln}(\cosh \varepsilon)$, $\text{Ln}(\cosh 2\varepsilon)$ e $\text{Ln}(\cosh 3\varepsilon)$	36
4.3	Gráficos das funções $\text{Ln}(\cosh 2\varepsilon)$ e ε^2 , onde podemos ver a maior inclinação da primeira, no intervalo $[-1; 1]$	37
4.4	Diagrama de blocos da modelagem adaptativa de uma planta	40
4.5	curvas de aprendizagem dos algoritmos LMS e SA com $\alpha = 2$	41
4.6	curvas de aprendizagem dos algoritmos LMS e SA com $\alpha = 3$	41

5.1	Captura de sinais de ICG	44
5.2	Diagrama de bloco do filtro adaptativo. s_k é o sinal determinístico, n_k é o ruído descorrelacionado com s_k . $[x_{1,k} x_{2,k} \cdots x_{2H,K}]^T$ é o vetor de entrada.	45
5.3	a) Sinal de interesse, s_k , uma onda quadrada; b) sinal de ruído, n_k , gaussiano de média zero e variância 1; c) Sinal de interesse mais ruído; d) Saída do filtro com o algoritmo LMS; e) Saída do filtro com o algoritmo SA	47
5.4	curvas de aprendizagens dos algoritmos LMS e SA para o mesmo desajuste e para os tamanhos dos passos dados pela relação (4.8). Vemos que o algoritmo SA converge com mais ou menos 500 iterações enquanto que o LMS converge com 1500 iterações	47
5.5	Comparação espectro-temporal entre o sinal de ICG real com os sinais de saída dos filtros LMS e SA. (a) Sinal de ICG real; (b) e (c) Sinais de saída dos filtros LMS e SA, respectivamente; (d), (e) e (f) Transformada de Fourier de (a), (b) e (c), respectivamente. As setas indicam a remoção do ruído de 1.8 Hz e seus harmônicos realizada pelo algoritmo SA, quando comparado com o sinal original e a saída do LMS	48

1 Introdução

O processamento de sinais utilizados na maioria das situações práticas envolve o processamento de dados contaminados por ruído, de forma a extrair alguma informação sobre o sinal de interesse. Nesta categoria incluem-se os processamentos chamados de filtragem, predição e estimação. Os sinais envolvidos são sinais aleatórios, os quais são caracterizados por suas propriedades estatísticas. O projeto de filtros para o processamento de sinais aleatórios requer o conhecimento prévio de alguma informação sobre as propriedades estatísticas dos sinais envolvidos. Quando isso é possível, trata-se o problema no âmbito do processamento estatístico de sinais. Nos casos em que tais informações não são conhecidas e não podem ser estimadas por falta de tempo (processamento em tempo real), a melhor solução é o emprego de filtros adaptativos. Tais filtros são programáveis por um algoritmo numérico que realiza um processo de otimização de acordo com uma figura de mérito especificada. O trabalho em filtragem adaptativa envolve o estudo de algoritmos e de estruturas de filtragem de forma a melhorar o desempenho dos sistemas adaptativos existentes.

Áreas de aplicação de filtragem adaptativa incluem a identificação de sistemas físicos, o cancelamento de ecos em sistemas de comunicação, a equalização de canais de sistemas de comunicação, o cancelamento de interferências, a codificação de sinais e o controle ativo de ruído acústico e de vibrações. Em especial, na área biomédica, diversas aplicações podem ser encontradas, tais como: Cancelamento de interferências de sinais de 60 Hz em Eletrocardiograma (ECG); cancelamento de interferências do coração-doador no ECG durante o transplante de coração; cancelamento da influência materna em ECG

fetal.

Muito do sucesso dos filtros adaptativos é devido ao desenvolvimento do popular e robusto algoritmo *least-mean-square* (LMS) [1], o qual tenta determinar parâmetros do filtro que minimizem o critério do erro quadrático médio (MSE), o que é válido para o combinador linear adaptativo, descrito no capítulo 2. Este algoritmo é importante em virtude de sua simplicidade e baixo custo computacional, uma vez que não requer estimações do gradiente dos dados com atraso, ou seja, modo *off-line*. Além disso, dado que o sinal de entrada é estatisticamente estacionário, o algoritmo convergirá para a solução de Wiener, na teoria de estimação de mínimos quadrados [2]. Algoritmos baseados no critério do mínimo erro quadrático médio, com características bem conhecidas, são referências para comparações de outros algoritmos.

1.1 Motivações

Devido à complexidade computacional, pouca atenção tem sido dada às **funções não lineares** como estimação do gradiente em filtragem adaptativa. Mas, a exploração das propriedades das funções não lineares pode nos conduzir a importantes descobertas concernentes à melhoria do desempenho da adaptação do algoritmo sob particulares condições estatísticas. Por exemplo, o algoritmo LMS está limitado a um grande comprometimento entre erro médio final e tempo de convergência, fato que tem forçado alguns pesquisadores a apelar a métodos computacionalmente mais custosos [3, 4]. Assim, simplesmente usando funções não lineares como estimativas do gradiente, pode-se obter, de uma forma geral, uma melhora no desempenho do algoritmo.

Neste trabalho, nós apresentamos o desenvolvimento de uma família de algo-

ritmos adaptativos que utilizam como função de custo aplicada sobre o erro, uma não linearidade par, e demonstramos suas propriedades que conduzem a um melhor desempenho quando comparado com outros filtros adaptativos.

1.2 Organização do texto

Este trabalho está dividido da seguinte forma: No capítulo 2, apresentamos o combinador linear adaptativo, como um dos principais elementos dos filtros adaptativos; fazemos uma revisão da superfície quadrática, gerada pela utilização do método dos mínimos quadrados; revisamos o algoritmo *least-mean square* (LMS), mostrando a sua derivação, convergência do vetor peso e excesso final.

No capítulo 3 apresentamos funções lineares como alternativa de critério a ser aplicado sobre o erro; desenvolvemos algoritmos que utilizem-se de função não lineares como estimação do gradiente, bem como analisamos matematicamente a convergência, a covariância do vetor peso bem como o desajuste final e mostramos um modo de comparar esses algoritmos com o algoritmo LMS.

No capítulo 4 derivamos o algoritmo sigmoidal (SA), que utiliza como critério sobre o erro a função $\text{Ln}(\cosh \varepsilon)$. Obtemos expressão para garantir a convergência e determinamos a constante de tempo e o desajuste; fizemos simulações e comparações com o LMS, aplicando as equações desenvolvidas no capítulo 3.

No capítulo 5 apresentamos a Impedância Cardiográfica (ICG), um novo método não invasivo para obtenção de informações cardíacas e aplicamos o algoritmo SA para recuperar as componentes determinísticas do ICG.

2 A Superfície Quadrática

2.1 Introdução

Muitos algoritmos adaptativos utilizam-se do erro quadrático médio como função de custo aplicada sobre o erro e que se deseja minimizar. O erro quadrático médio é uma função convexa dos componentes do vetor peso e gera uma superfície hiperparabolóide que garante a existência de um mínimo global. O problema é como determinar procedimentos de forma tal a encontrar esse mínimo, o mais rápido possível e com o menor erro final.

2.2 O Combinador Linear Adaptativo

O principal componente de muitos sistemas adaptativos é o combinador linear adaptativo (CLA), mostrado na figura (2.1). O sinal de entrada é um vetor, \mathbf{X}_k , definido como

$$\mathbf{X}_k = \begin{bmatrix} x_{0k} & x_{1k} & \cdots & x_{Lk} \end{bmatrix}^T \quad (2.1)$$

O vetor peso, \mathbf{W}_k , é definido por

$$\mathbf{W}_k = [w_{0k} \quad w_{1k} \quad \dots \quad w_{Lk}]^T. \quad (2.2)$$

e a saída, y_k , é igual ao produto interno de \mathbf{X}_k por \mathbf{W}_k :

$$y_k = \mathbf{X}_k^T \mathbf{W}_k = \mathbf{W}_k^T \mathbf{X}_k \quad (2.3)$$

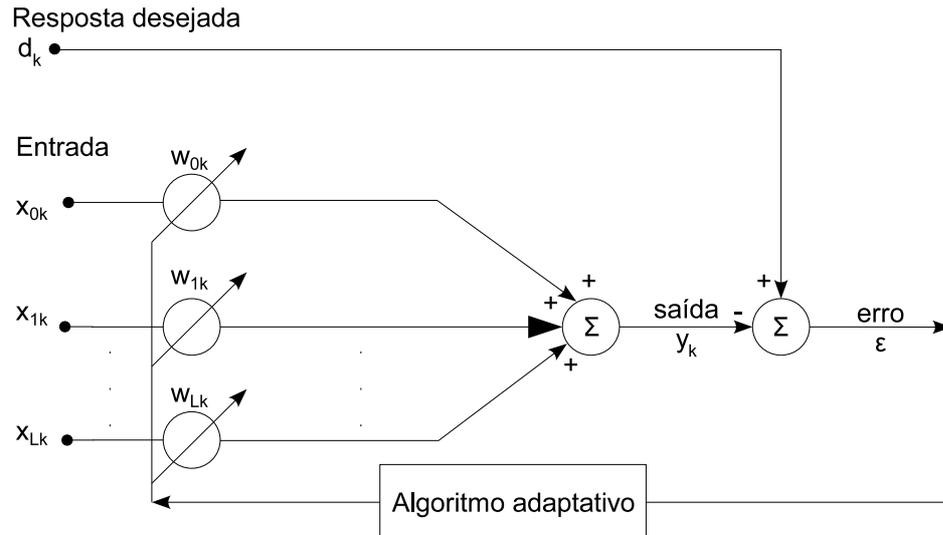


Figura 2.1: Combinador Linear Adaptativo

Conforme visto na figura (2.1), o sinal de erro, no instante k , é dado por

$$\varepsilon_k = d_k - y_k. \quad (2.4)$$

Substituindo (2.3) nesta expressão, temos:

$$\varepsilon_k = d_k - \mathbf{X}_k^T \mathbf{W} = d_k - \mathbf{W}^T \mathbf{X}_k. \quad (2.5)$$

Aqui, nós omitimos o subscrito k no vetor peso por conveniência, visto que, no momento, não queremos ajustar os pesos.

2.3 O algoritmo adaptativo LMS

A finalidade do algoritmo adaptativo mostrado na figura 2.1 é ajustar os pesos do CLA para minimizar o erro quadrático médio. Uma expressão geral para o erro quadrático médio como uma função dos valores dos pesos, supondo que os sinais de entrada e a resposta desejada são estatisticamente estacionários e que os valores do peso são fixos, pode ser derivada da seguinte maneira:

2.3.1 Derivação do algoritmo LMS

Expandindo o quadrado de 2.5, obtemos

$$\varepsilon_k^2 = d_k^2 + \mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W} - 2d_k \mathbf{X}_k^T \mathbf{W}. \quad (2.6)$$

Aplicando o operador expectância em ambos os lados de (2.6), obtemos:

$$E[\varepsilon_k^2] = E[d_k^2] + \mathbf{W}^T E[\mathbf{X}_k \mathbf{X}_k^T] \mathbf{W} - 2E[d_k \mathbf{X}_k^T] \mathbf{W}. \quad (2.7)$$

Definamos \mathbf{R} como a seguinte matriz quadrada:

$$\mathbf{R} = E[\mathbf{X}_k \mathbf{X}_k^T] = E \begin{bmatrix} x_{0k}^2 & x_{0k}x_{1k} & x_{0k}x_{2k} & \dots & x_{0k}x_{Lk} \\ x_{1k}x_{0k} & x_{1k}^2 & x_{1k}x_{2k} & \dots & x_{1k}x_{Lk} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{Lk}x_{0k} & x_{Lk}x_{1k} & x_{Lk}x_{2k} & \dots & x_{Lk}^2 \end{bmatrix} \quad (2.8)$$

Esta matriz é chamada de "matriz de correlação de entrada." Os termos da diagonal principal são as médias quadradas das componentes do sinal de entrada e os demais termos são as correlações cruzadas entre as componentes. Além do mais, ela é uma matriz simétrica, positiva definida e, em raros casos, positiva semidefinida.

Definamos, também, \mathbf{P} como o seguinte vetor:

$$\mathbf{P} = E[d_k \mathbf{X}_k] = [d_k x_{0k} \quad d_k x_{1k} \quad \dots \quad d_k x_{Lk}]^T \quad (2.9)$$

Este vetor é o conjunto das correlações cruzadas entre o sinal resposta desejada e as componentes do sinal de entrada. Quando \mathbf{X}_k e d_k são estacionários, as componentes de \mathbf{R} e \mathbf{P} são todas estatísticas de segunda ordem.

Fazendo $\xi \triangleq E[\varepsilon_k^2]$, e utilizando (2.8) e (2.9) reescrevemos (2.7), da seguinte maneira:

$$\xi \triangleq E[\varepsilon_k^2] = E[d_k^2] + \mathbf{W}^T \mathbf{R} \mathbf{W} - 2\mathbf{P}^T \mathbf{W} \quad (2.10)$$

Observemos que o erro quadrático médio é uma função quadrática dos pesos, cujo gráfico é uma superfície côncava hiperparabólica, conforme vemos na figura 2.2, onde consideramos apenas dois pesos. Esta função, obviamente, nunca pode ser negativa.

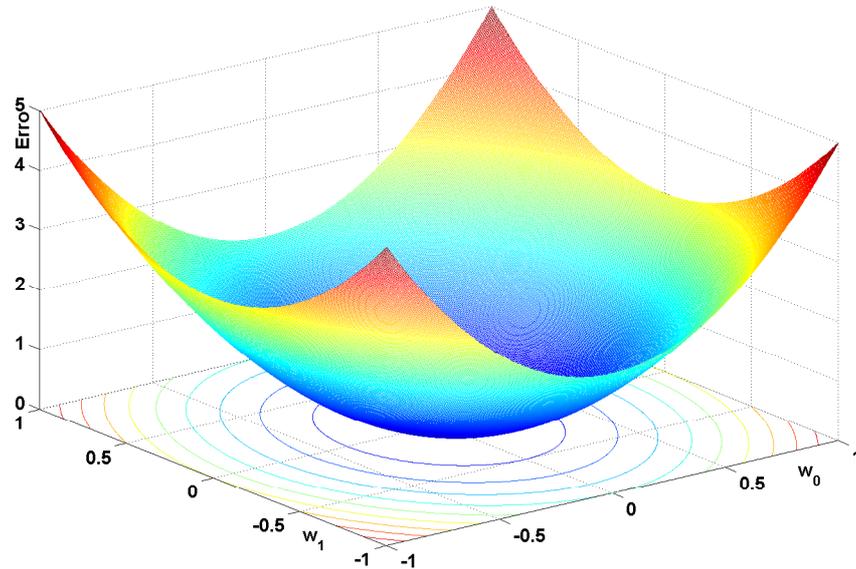


Figura 2.2: Porção de uma superfície quadrática tridimensional, juntamente com alguns contornos. O erro quadrático médio está plotado na vertical, w_0 e w_1 variam de -1 a 1

Ajustar os pesos, para minimizar o erro, significa “descer” sobre a superfície com o objetivo de atingir o ponto mínimo. Métodos do gradiente são geralmente utilizados com este objetivo.

O gradiente da superfície de desempenho do erro quadrático médio, designado por $\nabla(\xi)$, ou simplesmente, ∇ , pode ser obtido derivando (2.10) para obter o vetor coluna

$$\begin{aligned} \nabla &= \frac{\partial \xi}{\partial \mathbf{W}} = \left[\frac{\partial \xi}{\partial w_0} \quad \frac{\partial \xi}{\partial w_1} \quad \dots \quad \frac{\partial \xi}{\partial w_L} \right]^T \\ &= 2\mathbf{R}\mathbf{W} - 2\mathbf{P}. \end{aligned} \tag{2.11}$$

Para obter o erro quadrático médio mínimo, o vetor peso é ajustado para seu

valor ótimo, \mathbf{W}_* , onde o gradiente é zero:

$$\nabla = \mathbf{0} = 2\mathbf{R}\mathbf{W}_* - 2\mathbf{P} \quad (2.12)$$

Supondo que \mathbf{R} seja uma matriz não singular, o vetor peso ótimo, também chamado de vetor peso de Wiener, é determinado de (2.12) como

$$\mathbf{W}_* = \mathbf{R}^{-1}\mathbf{P}. \quad (2.13)$$

O algoritmo adaptativo LMS (*Least Mean Square*) [1], [10], é um método prático para encontrar soluções próximas de (2.13) em tempo real. Este algoritmo é importante em virtude de sua simplicidade computacional, visto que não requer inversões de matriz, nem derivações, nem integrações. A acurácia é limitada pelo tamanho da amostra estatística, visto que os valores dos pesos encontrados são baseados em medidas em tempo real dos sinais de entrada.

O algoritmo LMS é uma implementação do método da descida mais íngreme. De acordo com este método, o “próximo” vetor peso, \mathbf{W}_{k+1} , é igual ao vetor peso “atual”, \mathbf{W}_k , mais um quantidade proporcional ao negativo do gradiente:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla_k. \quad (2.14)$$

O parâmetro μ é um fator que controla a estabilidade e a taxa de convergência, denominada “tamanho do passo”. Cada iteração ocupa um período de tempo unitário.

Para desenvolver o algoritmo LMS, nós tomamos o próprio ε_k^2 como uma estimativa de ξ_k . Então, a cada iteração, no processo adaptativo, nós teremos uma estimação

do gradiente da forma

$$\hat{\nabla}_k = \begin{bmatrix} \frac{\partial \varepsilon_k^2}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k^2}{\partial w_L} \end{bmatrix} = 2\varepsilon_k \begin{bmatrix} \frac{\partial \varepsilon_k}{\partial w_0} \\ \vdots \\ \frac{\partial \varepsilon_k}{\partial w_L} \end{bmatrix} = -2\varepsilon_k \mathbf{X}_k. \quad (2.15)$$

As derivadas de ε_k , em relação aos pesos, seguem, diretamente de (2.5).

Usando (2.15), em lugar do verdadeiro gradiente, em (2.14) temos o algoritmo LMS:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu\varepsilon_k \mathbf{X}_k. \quad (2.16)$$

Este algoritmo é simples e fácil de implementar.

2.3.2 Convergência do vetor peso

A partir de (2.16), é mostrado [1], [10] que o vetor pesos \mathbf{W}_k é função apenas dos vetores de entradas passadas \mathbf{X}_{k-1} , \mathbf{X}_{k-2} , \dots , \mathbf{X}_0 . Se supormos que sucessivos vetores de entrada são independentes no tempo, \mathbf{W}_k será independente de \mathbf{X}_k . Desta forma, o valor esperado do vetor peso, $E[\mathbf{W}_k]$, após um número suficiente de iterações, convergirá para a solução ótima, $\mathbf{W}_* = \mathbf{R}^{-1}\mathbf{P}$. Iniciando com um vetor peso inicial arbitrário, o algoritmo convergirá, em média, e permanecerá estável enquanto o parâmetro μ for maior que zero e menor que o inverso do maior autovalor, λ_{max} , da matriz \mathbf{R} :

$$0 < \mu < \frac{1}{\lambda_{max}}. \quad (2.17)$$

Na figura 2.3 podemos ver uma típica curva de aprendizagem resultante do uso do algoritmo LMS.

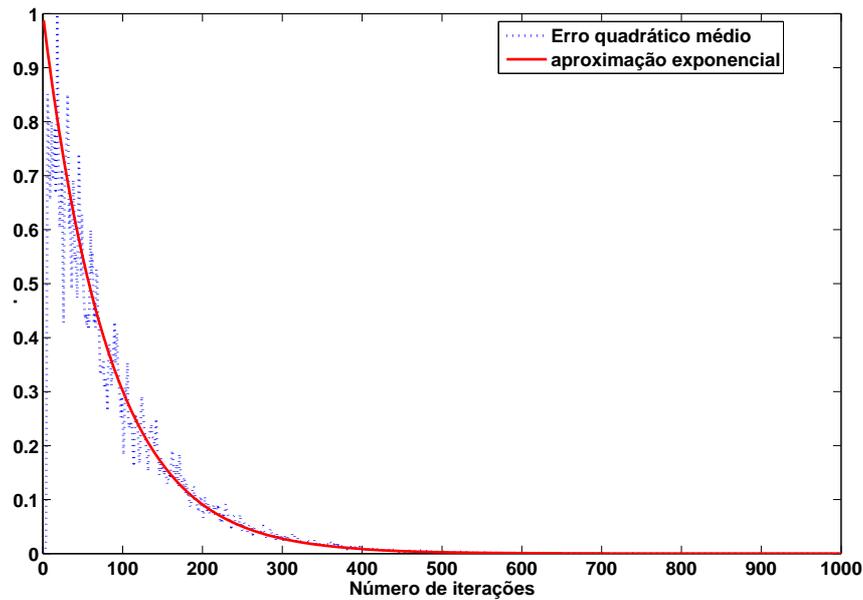


Figura 2.3: A linha pontilhada representa a curva de aprendizagem do algoritmo LMS e a linha cheia representa uma aproximação exponencial dessa curva.

Vemos que esta curva é de natureza exponencial. Podemos, desta forma, aproximá-la por um “envelope” exponencial dado por $e^{-\frac{t}{\tau}}$, onde t é o tempo e τ é uma grandeza chamada de a constante de tempo, de forma tal que uma iteração seja igual à uma unidade de tempo. A constante de tempo está relacionada com o n -ésimo autovalor da matriz \mathbf{R} da seguinte forma:

$$\tau_n = \frac{1}{2\mu\lambda_n}. \quad (2.18)$$

2.3.3 Excesso do erro quadrático médio

Na figura 2.3, podemos ver que quando os pesos não são iguais a \mathbf{W}_* , o erro quadrático médio (ξ) será maior que o erro quadrático médio mínimo (ξ_{min}). Temos, assim, um excesso no erro final

Definimos, então, o excesso do erro quadrático médio, $ExcessoMSE$, como a

diferença entre o erro quadrático médio atual (ξ_k) e o erro quadrático médio mínimo[1]:

$$\begin{aligned} ExcessoMSE &= E[\xi_k - \xi_{min}] \\ &= \mu E[n_k^2] \text{tr}[\mathbf{R}] \end{aligned} \quad (2.19)$$

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

Definimos, também, a diferença entre o erro quadrático médio atual e o erro quadrático médio mínimo, normalizado pelo erro quadrático médio mínimo, como o desajuste (M).

$$M = \frac{E[\xi_k - \xi_{min}]}{\xi_{min}}. \quad (2.20)$$

Desta forma, temos que:

$$M_{LMS} = \mu \text{tr}[\mathbf{R}] \quad (2.21)$$

2.4 Conclusão do Capítulo

Neste capítulo, realizou-se uma revisão da superfície quadrática gerada quando se utiliza o erro quadrático médio (EQM) como critério aplicado sobre o erro em um filtro adaptativo. Mostramos a derivação do popular algoritmo LMS e descrevemos as equações que determinam sua condição de convergência. A constante de tempo e o desajuste também são enfatizados, pois os mesmos são utilizados como referência comparativa de outros algoritmos adaptativos.

3 Uma família de algoritmos baseados em não linearidades do erro

3.1 Introdução

Neste capítulo, mostraremos o desenvolvimento de uma família de algoritmos adaptativos, do tipo descida mais íngreme, que utiliza como estimação instantânea do gradiente uma função ímpar não linear. A idéia básica é mostrar que a superfície de desempenho obtida por este método de estimação oferece maior velocidade de convergência, bem como um menor desajuste na busca do peso mínimo.

3.2 Estatística de Segunda Ordem e Estatística de Alta Ordem

Conforme vimos, entre os filtros adaptativos, o algoritmo LMS aparece como um dos mais utilizados. O LMS pertence a uma classe de algoritmos que pode ser denominada como estatística de segunda ordem (SOS), em oposição à estatística de alta ordem (HOS)[11]. O uso de métodos baseados em SOS é suficiente quando supomos que os sinais envolvidos no processo têm distribuições gaussianas, fornecendo um grande número de simplificações na análise do comportamento do algoritmo, bem como proporcionando um menor custo computacional, em oposição aos métodos baseados em HOS.

Por outro lado, provavelmente devido ao aumento no poder computacional nas últimas décadas, métodos baseados em HOS têm atraído a atenção dos pesquisadores. Certamente, em vez de tratar somente da potência do sinal (i.e. estatística de segunda ordem), HOS permite o acesso à informação contida em todos os momentos do sinal, fornecendo, conseqüentemente, uma melhor aproximação da distribuição real do sinal sob o estudo. Desta forma, podemos esperar que algoritmos projetados sob a ótica de HOS tenham comportamentos mais eficientes.

Neste campo, de particular interesse para o estudo aqui proposto, é o trabalho de Barros et al [11]. Eles 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.

Aqui, nós trabalharemos em uma direção alternativa, propondo como função de custo uma não linearidade par, que gera uma superfície convexa, que não tem mínimo local, apenas um mínimo global e que admita uma expansão em série de Taylor de forma tal a utilizar as informações contidas em todos os momentos do sinal. O resultado é uma família de algoritmos que mostraram-se mais eficazes em termos de velocidade de convergência e desajuste final, quando comparado com o LMS.

3.3 Uma função não linear

Definiremos $F(\varepsilon_k)$ como uma função contínua, não linear, par, simétrica, aplicada sobre o erro a cada iteração. Na figura 3.1, por exemplo, vemos a superfície tridimensional gerada pela função cosseno hiperbólico do erro, considerando-se apenas dois

pesos.

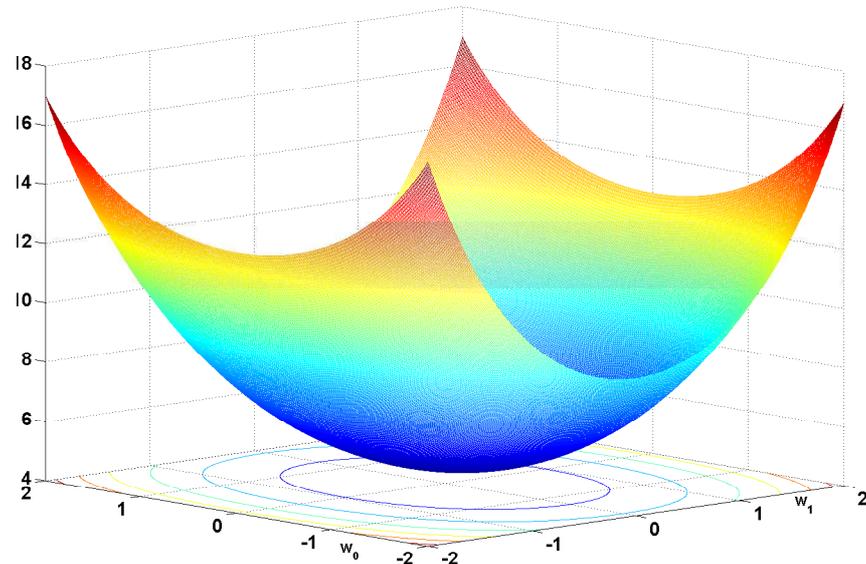


Figura 3.1: Gráfico da superfície gerada pela função $\cosh(\varepsilon)$ e algumas curvas de níveis.

Os pesos w_0 e w_1 variam de -2 a 2.

Sabemos que a forma da superfície depende do critério (função) aplicado sobre o erro. Como os critérios são funções do erro quadrático médio, onde a forma da superfície depende apenas dos sinais de entrada [6], podemos intuir que a forma da superfície, para funções não lineares, dependerá, também, dos sinais de entrada. Além disso, os eixos principais das curvas de níveis da superfície de desempenho, para a função quadrática, corresponde aos autovetores da matriz de correlação de entrada \mathbf{R} , e, os correspondentes autovalores determinam a taxa de variação do gradiente ao longo dos eixos principais da superfície de desempenho, afetando, portanto, o tempo de convergência.

3.4 Derivação do algoritmo

No Combinador Linear Adaptativo (CLA), descrito no capítulo 2, a saída, $y_k = \mathbf{W}_k^T \mathbf{X}_k$, é dada como uma combinação linear das amostras da entrada. O sinal desejado d_k é composto de um sinal que desejamos extrair, s_k , adicionado a um ruído que tem distribuição normal $(0,1)$, n_k , na forma $d_k = s_k + n_k$.

Façamos, agora, as seguintes suposições:

- cada vetor de entrada \mathbf{X}_k é estatisticamente independente de todos os outros vetores \mathbf{X}_j , $j < k$;
- \mathbf{X}_k é limitado em um intervalo $[-\delta, \delta]$, onde δ é um número positivo menor que ou igual a 1;
- n_k é estatisticamente independente de \mathbf{X}_k ;
- todas as variáveis têm distribuições de probabilidades não necessariamente gaussianas;
- o vetor peso \mathbf{W}_k é estatisticamente independente de \mathbf{X}_k .

Como em (2.5), temos

$$\varepsilon_k = d_k - \mathbf{X}_k^T \mathbf{W}_k. \quad (3.1)$$

Para desenvolver um algoritmo adaptativo usando o método da descida mais íngreme, nós utilizaríamos $\xi = E[\varepsilon_k^2]$ como função de custo ou critério a ser aplicado sobre o erro. Em vez disso, nós tomaremos $F(\varepsilon_k)$ como função de custo, a qual queremos minimizar. Então, a cada iteração, no processo iterativo, nós teremos uma estimação do

gradiente na forma

$$\begin{aligned}
 \hat{\nabla}_k &= \frac{\partial F(\varepsilon_k)}{\partial \mathbf{W}} \\
 &= F'(\varepsilon_k) \frac{\partial \varepsilon_k}{\partial \mathbf{W}_k} \\
 &= -F'(\varepsilon_k) \mathbf{X}_k,
 \end{aligned} \tag{3.2}$$

onde $F'(\cdot)$ representa a diferenciação de F . Definindo $f(\varepsilon_k) = F'(\varepsilon_k)$ podemos especificar um algoritmo adaptativo do tipo descida mais íngreme. De (2.14), temos:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu f(\varepsilon_k) \mathbf{X}_k. \tag{3.3}$$

3.5 Convergência do vetor peso

A tarefa de filtragem, como sabemos, é realizada através de mudanças nos pesos, os quais são dados por $\mathbf{W}_k = [w_{k1} \ w_{k2} \ \dots \ w_{kM}]$.

Seja $\mathbf{V} = \mathbf{W} - \mathbf{W}_*$ o vetor de desvio do peso, onde \mathbf{W}_* é o vetor peso ótimo, ou seja, $s_k = \mathbf{W}_*^T \mathbf{X}_k$. Assim, teremos

$$\begin{aligned}
 \varepsilon_k &= d_k - y_k \\
 &= s_k + n_k - \mathbf{W}_k^T \mathbf{X}_k \\
 &= \mathbf{W}_*^T \mathbf{X}_k + n_k - \mathbf{W}_k^T \mathbf{X}_k \\
 &= n_k - (\mathbf{W}_k - \mathbf{W}_*)^T \mathbf{X}_k \\
 &= n_k - \mathbf{V}_k^T \mathbf{X}_k.
 \end{aligned} \tag{3.4}$$

Substituindo esta equação em (3.3), teremos

$$\mathbf{V}_{k+1} = \mathbf{V}_k + \mu f(n_k - \mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k. \tag{3.5}$$

Expandindo $f(n_k - \mathbf{V}_k^T \mathbf{X}_k)$ em série de Taylor em torno do valor $-\mathbf{V}_k^T \mathbf{X}_k$, obtemos

$$\begin{aligned} f(n_k - \mathbf{V}_k^T \mathbf{X}_k) &= \sum_{i=0}^{\infty} \frac{f^{(i)}(-\mathbf{V}_k^T \mathbf{X}_k)}{i!} n_k^i \\ &= f(-\mathbf{V}_k^T \mathbf{X}_k) + f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k + \\ &\quad + \frac{1}{2} f''(-\mathbf{V}_k^T \mathbf{X}_k) n_k^2 + \frac{1}{6} f'''(\delta) n_k^3, \end{aligned} \quad (3.6)$$

onde $f^{(i)}$ representa a i -ésima derivada de f , e δ pertence ao intervalo $[0, n_k]$.

Substituindo (3.6) em (3.5), encontramos

$$\begin{aligned} \mathbf{V}_{k+1} &= \mathbf{V}_k + \mu [f(-\mathbf{V}_k^T \mathbf{X}_k) + f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k + \\ &\quad + \frac{1}{2} f''(-\mathbf{V}_k^T \mathbf{X}_k) n_k^2 + \frac{1}{6} f'''(\delta) n_k^3] \mathbf{X}_k. \end{aligned} \quad (3.7)$$

Aplicando o operador expectativa em ambos os lados de (3.7), podemos ver que

$$\begin{aligned} E[\mathbf{V}_{k+1}] &= E[\mathbf{V}_k] + \mu \{ E[f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k] + \\ &\quad + \frac{1}{2} E[f''(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k] \sigma_n^2 \}, \end{aligned} \quad (3.8)$$

onde usamos o fato de que os momentos ímpares do ruído são iguais a zero e $\sigma_n^2 = E[n_k^2]$ é a variância do ruído.

Demonstremos, agora, o seguinte teorema:

Teorema: Seja f uma função não linear, ímpar, definida e contínua em um intervalo $(-\delta, \delta)$, onde δ é um número positivo suficientemente pequeno. Desta forma, $f(\alpha\beta) \approx \alpha f(\beta)$, para todo $\alpha, \beta \in (-\delta, \delta)$.

Demonstração: Observe que $|\alpha| < \delta$, $|\beta| < \delta$ e $f(0) = 0$. De acordo com as hipóteses, existe $\rho > 0$, um número suficientemente pequeno, tal que $|f(\beta)| < \rho$.

Obviamente, também teremos $|f(\alpha\beta)| < \rho$. Multiplicando $|\alpha|$ por $|f(\beta)|$ obtemos a seguinte desigualdade: $|\alpha||f(\beta)| < \delta\rho$ ou $|\alpha f(\beta)| < \delta\rho$. Note que δ e ρ são números pequenos, então o produto então eles é, ainda, pequeno. Desta maneira, podemos fazer $\alpha f(\beta)$ tão próximo de $f(\alpha\beta)$ quanto desejarmos \blacklozenge .

Utilizando o teorema acima, reescrevemos a equação (3.8) como

$$\begin{aligned}
\mathbf{E}[\mathbf{V}_{k+1}] &= \mathbf{E}[\mathbf{V}_k] - \mu \left(\mathbf{E}[f(\mathbf{X}_k)\mathbf{V}_k^T\mathbf{X}_k] + \frac{1}{2}\mathbf{E}[f''(\mathbf{X}_k)\mathbf{V}_k^T\mathbf{X}_k]\sigma_n^2 \right). \\
&= \mathbf{E}[\mathbf{V}_k] - \mu \left(\mathbf{E}[f(\mathbf{X}_k\mathbf{X}_k^T)\mathbf{V}_k] + \frac{1}{2}\mathbf{E}[f''(\mathbf{X}_k\mathbf{X}_k^T)\mathbf{V}_k]\sigma_n^2 \right) \\
&= \mathbf{E}[\mathbf{V}_k] - \mu \left(f(\mathbf{R})\mathbf{E}[\mathbf{V}_k] + \frac{1}{2}f''(\mathbf{R})\mathbf{E}[\mathbf{V}_k]\sigma_n^2 \right) \\
&= \left\{ \mathbf{I} - \mu \left(f(\mathbf{R}) + \frac{1}{2}f''(\mathbf{R})\sigma_n^2 \right) \right\} \mathbf{E}[\mathbf{V}_k]. \tag{3.9}
\end{aligned}$$

Esta equação pode ser utilizada para determinar um limite no parâmetro μ para garantir convergência. Definindo \mathbf{Q} como sendo a matriz dos autovetores de \mathbf{R} , ou seja, as colunas de \mathbf{Q} são formados pelos autovetores de \mathbf{R} , e Λ uma matriz diagonal, cujos elementos da diagonal principal são os autovalores de \mathbf{R} , escrevemos a matriz de correlação de entrada como:

$$\mathbf{R} = \mathbf{Q}\Lambda\mathbf{Q}^{-1}. \tag{3.10}$$

Definindo, também, $\tilde{\mathbf{V}}_k = \mathbf{Q}^{-1}\mathbf{V}_k$ como uma rotação nos vetores pesos, reescrevemos (3.9) da seguinte maneira:

$$\begin{aligned}
\mathbf{Q}\mathbf{E}[\tilde{\mathbf{V}}_{k+1}] &= \left(\mathbf{I} - \mu f(\mathbf{R}) - \frac{\mu}{2}f''(\mathbf{R})\sigma_n^2 \right) \mathbf{Q}\mathbf{E}[\tilde{\mathbf{V}}_k] \implies \\
\mathbf{E}[\tilde{\mathbf{V}}_{k+1}] &= \mathbf{Q}^{-1} \left(\mathbf{I} - \mu f(\mathbf{R}) - \frac{\mu}{2}f''(\mathbf{R})\sigma_n^2 \right) \mathbf{Q}\mathbf{E}[\tilde{\mathbf{V}}_k] \\
&= \left(\mathbf{Q}^{-1} - \mu\mathbf{Q}^{-1}f(\mathbf{R}) - \frac{\mu}{2}\mathbf{Q}^{-1}f''(\mathbf{R})\sigma_n^2 \right) \mathbf{Q}\mathbf{E}[\tilde{\mathbf{V}}_k] \\
&= \left(\mathbf{I} - \mu\mathbf{Q}^{-1}f(\mathbf{R})\mathbf{Q} - \frac{\mu}{2}\mathbf{Q}^{-1}f''(\mathbf{R})\mathbf{Q}\sigma_n^2 \right) \mathbf{E}[\tilde{\mathbf{V}}_k] \\
&= \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2}f''(\Lambda)\sigma_n^2 \right) [\tilde{\mathbf{V}}_k]. \tag{3.11}
\end{aligned}$$

mas, agora, o que temos em (3.11), é justamente o valor esperado de

$$\tilde{\mathbf{V}}_{k+1} = \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right) \tilde{\mathbf{V}}_k, \quad (3.12)$$

a qual pode ser resolvida por indução da seguinte forma: iniciando com um peso inicial arbitrário, $\tilde{\mathbf{V}}_0$, temos, para as primeiras três iterações:

$$\begin{aligned} \tilde{\mathbf{V}}_1 &= \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right) \tilde{\mathbf{V}}_0 \\ \tilde{\mathbf{V}}_2 &= \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right)^2 \tilde{\mathbf{V}}_0 \\ \tilde{\mathbf{V}}_3 &= \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right)^3 \tilde{\mathbf{V}}_0. \end{aligned} \quad (3.13)$$

Generalizando, para a k -ésima iteração, obtemos:

$$\tilde{\mathbf{V}}_k = \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right)^k \tilde{\mathbf{V}}_0. \quad (3.14)$$

De (3.14) vemos que o algoritmo será convergente se

$$\lim_{k \rightarrow \infty} \left(\mathbf{I} - \mu f(\Lambda) - \frac{\mu}{2} f''(\Lambda) \sigma_n^2 \right)^k = \mathbf{0}. \quad (3.15)$$

O termo entre aspas em (3.15) é uma matriz diagonal, cujos elementos da diagonal principal são da forma:

$$\lim_{k \rightarrow \infty} \left(1 - \mu f(\lambda_i) - \frac{\mu}{2} f''(\lambda_i) \sigma_n^2 \right)^k, \quad (3.16)$$

com $i = 0, \dots, L$.

A condição de convergência será satisfeita com

$$\left| 1 - \mu f(\lambda_i) - \frac{\mu}{2} f''(\lambda_i) \sigma_n^2 \right| \leq 1, \quad (3.17)$$

que obtemos se tomarmos

$$0 < \mu < \frac{2}{f(\lambda_{max}) + \frac{1}{2} f''(\lambda_{max}) \sigma_n^2}, \quad (3.18)$$

onde λ_{max} é o maior autovalor da matriz \mathbf{R} .

Se esta condição é satisfeita, segue que

$$\lim_{k \rightarrow \infty} \tilde{\mathbf{V}}_k = \mathbf{0}. \quad (3.19)$$

Se substituirmos $\tilde{\mathbf{V}}_k = \mathbf{Q}^{-1} \mathbf{V} = \mathbf{Q}^{-1} (\mathbf{W} - \mathbf{W}_*)$ em (3.19), encontramos

$$\lim_{k \rightarrow \infty} \mathbf{W}_K = \mathbf{W}_*. \quad (3.20)$$

Desta forma, o algoritmo será convergente se a condição (3.18) for suprida.

Nós podemos determinar a constante de tempo, associada com o i -ésimo autovalor da matriz \mathbf{R} da seguinte maneira: tal como feito na seção (2.3.2), determinemos um envelope exponencial $e^{-t/\tau}$, onde t representa o tempo e τ representa a constante de tempo, que aproxime o comportamento da curva de aprendizagem. De (3.14) temos, para cada iteração, que

$$\tilde{v}_i = \left(1 - \mu f(\lambda_i) - \frac{\mu}{2} f''(\lambda_i) \sigma_n^2\right)^k \tilde{v}_0, \quad (3.21)$$

para $i = 0, \dots, L$.

Definamos

$$r_i \triangleq 1 - \mu f(\lambda_i) - \frac{\mu}{2} f''(\lambda_i) \sigma_n^2 \quad (3.22)$$

como a razão geométrica da seqüência de amostras de \tilde{v}_i . Considerando uma unidade de tempo igual a uma iteração temos que

$$e^{-1/\tau_i} = r_i, \quad (3.23)$$

a qual pode ser expandida como [1]:

$$r_i = e^{-1/\tau_i} = 1 - \frac{1}{\tau_i} + \frac{1}{2! \tau_i^2} - \frac{1}{3! \tau_i^3} + \dots \quad (3.24)$$

Visto que em muitas aplicações τ_i é maior ou igual a 10 e r_i é menor que 1, fazamos a seguinte aproximação:

$$r_i \cong 1 - \frac{1}{\tau_i}. \quad (3.25)$$

Igualando (3.22) e (3.25), obtemos

$$\tau_i = \frac{1}{\mu \left(f(\lambda_i) + \frac{1}{2} f''(\lambda_i) \sigma_n^2 \right)}. \quad (3.26)$$

Quando a relação sinal ruído (SNR) for alta, nós podemos negligenciar o segundo fator no denominador e reescrever (3.26) como [13]

$$\tau_i = \frac{1}{\mu f(\lambda_i)}. \quad (3.27)$$

3.6 Covariância do vetor peso e desajuste

Um sistema adaptativo modifica os seus pesos com o objetivo de encontrar a solução ótima. Como o sistema é ruidoso, no estado estacionário, após a convergência, os pesos assumirão valores em uma “vizinhança” do ótimo, mas não igual a ele. Podemos, assim, determinar a covariância do vetor peso.

Perto do peso ótimo, a superfície gerada por $F(\varepsilon_k)$ se aproxima da superfície quadrática. Desta forma, considerando apenas os dois primeiros termos de (3.6), temos:

$$\mathbf{V}_{k+1} = \mathbf{V}_k + \mu \left[f(-\mathbf{V}_k^T \mathbf{X}_k) + f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \right] \mathbf{X}_k. \quad (3.28)$$

Continuando, multiplicando cada lado de (3.28) pelo seu transposto:

$$\begin{aligned}
\mathbf{V}_{k+1}\mathbf{V}_{k+1}^T &= \left(\mathbf{V}_k + \mu f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k + \mu f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{X}_k \right) \cdot \\
&\quad \left(\mathbf{V}_k^T + \mu f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k^T + \mu f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{X}_k^T \right) \\
&= \mathbf{V}_k \mathbf{V}_k^T + \mu f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{V}_k \mathbf{X}_k^T + \\
&\quad \mu f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{V}_k \mathbf{X}_k^T + f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k \mathbf{V}_k^T + \\
&\quad \mu^2 f^2(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{X}_k \mathbf{X}_k^T + \\
&\quad \mu^2 f(-\mathbf{V}_k^T \mathbf{X}_k) f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{X}_k \mathbf{X}_k^T + \\
&\quad \mu f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{X}_k \mathbf{V}_k^T + \\
&\quad \mu^2 f(-\mathbf{V}_k^T \mathbf{X}_k) f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{X}_k \mathbf{X}_k^T + \\
&\quad \mu^2 f'^2(-\mathbf{V}_k^T \mathbf{X}_k) n_k^2 \mathbf{X}_k \mathbf{X}_k^T. \tag{3.29}
\end{aligned}$$

Como $\mathbf{V}_k \mathbf{X}_k^T = \mathbf{X}_k \mathbf{V}_k^T$, temos:

$$\begin{aligned}
\mathbf{V}_{k+1}\mathbf{V}_{k+1}^T &= \mathbf{V}_k \mathbf{V}_k^T + 2\mu f(-\mathbf{V}_k^T \mathbf{X}_k) \mathbf{V}_k \mathbf{X}_k^T + \\
&\quad 2\mu f'(-\mathbf{V}_k^T \mathbf{X}_k) n_k \mathbf{V}_k \mathbf{X}_k^T + \\
&\quad 2\mu^2 \left(f(-\mathbf{V}_k^T \mathbf{X}_k) f'(-\mathbf{V}_k^T \mathbf{X}_k) \right) n_k \mathbf{X}_k \mathbf{X}_k^T + \\
&\quad \mu^2 \left(f^2(-\mathbf{V}_k^T \mathbf{X}_k) + f'^2(-\mathbf{V}_k^T \mathbf{X}_k) n_k^2 \right) \mathbf{X}_k \mathbf{X}_k^T. \tag{3.30}
\end{aligned}$$

Aplicando o operador expectância em ambos os lados de (3.30) e lembrando que os momentos ímpares do ruído são iguais a zero e que $E[\mathbf{V}_{k+1}\mathbf{V}_{k+1}^T] = E[\mathbf{V}_k\mathbf{V}_k^T]$,

obtemos a expressão

$$\begin{aligned}
\text{Cov}[\mathbf{V}_k] &= \text{Cov}[\mathbf{V}_k] + 2\mu\mathbf{E}[f(-\mathbf{V}_k^T\mathbf{X}_k)\mathbf{V}_k\mathbf{X}_k^T] + \\
&\quad \mu^2\left(\mathbf{E}\left[f^2(-\mathbf{V}_k^T\mathbf{X}_k)\mathbf{X}_k\mathbf{X}_k^T\right] + \mathbf{E}\left[f'^2(-\mathbf{V}_k^T\mathbf{X}_k)n_k^2\mathbf{X}_k\mathbf{X}_k^T\right]\right) \\
&= \text{Cov}[\mathbf{V}_k] + \\
&\quad 2\mu\mathbf{E}\left[f(-\mathbf{V}_k^T\mathbf{X}_k)\mathbf{V}_k\mathbf{X}_k^T\right] + \mu^2\left[f'^2(-\mathbf{V}_k^T\mathbf{X}_k)\right]\mathbf{R}\sigma_n^2, \tag{3.31}
\end{aligned}$$

pois $f^2(-\mathbf{V}_k^T\mathbf{X}_k) \rightarrow 0$ com $\mathbf{V} \rightarrow 0$. Utilizando o Teorema demonstrado na seção (3.5), reescrevemos (3.31) como

$$\begin{aligned}
\text{Cov}[\mathbf{V}_k] &= \text{Cov}[\mathbf{V}_k] - 2\mu f(\mathbf{R})\text{Cov}[\mathbf{V}_k] + \mu^2\mathbf{E}[f'^2(-\mathbf{V}_k^T\mathbf{X}_k)]\mathbf{R}\sigma_n^2 \\
&= \frac{\mu}{2}\mathbf{E}[f'^2(-\mathbf{V}_k^T\mathbf{X}_k)]f^{-1}(\mathbf{R})\mathbf{R}\sigma_n^2, \tag{3.32}
\end{aligned}$$

onde $f^{-1}(\mathbf{R})$ indica a inversa da matriz $f(\mathbf{R})$.

Observemos, agora, que $f'^2(-\mathbf{V}_k^T\mathbf{X}_k)$ é sempre positiva, e que, no estado estacionário, $k \rightarrow \infty$ implica que $n_k \rightarrow \mathbf{V}_k^T\mathbf{X}_k$. Desta forma, simplificamos (3.32):

$$\text{Cov}[\mathbf{V}_k] = \frac{\mu}{2}\mathbf{E}[f'^2(n_k)]f^{-1}(\mathbf{R})\mathbf{R}\sigma_n^2. \tag{3.33}$$

No processo adaptativo, quando os pesos finais estão próximos, mais não iguais a \mathbf{W}_* , temos um excesso no erro final (veja figura 3.2). Na seção 2.3.3, definimos, para a superfície quadrática, o excesso do erro quadrático médio. Douglas e Meng [12] comentando a respeito da dificuldade de se derivarem expressões para medir o desajuste de não linearidades, propõem uma aproximação, a qual utilizaremos aqui, dada por:

$$\text{M} = \frac{\mu\mathbf{E}[f^2(n_k)]\text{tr}[\mathbf{R}]}{2\mathbf{E}[f'(n_k)]\sigma_n^2}. \tag{3.34}$$

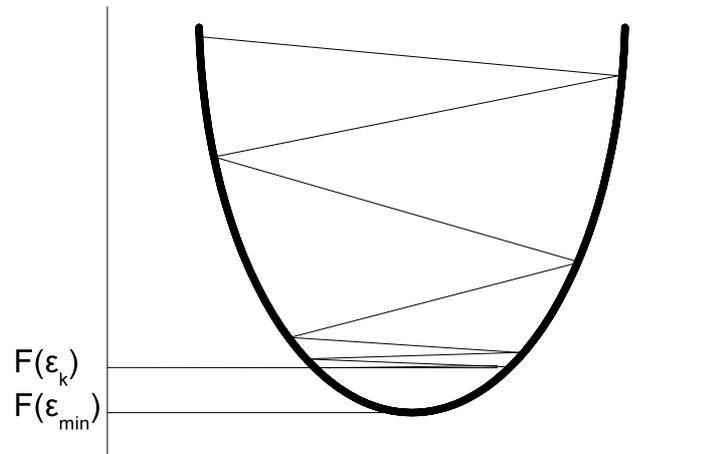


Figura 3.2: Excesso no erro final em relação ao erro mínimo

3.7 Comparação com o LMS

Dado os resultados acima, para o comportamento da adaptação para funções não lineares, surge uma questão: Como escolher uma não linearidade $F(\epsilon_k)$ e consequentemente, seu gradiente $f(\cdot)$ de forma tal que maximize o desempenho do algoritmo adaptativo? Desempenho pode ser mensurado em termos de velocidade de convergência e mínimo excesso. A velocidade de convergência pode ser medida a partir da constante de tempo e o excesso através do desajuste. A implementação de qualquer algoritmo deve levar em conta estas duas características.

Uma boa maneira de mensurar o desempenho do algoritmo é compará-lo com o desempenho do LMS, como Walach e Widrow fizeram [3]. Definimos, assim, um fator de desempenho, χ , como a razão entre a constante de tempo do LMS e a constante de tempo do algoritmo proposto, onde os tamanhos dos passos foram escolhidos de forma tal

que os desajustes sejam os mesmos.

$$\chi = \frac{\tau_{LMS}}{\tau}. \quad (3.35)$$

Observe que é mais vantajoso utilizar o algoritmo proposto em vez do LMS quando $\chi > 1$.

3.8 Conclusão do Capítulo

Neste capítulo, descrevemos a idéia básica da utilização de estatística de alta ordem como uma forma de obtenção de mais informações sobre os sinais envolvidos em um processo adaptativo.

Descrevemos a aplicação de funções não lineares, pares e contínuas, as quais admitem expansão em série de Taylor, como critério aplicado sobre o erro. Realizamos minuciosa análise matemática para descrever as condições de convergência dos algoritmos e deduzimos equações para mensurar a covariância do vetor peso em regime estacionário.

Obtivemos, aqui, um resultado surpreendente, nunca, antes obtido, em outros trabalhos que versavam sobre funções não lineares. De acordo com a equação 3.27 a constante de tempo de um algoritmo adaptativo, em determinadas situações, é influenciada apenas pelas características do sinal de entrada.

4 O Algoritmo Sigmoidal

4.1 Introdução

Neste capítulo aplicaremos a teoria desenvolvida no capítulo anterior, no desenvolvimento de um algoritmo adaptativo denominado Algoritmo Sigmoidal (SA), onde escolhemos a função $\text{Ln}(\cosh \varepsilon)$ como critério a ser aplicado sobre o erro, o qual queremos minimizar. A idéia básica é mostrar que a superfície de desempenho obtida por este critério oferece maior velocidade de convergência, bem como um menor desajuste na busca do peso mínimo.

4.2 A função $\text{Ln}(\cosh \varepsilon)$

A função $\text{Ln}(\cosh \varepsilon)$ é uma não linearidade par, contínua, simétrica, cujo gráfico está representado na figura (4.1). Esta função, como podemos ver, não tem mínimo local, apenas o mínimo global.

A partir desta função, podemos gerar uma família de funções, $\text{Ln}(\cosh \alpha\varepsilon)$, multiplicando o argumento ε por um inteiro positivo α , conforme observamos na figura (4.2).

Uma outra característica dos elementos deste conjunto de funções é que, para um valor fixo de α , podemos determinar intervalos $[-\delta, \delta]$ onde as curvas destas funções tem inclinações maiores do que a curva da função quadrática, neste mesmo intervalo.

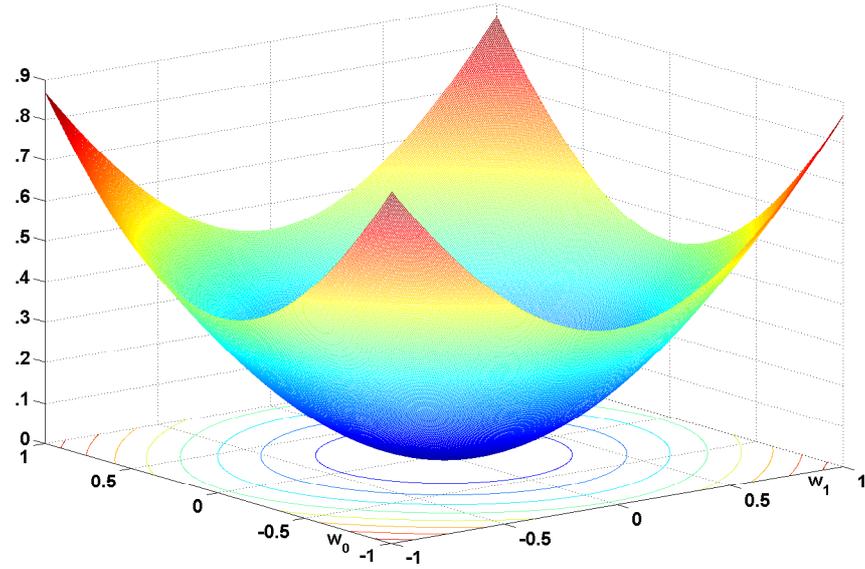


Figura 4.1: Porção da superfície tridimensional gerada pela função $\text{Ln}(\cosh \varepsilon)$, juntamente com alguns contornos.

Podemos observar esta característica na figura (4.3), onde temos plotados os gráficos das funções $\text{Ln}(\cosh 2\varepsilon)$ e ε^2 .

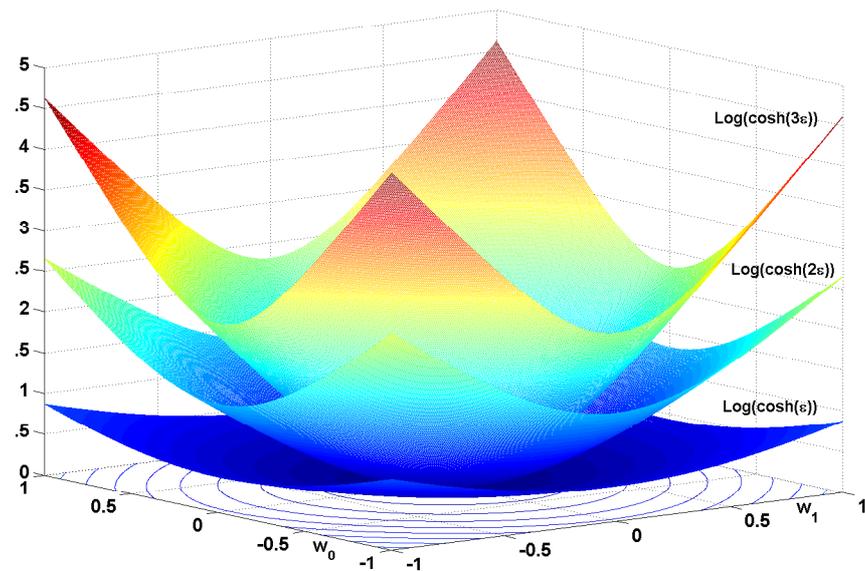


Figura 4.2: Gráficos das funções $\text{Ln}(\cosh \varepsilon)$, $\text{Ln}(\cosh 2\varepsilon)$ e $\text{Ln}(\cosh 3\varepsilon)$.

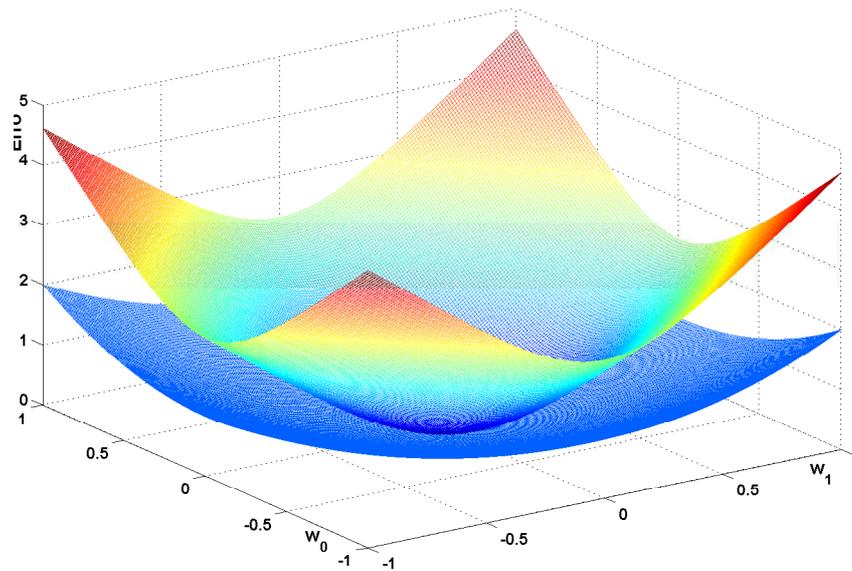


Figura 4.3: Gráficos das funções $\text{Ln}(\cosh 2\varepsilon)$ e ε^2 , onde podemos ver a maior inclinação da primeira, no intervalo $[-1; 1]$

4.3 Derivação do algoritmo Sigmoidal (SA)

Para desenvolver o algoritmo SA, nós tomamos como função de custo a função

$$F_k = \text{Ln}(\cosh \alpha\varepsilon), \quad (4.1)$$

Então, a cada iteração, no processo adaptativo, nós teremos uma estimação do gradiente da forma

$$\hat{\nabla}F_k = -\alpha \tanh(\alpha\varepsilon_k) \mathbf{X}_k. \quad (4.2)$$

Com esta simples estimação do gradiente, nós podemos especificar um algoritmo adaptativo dado por:

$$\begin{aligned} \mathbf{W}_{k+1} &= \mathbf{W}_k - \mu \hat{\nabla}_k \\ &= \mathbf{W}_k + \alpha \mu \tanh(\alpha\varepsilon_k) \mathbf{X}_k. \end{aligned} \quad (4.3)$$

Este é o algoritmo Sigmoidal.

Como antes, μ é uma constante que regula a velocidade e a estabilidade da adaptação.

4.3.1 Convergência do vetor peso, constante de tempo e Desajuste

De acordo com (3.18), o limite de μ para garantir a convergência do algoritmo será dada por:

$$0 < \mu < \frac{2}{\alpha \tanh(\alpha \lambda_{max}) - \alpha^2 \tanh(\alpha \lambda_{max}) \operatorname{sech}(\alpha \lambda_{max}) \sigma_n^2}. \quad (4.4)$$

As constantes de tempo, derivadas a partir de (3.26) e (3.27), serão

$$\tau_i = \frac{1}{\mu \left(\alpha \tanh(\alpha \lambda_{max}) - \alpha^2 \tanh(\alpha \lambda_{max}) \operatorname{sech}(\alpha \lambda_{max}) \sigma_n^2 \right)} \quad (4.5)$$

e

$$\tau_i = \frac{1}{\alpha \mu \left(\tanh(\alpha \lambda_{max}) \right)}. \quad (4.6)$$

E o Desajuste, de acordo com (3.34), será determinado por:

$$\begin{aligned} M &= \frac{\mu E \left[\alpha^2 \tanh^2(\alpha n_k) \right] \operatorname{tr}[\mathbf{R}]}{2E \left[\alpha^2 \operatorname{sech}^2(\alpha n_k) \right] \sigma_n^2} \\ &= \frac{\mu E \left[\tanh^2(\alpha n_k) \right] \operatorname{tr}[\mathbf{R}]}{2E \left[\operatorname{sech}^2(\alpha n_k) \right] \sigma_n^2}. \end{aligned} \quad (4.7)$$

4.3.2 SA versus LMS

Para fazer a comparação, exposta na seção 3.7, com o LMS, inicialmente determinamos a relação entre os parâmetros tamanho do passo dos algoritmos LMS e SA,

considerando iguais desajustes. Igualando (2.21) a (4.7), determinamos:

$$\begin{aligned}\mu_{LMS}\text{tr}[\mathbf{R}] &= \frac{\mu_{SA}\text{E}[\tanh^2(\alpha n_k)]\text{tr}[\mathbf{R}]}{2\text{E}[\text{sech}^2(\alpha n_k)]\sigma_n^2} \Rightarrow \\ \mu_{SA} &= \frac{2\text{E}[\text{sech}^2(\alpha n_k)]\sigma_n^2}{\text{E}[\tanh^2(\alpha n_k)]}\mu_{LMS}\end{aligned}\quad (4.8)$$

Lembrando que a constante de tempo do LMS e do SA foram dadas respectivamente por (2.18) e (4.6), substituímos estas equações em (3.35), utilizando a relação dada por (4.8), obtendo:

$$\begin{aligned}\chi &= \frac{\tau_{LMS}}{\tau_{SA}} \\ &= \frac{1}{2\mu_{LMS}\lambda} \\ &= \frac{1}{\alpha\mu_{SA}\tanh(\alpha\lambda)} \\ &= \frac{\mu_{SA}}{\mu_{LMS}} \frac{\alpha\tanh(\alpha\lambda)}{2\lambda} \\ &= \frac{2\text{E}[\text{sech}^2(\alpha n_k)]\sigma_n^2}{\text{E}[\tanh^2(\alpha n_k)]} \frac{\alpha\tanh(\alpha\lambda)}{2\lambda} \\ &= \frac{\alpha\tanh(\alpha\lambda)\text{E}[\text{sech}^2(\alpha n_k)]\sigma_n^2}{\lambda\text{E}[\tanh^2(\alpha n_k)]}.\end{aligned}\quad (4.9)$$

4.3.3 Simulações com o Algoritmo Sigmoidal

Objetivando verificar a exatidão das equações derivadas na seção anterior, fizemos simulações, onde comparamos os desempenhos dos algoritmos LMS e SA.

Muitos problemas de processamento de sinais, tais como modelagem de planta, cancelamento de ruído, etc., pode ser representado na forma mostrada na figura (4.4) [3], onde temos uma planta representada pela função de transferência polinomial $P(z) = 0.2037z^{-1} + 0.5926z^{-2} + 0.2037z^{-3}$, cuja saída é corrompida por um ruído, n_k . Nosso objetivo é encontrar, de um modo adaptativo, um modelo da planta, $\tilde{P}(z)$. Com este objetivo, utilizamos os algoritmos LMS e SA.

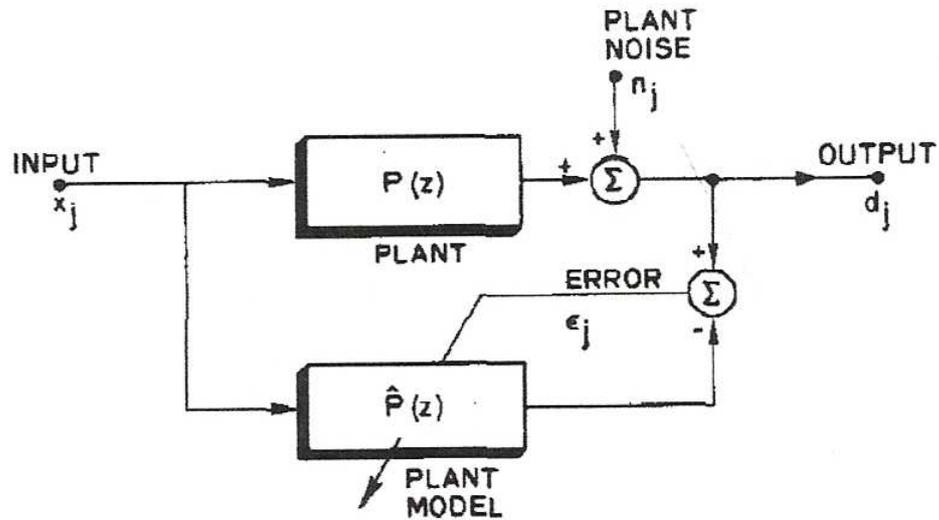


Figura 4.4: Diagrama de blocos da modelagem adaptativa de uma planta

O sinal de entrada foi simulado como um sinal aleatório uniformemente distribuído, limitado no intervalo $[-1, 1]$. Como ruído utilizamos ora um sinal gaussiano de média zero e variância unitária, ora um sinal com distribuição uniforme de probabilidades. O sinal desejado foi posto como a soma do sinal de entrada mais ruído. Os parâmetros tamanho do passo foram $\mu_{LMS} = 0.3e^{-1}$ e μ_{SA} foi posto satisfazendo a equação (4.8) e utilizamos vários valores para o parâmetro α . Observemos que para cada valor de α , modificamos o algoritmo dado por (4.3). Para cada algoritmo, em cada uma das distribuições do ruído, 100 simulações de Monte Carlo foram realizadas. Nas figuras (4.5) e (4.6), abaixo, plotamos as curvas de aprendizagens dos dois algoritmos, onde em uma utilizamos $\alpha = 2$ e na outra foi utilizado $\alpha = 3$, ambas com ruído gaussiano.

4.4 Conclusões do Capítulo

Neste capítulo mostramos o desenvolvimento de uma família de Algoritmos adaptativos, que utilizam como critério aplicado sobre o erro a função $\text{Ln}(\cosh \alpha \epsilon)$, a qual

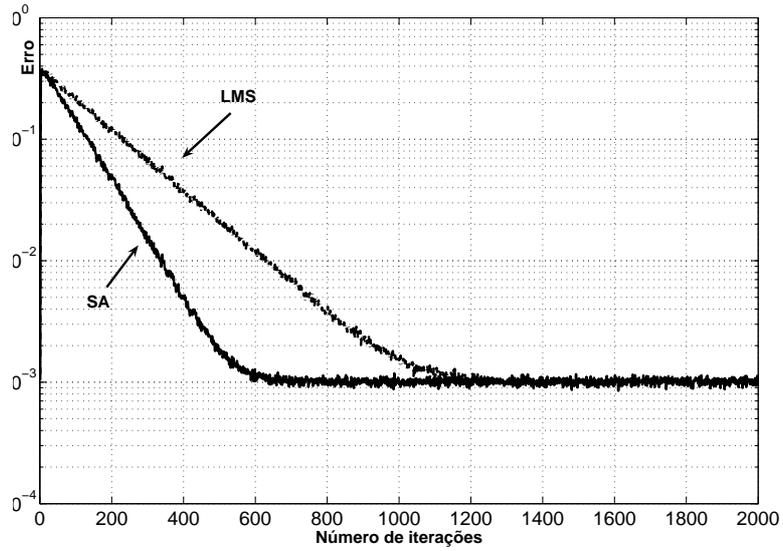


Figura 4.5: curvas de aprendizagem dos algoritmos LMS e SA com $\alpha = 2$

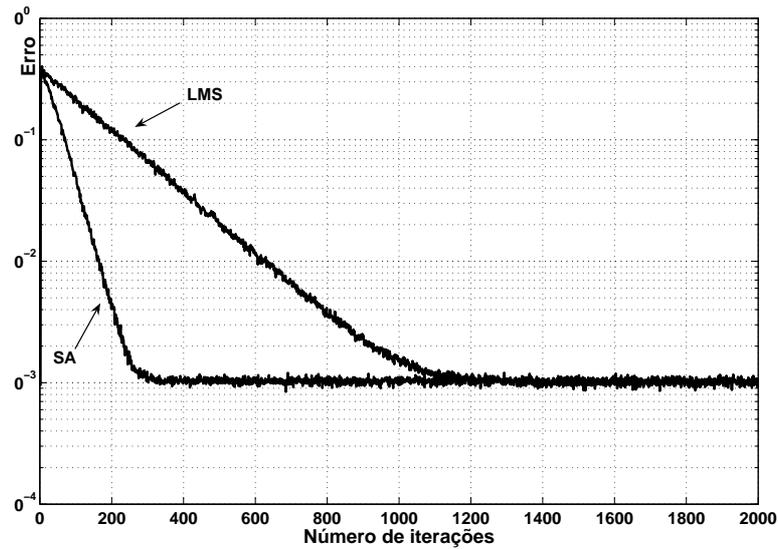


Figura 4.6: curvas de aprendizagem dos algoritmos LMS e SA com $\alpha = 3$

queremos minimizar. isto origina o algoritmo Sigmoidal (SA), dada pela regra (4.3).

Este algoritmo mostrou uma melhora no desempenho, quando comparado com o LMS. Melhora esta que mostrou-se dependente do parâmetro α , ou seja, ao aumentarmos o valor de α , conseqüentemente, aumentando a inclinação da superfície de desempenho, o algoritmo SA aumenta a velocidade de convergência dos pesos, mantendo o mesmo

desajuste. Dando mais ênfase à esta característica do algoritmo SA, temos, na tabela 4.1, os valores do fator de desempenho, χ , obtidos para vários valores de α , em cada uma das duas distribuições do ruído.

Tabela 4.1: Valores de χ para vários α e distribuições de probabilidades gaussiana e uniforme para o ruído.

α	Gaussiano	Uniforme
1	1,53	1,18
2	2,29	1,61
3	3,04	2,12
4	3,79	2,65
5	4,50	3,17
6	5,21	3,66
10	7,64	5,45

5 Estimação Adaptativa Estocástica de Sinal de Impedância Cardiográfica

5.1 Introdução

Um dos grandes desafios da tecnologia diagnóstica em cardiologia é a obtenção de dados não invasivos mais detalhados sobre as funções cardíacas, que representem alternativas simples e baratas aos métodos já existentes. A impedância cardiográfica é uma das opções interessantes neste aspecto, embora seja relativamente pouco utilizada na prática clínica. Ela mede variações cardíacas a cada batida, baseada nas propriedades elétricas da região torácica. Impedância é um valor elétrico alterado pelas variações concomitantes no volume torácico, causadas pelo movimento do sangue em seu interior, durante o ciclo cardíaco [14].

A impedância cardiográfica é medida usando-se quatro eletrodos de superfície, veja figura 5.1, sendo dois colocados em volta do pescoço, e mais dois na região peitoral. Os eletrodos exteriores são usados para injetar um sinal elétrico alternado, de baixa intensidade, no tórax e pescoço, a partir do qual se mede a impedância, através dos eletrodos interiores. Não há risco para o paciente, e o procedimento de obtenção de dados é rápido, simples e não traumático, exatamente como em um ECG. A impedância cardiográfica pode fornecer diversos dados, mas o volume por batida cardíaca é o mais comumente registrado na prática clínica. Este valor, em função do tempo, é calculado pelo aparelho através do que se chama, em termos matemáticos, de derivada do sinal de impedância (em

outras palavras, a velocidade de variação da mesma, em função do tempo). A impedância cardiográfica é o resultado de vários eventos fisiológicos, principalmente mudanças no volume sanguíneo nos tecidos durante o ciclo cardíaco, e mudanças na orientação dos eritrócitos, causada pela variação de velocidade do sangue na aorta.

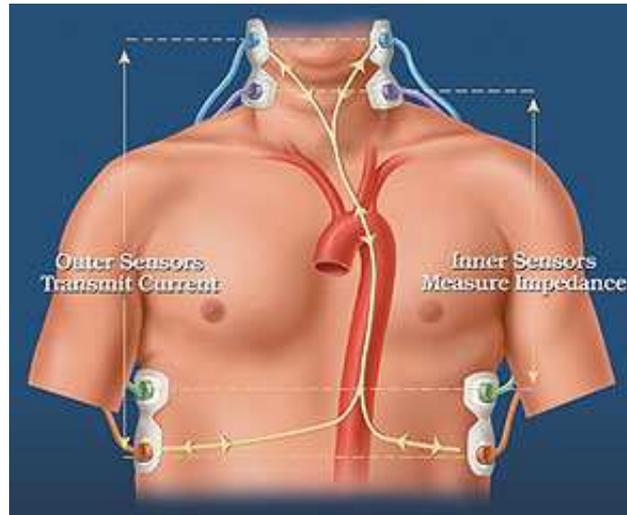


Figura 5.1: Captura de sinais de ICG

O maior problema em Impedância Cardiográfica (ICG) é como eliminar a influência de ruídos, tais como os provenientes da respiração e de movimentos internos do corpo humano, que alteram a forma de onda do ICG. Uma maneira de superar essas limitações é usando filtragem adaptativa, tal como proposto por Barros [14].

Neste tipo de filtro, a componente determinística de um sinal, vinculado a um estímulo, é estimada enquanto o ruído é removido. Esses filtros têm duas entradas: uma primária, que é o sinal a ser filtrado e uma entrada de referência vinculada a um estímulo.

Neste trabalho, o sinal determinístico do ICG é recuperado usando as propriedades de um sinal, em um dado período de tempo, como uma soma de senos e cossenos, ou, como uma Série de Fourier, como proposto por Vaz e Thakor[16]. A estimação dos coeficientes da Série de Fourier é realizada pelo algoritmo SA em cada intervalo RR do

ECG.

5.2 Simulação

Nesta simulação, o filtro adaptativo, na k -ésima iteração, é composto do sinal a ser filtrado d_k , que chamamos de sinal desejado e um vetor sinal de entrada $\mathbf{X}_k = [x_{1,k} \ x_{2,k} \ \dots \ x_{2H,K}]$, conforme figura 5.2, onde H é o número de harmônicos necessários para reconstruir o sinal. O sinal d_k é composto do sinal de interesse, s_k , e um ruído, n_k , descorrelacionado com s_k , ou seja, $d_k = s_k + n_k$. s_k é representado por uma série de Fourier, na forma:

$$s_k = \sum_{i=1}^H w_{*,i} e^{j i \omega_0 k} + \sum_{i=1}^H w_{*,H+i} e^{-j i \omega_0 k}, \quad (5.1)$$

onde j é a unidade complexa e $w_{*,i}$ é o coeficiente de Fourier do i -ésimo harmônico [18].

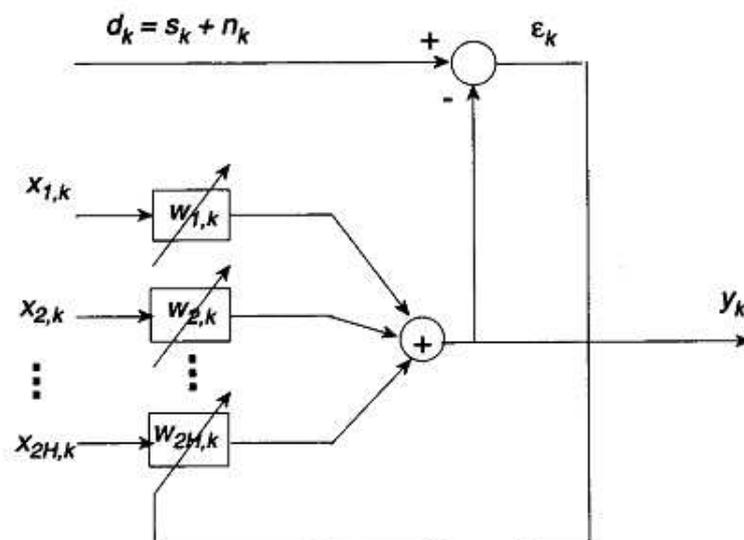


Figura 5.2: Diagrama de bloco do filtro adaptativo. s_k é o sinal determinístico, n_k é o ruído descorrelacionado com s_k . $[x_{1,k} \ x_{2,k} \ \dots \ x_{2H,K}]^T$ é o vetor de entrada.

O sinal de entrada é definido como

$$\mathbf{X}_k = [1 \quad e^{-j\omega_0 t} \quad \dots \quad e^{-j\omega_0 kt} \quad e^{j\omega_0 t} \quad \dots \quad e^{j\omega_0 kt}], \quad (5.2)$$

cuja frequência fundamental ω_0 foi recuperada utilizando-se o algoritmo HIF [17]. Mais uma vez, utilizamos os algoritmos LMS e SA e comparamos os seus desempenhos.

Nosso objetivo é estimar s_k , após calcular os pesos \mathbf{W}_k e o erro atual $\varepsilon_k = d_k - y_k$.

Para testar o desempenho dos filtros, um sinal atual de ICG, sem ruído, seria necessário. Entretanto, até hoje, não vimos nenhum trabalho que utilizasse esse tipo de sinal. Deste modo, simulações computacionais foram realizadas para avaliar o desempenho dos filtros, usando uma onda quadrada, s_k , como sinal de interesse, adicionado a um ruído, n_k , o qual foi simulado como um sinal gaussiano de média zero e variância unitária, como podemos ver na figura 5.3, onde, também, vemos as saídas quando utilizamos o algoritmo LMS e o algoritmo SA com $\alpha = 2$. Neste caso, 100 simulações de Monte Carlo foram realizadas.

Na figura 5.4 podemos ver as curvas de aprendizagens dos dois algoritmos.

Comparamos, também, os desempenhos dos algoritmos SA e LMS com dados reais. Na figura 5.5 podemos ver o sinal de ICG e outros filtrados pelos algoritmos LMS e SA, no domínio do tempo e no espectro de potência, onde os tamanhos do passo foram escolhidos de forma tal a satisfazerem a relação (4.8).

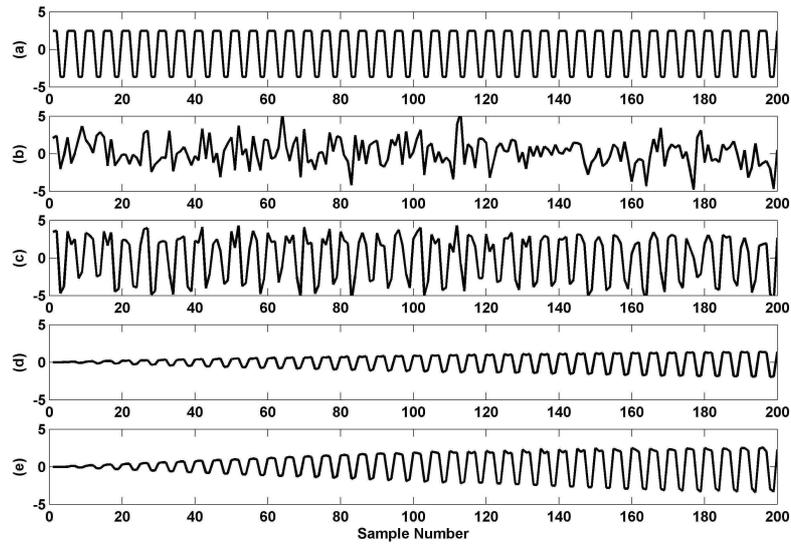


Figura 5.3: a) Sinal de interesse, s_k , uma onda quadrada; b) sinal de ruído, n_k , gaussiano de média zero e variância 1; c) Sinal de interesse mais ruído; d) Saída do filtro com o algoritmo LMS; e) Saída do filtro com o algoritmo SA

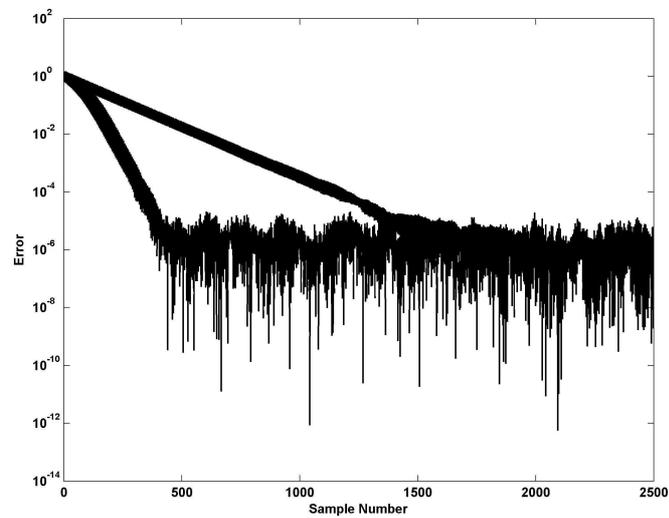


Figura 5.4: curvas de aprendizagens dos algoritmos LMS e SA para o mesmo desajuste e para os tamanhos dos passos dados pela relação (4.8). Vemos que o algoritmo SA converge com mais ou menos 500 iterações enquanto que o LMS converge com 1500 iterações

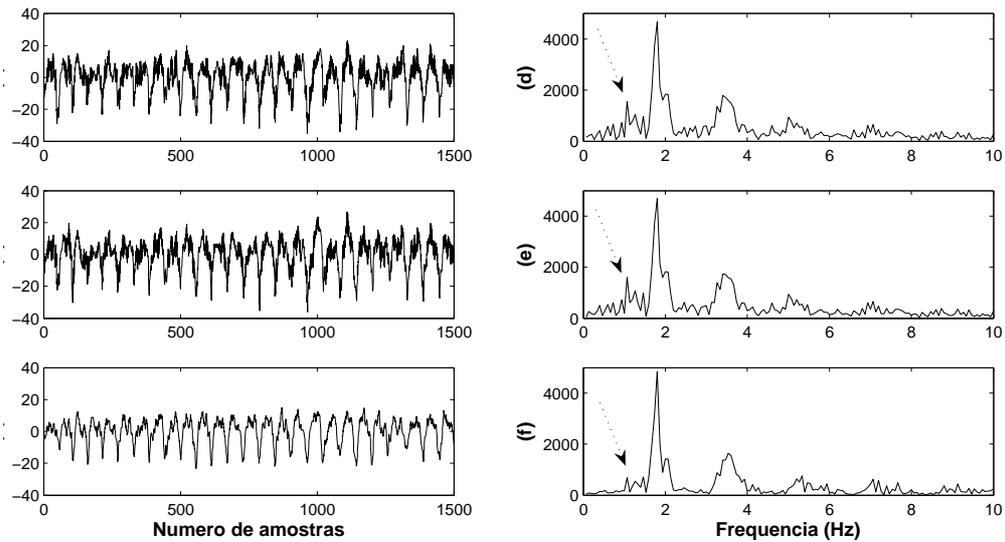


Figura 5.5: Comparação espectro-temporal entre o sinal de ICG real com os sinais de saída dos filtros LMS e SA. (a) Sinal de ICG real; (b) e (c) Sinais de saída dos filtros LMS e SA, respectivamente; (d), (e) e (f) Transformada de Fourier de (a), (b) e (c), respectivamente. As setas indicam a remoção do ruído de 1.8 Hz e seus harmônicos realizada pelo algoritmo SA, quando comparado com o sinal original e a saída do LMS

5.3 Conclusões do capítulo

Neste capítulo, descrevemos, de forma sucinta, a impedância cardiográfica, um novo método não invasivo, não traumático, repetitivo, para obtenção de dados cardíacos. Contudo, as gravações de ICG são alteradas por ruídos, tais como os oriundos da respiração do paciente. Para cancelar estes ruídos é sugerida a utilização de filtros adaptativos, tal como feito por Barros [15], que utilizou o algoritmo LMS. Aqui, nós propomos a utilização do algoritmo SA para recuperar as componentes determinísticas do ICG e como previsto pela nossa teoria e mostrado nas figuras 5.3 e 5.4, usando o algoritmo sigmoidal melhoramos, significativamente, o desempenho na estimação das componentes determinísticas de um sinal.

Em relação às aplicações para dados reais de ICG, podemos ver na figura 5.5 que também, obtivemos uma melhor performance quando comparamos o algoritmo SA com o algoritmo LMS. Isto pode ser visto na análise espectral da referida figura. Comparando as saídas do SA e do LMS, podemos ver que o SA removeu o ruído de 1.8 Hz e seus harmônicos, como indicado pelas setas nas baixas frequências. Além do mais, comparando os sinais temporários em (a), (b) e (c), podemos observar que em (c) temos um sinal menos ruidoso em relação àqueles que temos em (a) e em (b).

6 Conclusões e Proposta de Continuidade

6.1 Conclusões

A utilização de estatística de alta ordem, como uma forma de obtenção de mais informações sobre sinais, tem-se demonstrado de grande valia em sistemas adaptativos. Apesar disso, poucos pesquisadores tem-se utilizado de tais técnicas, provavelmente devido às dificuldades matemáticas advindas das não linearidades.

Neste trabalho, nós desenvolvemos uma substancial ferramenta matemática para analisar a aplicação de funções não lineares, pares e contínuas, as quais admitem expansão em série de Taylor, como critério aplicado sobre o erro. As equações obtidas mostraram-se adequadas e foram comprovadas através das simulações.

Nas simulações, o algoritmo sigmoidal mostrou-se mais eficiente quando comparado com o LMS. Esta eficiência acentua-se ao aumentarmos a inclinação da superfície de desempenho.

Na utilização do algoritmo SA para a determinação das componentes determinísticas de um sinal de impedância cardiográfica, obtivemos uma melhor performance quando comparamos os resultados com os resultados advindos da utilização do algoritmo LMS para realizar a mesma função.

6.2 Proposta de Continuidade

O desenvolvimento matemático aqui apresentado, foi baseado nas características das superfícies de desempenho geradas pelas não linearidades aplicadas sobre o erro. Baseado nesta idéia alguns tópicos de pesquisa podem ser identificados, tais como:

- Utilização de processos geométricos na determinação de funções não lineares a serem aplicadas como critério sobre o erro;
- Desenvolvimento de equações mais adequadas para o desajuste;
- Estudos mais aprofundados sobre a constante de tempo.

Referências

- [1] B. Widrow and Samuel D. Stearns. "Adaptive Signal Processing". Prentice-Hall signal processing series, 1985.
- [2] B. N. Wiener. "Extrapolation, Interpolation and smoothing of Stationary Time Series, with Engineering Applications". New York: Wiley, 1949.
- [3] E. Walach and B. Widrow. "The least mean fourth (LMF) adaptive algorithm and its family". IEEE Trans. Inform. Theory, vol. IT-30, n. 2, p.275-283, Mar, 1984.
- [4] J. A. Chambers, O. Tanrikulu and A. G. Constantinides. "Least mean mixed-norm adaptive filtering". IEEE Electronic Letters, vo. 30, n. 19, sept. 1994.
- [5] Sheldon M. Ross. "Introduction to Probability Models" 6 edition, Academic Press. New york, 1997.
- [6] José C. Principe , Neil R. Euliano, W.Curt Lefebvre,. "Neural and Adaptive Systems: Fundamentals Through Simulations" John Wiley and sons. New York, 2000.
- [7] Athanasios Papoulis, "Probability, random variables, and stochastic processes". 3a. Ed. McGraw-Hill series in electrical engineering, Communications and signal processing, 1991.
- [8] B. Widrow and M. E. Ted Hoff, Jr. "Adaptive switching circuits", IRE WESCON conv. REC. pt. 4, pp. 96-104, 1960.
- [9] B. Widrow et al, "Adaptive noise cancelling: principles and applications", Proceedings IEEE, vol.64, n.8, pp. 1692-1716, Dec.1975.
- [10] Simon Haykin, "Adaptive Filter Theory". 3rd. edition. Prentice Hall, Englewood Cliffs, NJ. 1995.
- [11] Allan K. Barros, José Príncipe, Carlos H. Sales and Noboru Ohnishi, "An algorithm based on the even moments of the error". XIII workshop on neural networks for signal processing. p 879-885, 2003.
- [12] S. C. Douglas and T. Meng, "Stochastic gradient adaptation under general error criteria". IEEE transaction on signal processing, v. 42, p 1335-1351, 1994.
- [13] Ewaldo E. C. Santana, Allan Kardec Barros, Yoshifumi Yasuda e Raimundo C. S. Freire , "Analysis of the Time Constant for the Sigmoidal Algorithm Applied to Biomedical Signals". Aceito para apresentação no **2006 IEEE International Workshop On Medical Measurements and Applications** e a ser publicado nos respectivos Anais. 20 a 21 de abril de 2006. Benevento, Itália.

- [14] Allan K. Barros “Impedância Cardiográfica: Um Novo Método Não Invasivo”. Revista Informédica, 2(7): 19-20, 1994.
- [15] Allan K. Barros, M.Yoshizawa and Y. Yasuda “Filtering Noncorrelated noise in Impedance Cardiography”. IEEE Transactions on Biomedical Engineering, vol. 42, n. 3, march, 1995.
- [16] C. Vaz and N. V. Thakor, “Adaptive Fourier Estimation of time-varying potentials”. IEEE Transactions on Biomedical Engineering, BME-36,p.448-455, 1987.
- [17] Allan K. Barros e Noboru Ohnishi, “Heart Instantaneous Frequency (HIF): an alternative Approach to Extract Heart Rate Variability”. IEEE transaction on Biomedical engineering. v. 48, n.8, august 2001.
- [18] Ewaldo E. C. Santana, Yoshifumi Yasuda, Yoshinori Takeuchi and Allan Kardec Barros, “Adaptive Estimation of Impedance Cardiographic Signal by the Sigmoidal Algorithm”. Proceedings of the fifth International Workshop on Biosignal Interpretation, p.117-120. September 6-8, 2005. Tokio Japan.