

APLICAÇÃO DE WAVELETS PARA REMOÇÃO DE RUÍDOS EM SINAIS UNIDIMENSIONAIS

Aderson Cleber Pifer
UFRN, acpifer@dca.ufrn.br

Carlo Frederico Pereira de Siqueira Campos
PETROBRAS, fredampos@gmail.com

Antonio Luiz Pereira de Siqueira Campos
Professor do CEFET-RN. Doutor em Engenharia Elétrica.
antonioluiz@cefetrn.br

RESUMO

Este trabalho descreve uma investigação da filtragem de diferentes tipos de sinais utilizando *Wavelets*, através do estudo empírico do nível de *Threshold*. Foram utilizados sinais senoidais padrão e sinais de voz. O ruído considerado neste estudo foi o Ruído Gaussiano Branco. As *Wavelets* investigadas foram as de: Daubechies, Haar, Coiflets e Symlets. Resultados numéricos são apresentados para as quatro *Wavelets* investigadas.

PALAVRAS-CHAVE: *Wavelets*, Ruído Gaussiano Branco, Filtragem, *Threshold*.

WAVELET APPLICATION TO REMOVE NOISES ON ONEDIMENSIONAL SIGNALS

ABSTRACT

This paper describes an investigation of the use of wavelets to filtering different types of signals through the empirical study of the threshold level. Voice signal and senoidal signal were used. The considered noise in this study was the white Gaussian noise. The investigated wavelets were: Daubechies, Haar, Coiflets, and Symlets. Numerical results are presented for the four investigated wavelets.

KEY-WORDS: Wavelets, White Gaussian Noise, Filtering, Threshold.

APLICAÇÃO DE WAVELETS PARA REMOÇÃO DE RUÍDOS EM SINAIS UNIDIMENSIONAIS

INTRODUÇÃO

O avanço computacional e o desenvolvimento de novas tecnologias de aquisição de dados fizeram com que surgissem novas aplicações em muitos campos tais como, astronomia, processamento de imagens e visão computacional. Entretanto, dados coletados geralmente possuem ruídos, quer seja devido aos processos de aquisição dos dados ou aos fenômenos naturais ocorridos durante o processo. Para que os dados sejam corretamente analisados faz-se necessário a utilização de técnicas para remoção de ruídos (BURRUS, 1998), (BÖMERS, 2000) e (FODOR, 1998).

Um dos processos utilizados para a remoção de ruídos é realizado através do método numérico computacional conhecido como *Wavelet*; que apresenta como características a concentração de energia localizada em um dado ponto no tempo, permitindo a análise de fenômenos transitórios, não-estacionários e variantes no tempo; e mantendo as características oscilatórias das ondas, o que possibilita a análise no domínio do tempo e da frequência (TASWELL, 1994), (NGUYEN, 1995) e (TASWELL, 1996).

Neste artigo foi realizado um estudo comparativo entre algumas *Wavelets* existentes. No estudo procurou-se encontrar dentre as diversas combinações possíveis de variáveis do sistema, tais como: Tipo e ordem da *wavelet* e escolha do nível de *Threshold*; aquelas que melhor recuperassem os sinais originais.

Este trabalho está subdividido em mais cinco partes: a segunda parte descreve as principais características das *wavelets* consideradas aqui. A terceira parte detalha a técnica de remoção de ruído a partir de uma escolha empírica do nível de *threshold*. A quarta parte mostra os resultados numéricos obtidos. Por fim, a quinta parte apresenta as principais conclusões.

CARACTERÍSTICAS DAS WAVELETS

Nesta seção são descritas algumas características das *Wavelets* utilizadas neste estudo (BÖMERS, 2000). Essas características, bem como as *Wavelets* utilizadas são listadas a seguir:

1. Haar;
2. Daubechies;
3. Coiflets;
4. Symlets.

A *Wavelet* Haar é um caso especial. Ela possui apenas dois coeficientes de filtros e deste modo uma faixa de transição longa é garantida. A função da *Wavelet* Haar é representada por uma onda quadrada, onde sinais “suaves” não são bem reconstruídos. Esta *Wavelet* é a única simétrica e ortogonal. Com relação ao esforço computacional, ela é ideal para processamento em tempo real. Entretanto, a qualidade fica comprometida como será visto nas seções seguintes. Pequenas variações dos coeficientes da *Wavelet* produzem um forte *aliasing*.

As *Wavelets* ortogonais criadas por Ingrid Daubechies⁰ no início dos anos 80, ganharam muita atenção. Estas *Wavelets* foram as primeiras a tornar prática a realização da análise discreta. Ingrid as construiu através do projeto de filtros ortogonais com máximo de relação na resposta em frequência em 0 e metade da taxa de amostragem, impondo como restrição o número de momentos de decaimento (que definem a banda de transição) em uma determinada faixa, obtendo-se assim uma melhor resolução no domínio do tempo. De acordo com a ordem da *Wavelet*, p , são produzidos $2p$ coeficientes dos filtros. As *Wavelets* de Daubechies são assimétricas, contínuas e quanto maior o número de decaimentos mais suave será a banda de transição.

Daubechies construiu uma série de outras *Wavelets* ortogonais: As *Symlets* mantêm algumas características positivas da família das Daubechies. Entretanto, elas foram projetadas para satisfazer as condições de simetria e fase linear, não sendo perfeitamente simétricas, pois a condição de ortogonalidade torna isto impossível.

As *Wavelets* de *Coiflet*, também construídas por Daubechies, limita os momentos de decaimento na função *Wavelet* e também na função escala. Como vantagem, os coeficientes de aproximação podem ser representados pelas amostras do sinal. No entanto a ordem dos filtros torna-se elevada.

TÉCNICA DE REMOÇÃO DO RUÍDO

As três principais etapas para a remoção de ruídos são (FODOR, 1998):

1. Analisar o sinal (decomposição), realizando a transformada *Wavelet* através de bancos de filtros e decimadores.
2. Selecionar o tipo do *Threshold* e o limiar apropriado para cada nível de decomposição.
3. Realizar a síntese do sinal, através de bancos de filtros que possuem a função inversa *Wavelet* e dos interpoladores.

Na primeira e na terceira etapa definem-se o tipo de *Wavelet* e o nível de decomposição e o de reconstrução a serem utilizados, parâmetros esses determinados empiricamente, de acordo com as ondas geradas após a aplicação desses.

Na segunda etapa é preciso ter bastante cuidado, pois a escolha de um tipo ou NÍVEL de *Threshold* inadequado poderá resultar numa significativa perda do sinal original. Dentre alguns tipos de *Threshold*, foram escolhidos dois para serem utilizados:

1. *Soft* – Dado um limiar (nível), os valores do sinal abaixo deste serão zerados e os valores acima deste sofrerão uma redução igual ao valor determinado no limiar (FODOR, 1998):

$$\delta_{\lambda}^S(w) = \text{sgn}(w)(|w| - \lambda) I_{\{|w| > \lambda\}} \quad (1)$$

em que w é a energia do sinal, λ é o nível do *Threshold*, $sgn(w)$ é a função sigmóide e $I_{\{|w|>\lambda\}}$ é a função indicadora.

2. *Hard* – Dado um limiar (nível), os valores do sinal menores que este limiar serão zerados e os valores acima serão preservados (FODOR, 1998).

$$\delta_{\lambda}^h(w) = wI_{\{|w|>\lambda\}} \quad (2)$$

Há na literatura algumas maneiras de se estimar o ruído fazendo uso de fórmulas e com isso, determinando assim, melhor o nível do *Threshold*. Nesse trabalho, porém, optou-se por determinar este limiar empiricamente.

Pôde-se perceber no decorrer deste estudo que uma escolha mal feita do nível do *Threshold* poderá prejudicar a reconstrução do sinal após a filtragem do mesmo. O que se deseja eliminar é a energia do ruído e não a do sinal.

Quando se escolhe um nível de *Threshold* superior ao do ruído, juntamente com o mesmo eliminamos parte da energia do sinal, prejudicando assim a reconstrução deste último conforme pode ser visto na Figura 1.

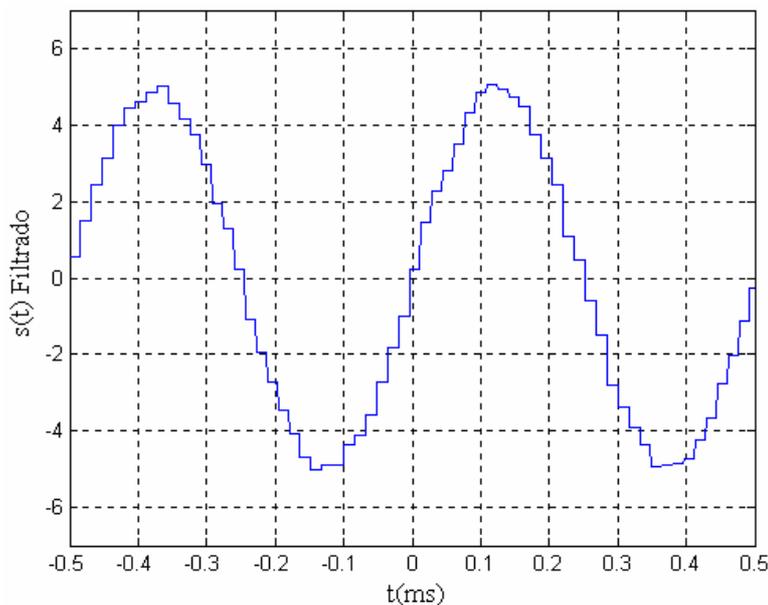


Figura 01 – Eliminação do ruído e de parte do sinal original.

Quando a escolha deste limiar é inferior à energia do ruído, parte deste não será eliminada, sendo reconstruído juntamente com o sinal original, conforme pode ser visto na Figura 02.

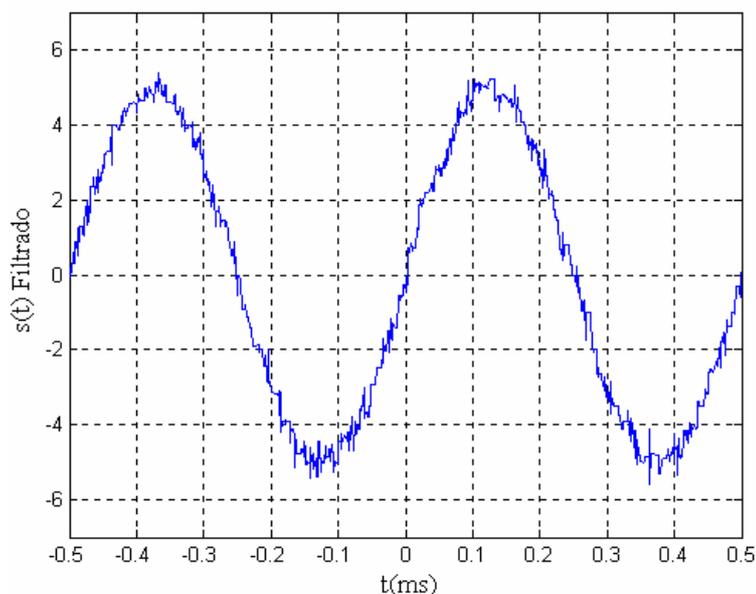


Figura 02 – Eliminação apenas de parte do ruído.

RESULTADOS NUMÉRICOS

Os resultados numéricos apresentados nesta seção foram obtidos utilizando-se o programa computacional Matlab®. Durante a elaboração da transformada *Wavelet* inversa (*idwt*), verificou-se que o sinal reconstruído possuía um número maior de amostras que o sinal ruidoso original. Para solucionar este problema, gerado pela convolução, o sinal resultante foi truncado. Os sinais utilizados no estudo foram:

1. Um sinal senoidal discreto com frequência de 2kHz, amplitude de 5 volts e representado com 1000 amostras ilustrado na Figura 03;
2. Um sinal de voz PCM de 8 bits gravado no ambiente Windows, obtido a uma taxa de 11025 amostras/segundo, conforme pode ser visto na Figura 04;
3. Um ruído Gaussiano branco, que foi adicionado aos sinais anteriormente citados. Esse tipo de ruído foi escolhido por apresentar uma distribuição uniforme por todo o espectro de frequência. Este ruído foi gerado através da função *awgn* do Matlab® e pode ser visto na Figura 05.

Para se comparar os resultados obtidos para as diferentes *wavelets* foram utilizados parâmetros estatísticos de qualidade de filtragem como: o erro médio quadrático (MSE), o erro médio quadrático normalizado (MSE_n) e a relação sinal-ruído (SNR), listados nas equações (3) – (5).

$$MSE = \frac{1}{N} \sum_{i=1}^N (X_i - \hat{X}_i)^2 \quad (3)$$

$$MSE_n = \frac{\sum_{i=1}^N (X_i - \hat{X}_i)^2}{\sum_{i=1}^N (X_i)^2} \quad (4)$$

$$SNR = 10 \log_{10} \left(\frac{1}{MSE_n} \right) \quad (5)$$

em que X_i é o sinal original e \hat{X}_i é o sinal filtrado.

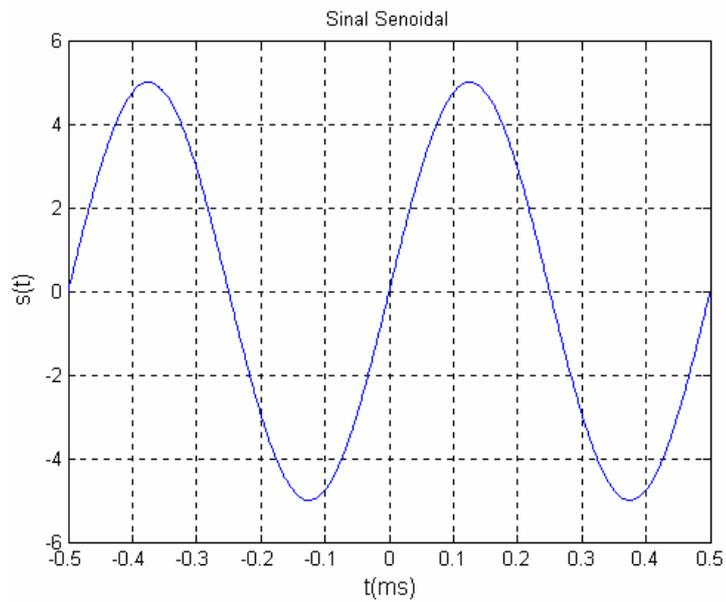


Figura 03 – Sinal senoidal de 2kHz.

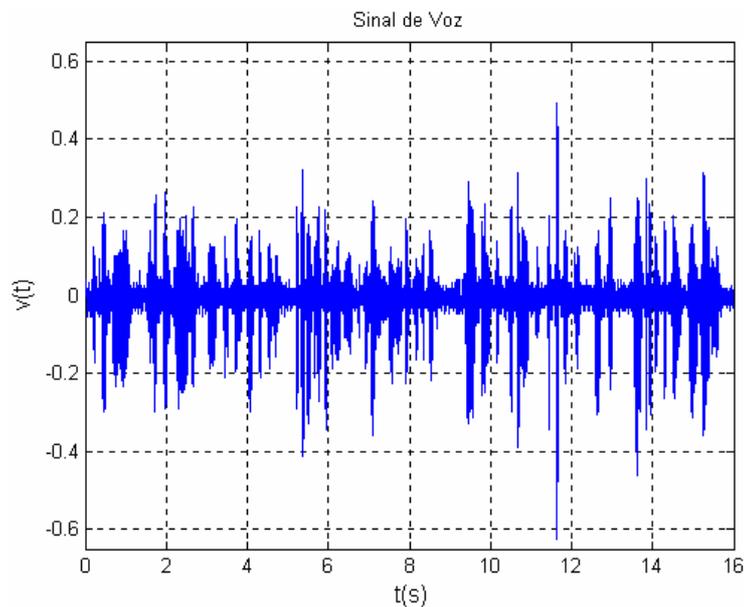


Figura 04 – Sinal de voz gravado.

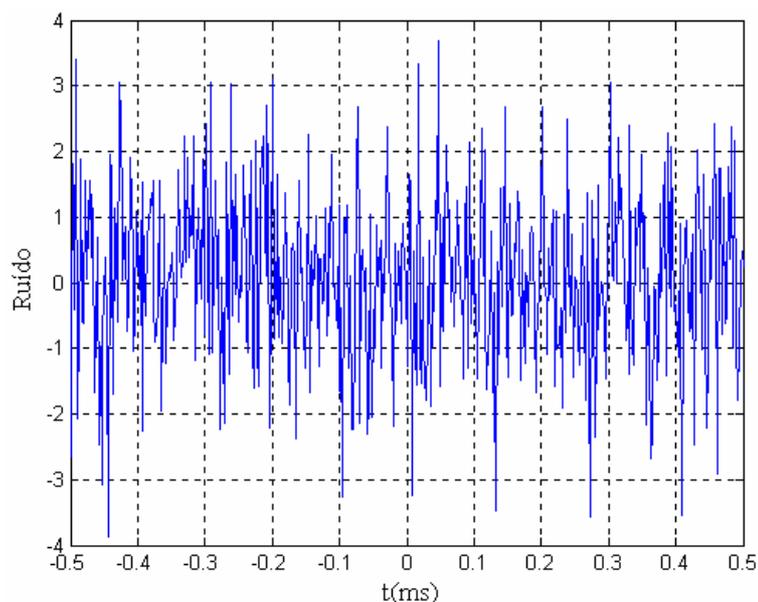


Figura 05 – Ruído Gaussiano branco.

Na Tabela 1 são apresentados os resultados obtidos com a remoção do ruído no sinal senoidal para as *Wavelets* consideradas nesse trabalho, mostrando os melhores e piores resultados obtidos a partir da escolha empírica do nível de *threshold*.

Tabela 01: Resultados obtidos para o sinal senoidal.

| Wavelet Haar | | | | | |
|---|-------------|--------|------------------|---------|---------------------------|
| Tipo de <i>threshold</i> | Decimadores | MSE | MSE _n | SNR | Nível do <i>threshold</i> |
| (*) | 3 | 0,0217 | 0,0017 | 27,5937 | 1 |
| (*) | 5 | 0,1685 | 0,0135 | 18,6975 | - |
| Wavelet Daubechies de Terceira Ordem – db3 | | | | | |
| (*) | 5 | 0,0036 | 0,0003 | 35,4455 | 1 |
| (*) | 7 | 0,3358 | 0,0269 | 15,7037 | - |
| Wavelet Coiflets de Primeira Ordem – Coif1 | | | | | |
| <i>Soft</i> | 5 | 0,0068 | 0,0005 | 32,6088 | 0,70 |
| (*) | 7 | 0,7284 | 0,0583 | 12,3409 | - |
| Wavelet Daubechies de Décima Quinta Ordem – db15 | | | | | |
| <i>Soft</i> | 5 | 0,0053 | 0,0004 | 33,7674 | 0,75 |
| <i>Hard</i> | 1 | 0,0451 | 0,0036 | 24,4201 | 0,90 |
| Coiflets de Quinta Ordem – Coif1 | | | | | |
| <i>Soft</i> | 5 | 0,0050 | 0,0004 | 34,0061 | 1 |
| <i>Hard</i> | 1 | 0,0443 | 0,0035 | 24,4986 | - |

Pode-se observar na Tabela 01 que o melhor resultado obtido para a filtragem do sinal senoidal foi através da *Wavelet* Daubechies de 3^a Ordem, com 5 níveis de resolução e sem a necessidade do uso de *Threshold*. A não necessidade da utilização de um determinado nível de *Threshold* foi devido ao sinal senoidal possuir apenas uma frequência, facilitando a separação entre o ruído e o sinal original, ao passar nos bancos de filtros. É importante ressaltar também, que apesar de algumas *Wavelets* suportarem um nível maior de resolução nem sempre é necessário ou desejável utilizá-los, pois a partir de um nível “ótimo”

verificou-se a perda de sinal original. Conclui-se ainda que, para este estudo, o número de coeficientes não teve uma importância direta na qualidade do sinal reconstruído. A Figura 06 ilustra os gráficos do sinal ruidoso (Figura 06) e do sinal filtrado com o melhor resultado obtido (Figura 07).

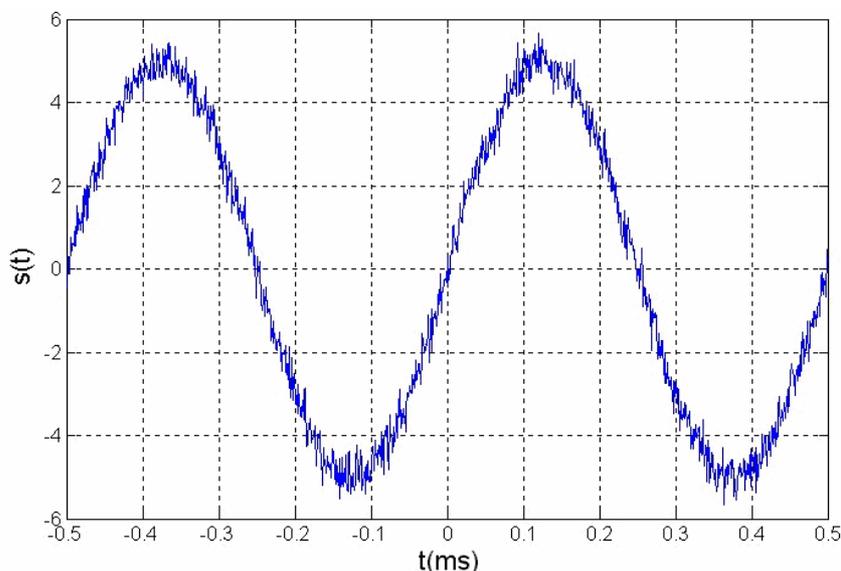


Figura 06 – Sinal senoidal ruidoso.

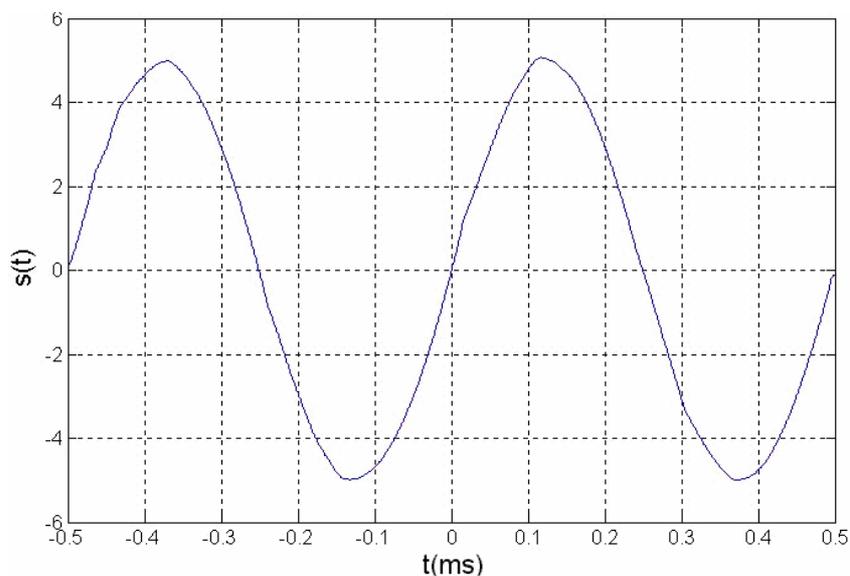


Figura 07 – Sinal senoidal reconstruído.

No segundo caso, foi utilizado um sinal de voz e verificou-se que, devido ao espalhamento espectral do sinal de voz, torna-se mais complicada a separação das altas e baixas frequências do sinal ruidoso, dificultando assim, a remoção do ruído. Com isso aumenta-se o número de tentativas de se aperfeiçoar os parâmetros. Pequenas variações no tipo e nível do *Threshold* podem causar distorções e/ou perdas no sinal a ser filtrado. Ao término de alguns processos o sinal de voz estava metalizado ou extremamente ruidoso, não sendo perceptível ao ouvido humano nenhuma melhora. Como o sinal não foi gravado em ambiente apropriado, o mesmo já possuía um determinado nível de ruído, neste caso o *Threshold* foi de fundamental importância para a redução do ruído e, principalmente, para

a reconstrução do sinal. A Tabela 02 ilustra os principais resultados obtidos. Para esse tipo de sinal o melhor resultado obtido foi para a *wavelet* Symlet de quinta ordem.

Tabela 02: Resultados obtidos para o sinal de voz.

| Wavelet Haar | | | | | |
|---|-------------|---------|------------------|---------|---------------------------|
| Tipo de <i>threshold</i> | Decimadores | MSE | MSE _n | SNR | Nível do <i>threshold</i> |
| <i>Soft</i> | 3 | 11,5220 | 0,0325 | 14,8798 | 0,010 |
| <i>Hard</i> | 5 | 15,0182 | 0,0424 | 13,7289 | 0,020 |
| Wavelet Daubechies de Terceira Ordem – db3 | | | | | |
| <i>Soft</i> | 3 | 9,9861 | 0,0282 | 15,5011 | 0,010 |
| <i>Hard</i> | 5 | 13,0492 | 0,0368 | 14,3392 | 0,025 |
| Wavelet Coiflets de Primeira Ordem – Coif1 | | | | | |
| <i>Soft</i> | 3 | 10,2580 | 0,0289 | 15,3844 | 0,010 |
| <i>Hard</i> | 5 | 13,3948 | 0,0378 | 14,2257 | 0,025 |
| Symlets Quinta Ordem – Sym5 | | | | | |
| <i>Soft</i> | 3 | 9,8014 | 0,0277 | 15,5822 | 0,010 |
| <i>Hard</i> | 5 | 12,7359 | 0,0359 | 14,4447 | 0,025 |

CONCLUSÕES

Há ainda um enorme campo a ser explorado e aperfeiçoado dentro das *Wavelets*. Este estudo se restringiu a remoção de ruídos de sinais senoidais padrão e sinais de voz. Pôde-se verificar que esta ferramenta possui particularidades para cada situação, pois a variação de parâmetros, tais como: Tipo da *Wavelet*, nível de decomposição, características do sinal e/ou ruído que se deseja analisar, nível e tipo de *Threshold*, etc. é muito grande, permitindo obter inúmeras soluções para determinados tipos de problemas. Sugere-se que, em trabalhos futuros, sejam realizados estudos mais aprofundados neste campo, dentre os quais podemos citar uma investigação do processo *Lifting*, que altera a estrutura de cálculo das *Wavelets*.

REFERÊNCIAS

1. BÖMERS, F., Wavelets in real time digital audio processing: Analysis and sample implementations, Maio – 2000, 119 folhas, Departamento de Ciências da Computação, Universidade de Mannheim, Mannheim, Alemanha.
2. BURRUS, C.S., GOPINATH, R.A. e GUO, H., **Introduction to Wavelets and Wavelet transforms**, Prentice Hall, Texas, Estados Unidos, 1998.
3. FODOR, I.K. e KAMATH, C., Denoising Through Wavelet Shrinkage: An Empirical Study, *Journal of Electronic Imaging*, pp. 1 – 16, 2001.
4. NGUYEN, T.Q., A Tutorial on Filter Banks and Wavelets, International Conference on Digital Signal Processing, Cypress, 1995.
5. TASWELL, C. e MCGILL, K.C., Algorithm 735: Wavelet Transform Algorithms for Finite-Duration Discrete-Time Signals, *Transactions on Mathematical Software*, Vol. 20, No. 3, pp. 398 – 412, 1994.

6. TASWELL, C., Specifications and Standards for Reproducibility of Wavelet Transforms, Proceedings of 7th ICSPAT, 1996.