

Redução de Ruído em Áudio Digital

Aluno: Fernando Raganhan Barbosa
Supervisor: Marcelo Gomes de Queiroz

7 de fevereiro de 2010

Sumário

I	Parte Objetiva	1
1	Introdução	1
2	Representação de Áudio Digital	2
2.1	PCM	2
2.2	Teorema da Amostragem	2
3	Ruído	4
3.1	Ruído Branco	4
3.2	SNR - Signal to Noise Ratio	5
4	Filtros	5
4.1	Transformada z	5
4.2	Filtros Lineares	6
4.3	Filtros FIR e IIR	6
5	Processos Estacionários e Não Estacionários	7
6	Correlação de processos aleatórios	8
6.1	Processo Ergódicos	8
7	Notação	9
8	Filtro de Wiener	9
8.1	Calculando as correlações	11
9	Filtros Adaptativos	12
9.1	Adaptação por Bloco (BLS)	12
9.2	RLS - Recursive Least Square	13
9.3	LMS - Least Mean Squared Error	15
9.4	Comparação	15
10	Alguns Parâmetros utilizados	16
10.1	Índice de Distorção de Voz	16
10.2	Fator de Redução de Ruído	17

11 Filtro de Wiener com Trade-off	17
12 Implementação	18
13 Experimentos	19
II Parte Subjetiva	22
14 Dificuldades e Frustrações	22
15 Matérias Relacionadas	22
15.1 MAC0122 - Princípios de Desenvolvimento de Algoritmos . . .	22
15.2 MAE0121 - Introdução a Probabilidade e a Estatística I . . .	22
15.3 MAE0228 - Noções de Probabilidade e Processos Estocásticos	23
15.4 MAT0221 - Cálculo Diferencial e Integral IV	23
15.5 MAT0139 - Álgebra Linear para Computação	23
15.6 MAC0300 - Métodos Numéricos da Álgebra Linear	23
15.7 MAC0337 - Computação Musical	23
16 Aplicação de Conceitos	23

Resumo

Ruído é uma característica ubíqua dentre os sinais digitais, e daí a importância de poder analisar e reduzir esse sinal indesejado para poder preservar a informação sendo transmitida (em uma conversa, por exemplo). O Filtro de Wiener é uma das técnicas fundamentais em processamento de sinais, inclusive em redução de ruídos. Neste trabalho vamos analisar essa técnica e algumas variações para reduzir ruído em sinais de voz.

Parte I

Parte Objetiva

1 Introdução

Todo sinal digital de áudio está, de uma forma ou de outra, contaminado por ruído. Seja devido a interferência eletromagnética existente na transmissão de um ligação pelas linhas telefônicas, ou pelo ruído ambiente como o barulho de carros ou outras pessoas conversando durante a gravação do áudio em questão.

Amenizar esse sinal indesejado é extremamente importante, principalmente em áreas como telefonia e VoIP, já que a qualidade do equipamento de gravação é baixa, a transmissão ocorre por longas distâncias, e em muitos casos o meio não é muito resistente à interferência de outros sinais. Além disso existem limitações técnicas devido a qualidade da representação digital do sinal analógico para possibilitar a transmissão em tempo real.

Redução de ruídos é uma tarefa complexa devido ao número de fatores que influenciam esse processo. As características do que é considerado ruído muda muito dependendo do contexto além de poderem variar ao longo do tempo. Outro problema é definir que aspecto deseja-se melhorar no sinal, como diminuir a fadiga auditiva ou melhorar aspectos gerais da qualidade da fala. Por isso neste trabalho vou supor algumas coisas sobre o sinal ruidoso: que ele é aditivo e varia lentamente ao longo do tempo. Além disso vou focar o caso da telefonia onde somente um microfone é utilizado para captar o sinal e o sinal limpo é composto por voz de uma ou mais pessoas.

Dentre as diversas técnicas utilizadas para resolver este problema vou analisar o filtro de Wiener. Filtros de Wiener e outras técnicas de redução de ruído como *signal subspace* [7] conseguem atingir redução de ruído, mas como subproduto introduzem distorções no sinal de fala. O método de *signal subspace* até define explicitamente sua análise neste *trade-off* entre ruído e clareza. Escolhi os filtros de Wiener em uma das abordagens mais clássicas, além de que a maioria desses métodos se relaciona com os filtros de Wiener de uma forma ou de outra.

Primeiramente vou passar pelos conceitos necessários para entender o funcionamento dos filtros de Wiener, depois vem a apresentação do filtro de Wiener ótimo e suas versões adaptativas. Por fim vou analisar as mudanças

propostas no artigo *New Insights Into the Noise Reduction Wiener Filter* [8] e qual a vantagem de usar ela com as versões adaptativas, seguido de alguns experimentos.

2 Representação de Áudio Digital

O som é formado ondas e, conseqüentemente, é um processo analógico não podem ser representados fielmente por um computador. Por isso diversos métodos de digitalização foram criados com este problema em mente. Esses métodos, em geral, precisam realizar duas tarefas: a amostragem ou processo de discretizar o tempo, e a quantização ou processo de discretização dos valores do sinal de cada amostra.

Apesar de existirem diversas formas de realizar a digitalização do sinal analógico, o PCM(*Pulse Code Modulation*) é amplamente usado e considerado como um padrão. É o método utilizado em CDs de áudio, telefones digitais e em diversos outros formatos de áudio digital.

2.1 PCM

O PCM funciona em duas etapas:

- Amostragem: A amplitude é amostrada em intervalos regulares. O número de intervalos por segundo é chamado de taxa de amostragem, *Sample Rate* em inglês, e é medida em Hertz.
- Quantização: Para cada uma dessas amostras a amplitude do sinal é então arredondada para o número mais próximo dentro de uma escala discreta.

2.2 Teorema da Amostragem

A escolha da taxa de amostragem é de extrema importância já que afeta duas características fundamentais do sinal digital: o tamanho do arquivo em bits e a faixa de frequência que pode ser representada. O primeiro é bem natural, quando mais amostras por segundo maior é o espaço necessário para armazenar esse sinal. O segundo se deve ao problema de aliasing.

Aliasing é o nome dado a distorção que frequências muito altas sofrem ao serem representadas em uma taxa de amostragem muito baixa que fazem

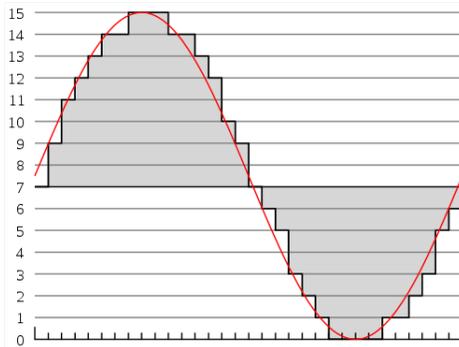


Figura 1: Sinal amostrado com 4 bits por amostra, [1]

com que elas sejam reconhecidas como frequências mais baixas. Isso acontece porque essas duas frequências seriam amostradas da mesma forma, como mostra a figura 2.

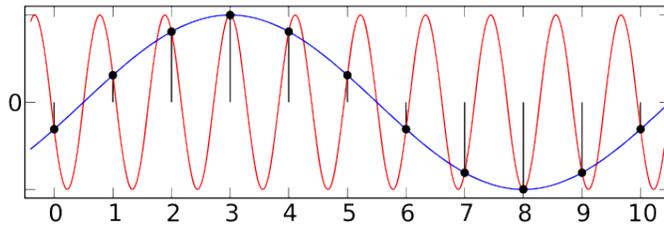


Figura 2: Os pontos marcam onde a amostragem é feita, [2]

O teorema de de Shannon-Nyquist mostra que a maior frequência que um arquivo com taxa de amostragem SR consegue representar é $SR/2$. É por esse motivo que a taxa de amostragem em CDs é de 44100Hz ou frequência limite 22050Hz , o que passa do limite de audição humana. Em telefonia a taxa escolhida é bem menor, em torno de 8000Hz , devido à maior importância em diminuir o tamanho do sinal para poder transmiti-lo em tempo real. Mas essa taxa é satisfatória para representar voz já que a fala humana normalmente não ultrapassa os 4kHz [3].

3 Ruído

Antes de começar a falar sobre como reduzir ruído é interessante definir o que vamos considerar como ruído. Ruído é qualquer sinal que interfira com outro sinal portador de informação, nesse caso a voz. Pode ser classificado de diferentes formas:

- Forma de produção: acústico, eletromagnético, eletrostático, eco, etc.
- Relação com o sinal desejado: aditivo, multiplicativo, etc.
- Forma espectral: branco, colorido, impulsivo, entre outros.

Todo ruído neste trabalho será considerado aditivo, ou seja, o sinal contaminado $y(m)$ é definido por:

$$y(m) = x(m) + r(m) \tag{1}$$

onde $x(m)$ é o sinal portador de informação ou sinal original e $r(m)$ é o ruído.

3.1 Ruído Branco

Nos experimentos foi utilizado ruído branco nos testes então é importante defini-lo. Ele foi escolhido por ser simples de gerar, por ser não correlacionado com o sinal original e por ser considerado como um ruído canônico. Além disso ele é usado para modelar tanto ruídos elétricos, já que são praticamente aleatórios, quanto ruídos de fundo. É usado para ruído de fundo, porque numa situação crítica a mistura dos diversos ruídos presentes no ambiente o torna praticamente aleatório. Esses dois ruídos são os contaminantes mais comuns de sinais telefônicos tornando a escolha do ruído branco como modelo adequada.

Ruído branco é definido como um processo aleatório não correlacionado formado pela combinação simultânea de sinais em todas as frequências. O adjetivo branco vem de uma analogia com a luz branca que é formada pela combinação de todas as as frequências cromáticas. Ruído verdadeiramente branco é impossível de conseguir porque ele é formado da combinação de infinitas frequências, mas é possível construir um ruído limitado a uma certa banda de frequências com as mesmas características.

3.2 SNR - Signal to Noise Ratio

SNR é a principal medida de ruído e é definida da seguinte forma:

$$SNR = \frac{P_{sinal}}{P_{ruído}} \quad (2)$$

ou seja a relação entre a potência do sinal original e a potência do ruído. Quando temos um sinal livre de ruídos o SNR tende ao infinito e quando temos somente ruído o SNR é igual a zero. Também pode ser definido em termos de dB da seguinte forma:

$$SNR(dB) = 10 \log_{10} \left(\frac{P_{sinal}}{P_{ruído}} \right) = P_{sinal,dB} - P_{ruído,dB} \quad (3)$$

4 Filtros

Filtros são os componentes básicos de qualquer sistema de processamento de sinal digital. Suas aplicações são as mais variadas e vão de filtros passa banda, redução de ruídos, radares e simulação de instrumentos musicais entre outros. Nesta seção vamos apresentar alguns conceitos importantes sobre filtros além da formulação geral de filtros lineares, categoria que inclui o filtro de Wiener.

4.1 Transformada z

A transformada z é essencial para entender o comportamento de filtros temporais discretos, como o filtro de Wiener por exemplo. Ela é uma generalização da transformada discreta de Fourier. Ela é definida da seguinte forma:

$$X(z) = \sum_{m=-\infty}^{\infty} x(m)z^{-m} \quad (4)$$

Se escolhermos z tal que $z = re^{j2\pi f}$ temos que

$$X(z = re^{j2\pi f}) = \sum_{m=-\infty}^{\infty} x(m)e^{-j2\pi fm} \quad (5)$$

que é a transformada discreta de Fourier, confirmando que a transformada z é uma generalização da transformada discreta de Fourier. Ela é uma transformada linear e possui a seguinte propriedade importante:

- *Time-Shifting*: a variável z pode ser encarada como um fator de atraso ou mudança de fase. z^{-k} corresponde a um atraso de k amostras.

$$\begin{aligned}
\hat{X}(z) &= \sum_{m=-\infty}^{\infty} x(m-k)z^{-m} \\
&= \sum_{n=-\infty}^{\infty} x(n)z^{-(n+k)} \\
&= z^{-k} \sum_{n=-\infty}^{\infty} x(n)z^{-n} \\
&= z^{-k} X(z)
\end{aligned} \tag{6}$$

A resposta de qualquer filtro digital pode ser expressa pela seguinte relação:

$$H(z) = \frac{Y(z)}{X(z)} \tag{7}$$

onde $X(z)$ é a transformada z do sinal de entrada e $Y(z)$ é a transformada z do sinal de saída. A partir desta relação é possível recuperar os polos e os zeros do filtro. Os pólos correspondem às frequências que serão amplificadas e os zeros às que serão atenuadas no processo de filtragem.

4.2 Filtros Lineares

Filtros lineares são uma subclasse importante de filtros digitais que podem ser representados pela seguinte equação de filtro:

$$y(n) = \sum_{i=0}^M a_i x(n-i) - \sum_{i=1}^N b_i y(n-i) \tag{8}$$

Onde $x(n)$ é o sinal de entrada, $y(n)$ a saída e a_i e b_i os coeficientes reais que definem as características do filtro.

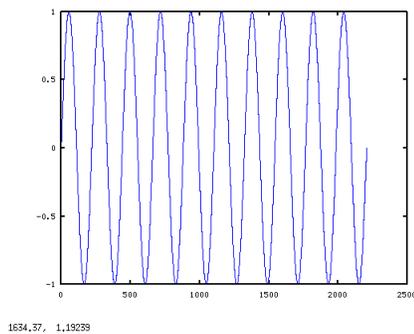
4.3 Filtros FIR e IIR

Da equação 8 podemos perceber que a saída depende tanto do sinal de entrada quando das saídas anteriores. É nessa dependência em relação às saídas anteriores que podem definir os filtros lineares em duas categorias:

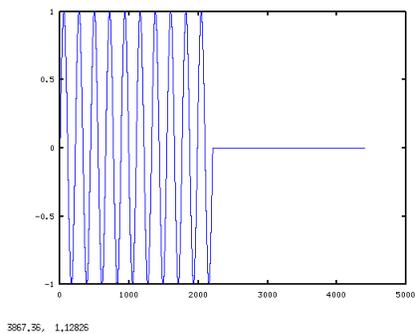
- FIR, *Finite Impulse Response*: Filtros FIR dependem somente do sinal de entrada, portanto todos os b_i são zero. Eles são sempre estáveis, ou seja, sua resposta impulsiva converge para zero. Isso acontece porque eventualmente o sinal de entrada termina e como o filtro não depende da saída em algum momento os $x(n - i)$ serão sempre 0 e a resposta impulsiva converge a zero.
- IIR, *Infinite Impulse Response*: Filtros IIR dependem do sinal de saída, eles tem um ou mais b_i diferente de zero. Eles não são necessariamente estáveis pois podem entrar num estado onde o *feedback* alimenta o filtro e a resposta impulsiva não converge para zero.

5 Processos Estacionários e Não Estacionários

Processos aleatórios podem ser classificados em duas grandes classes: os estacionários onde medidas como média e desvio padrão não mudam ao longo do tempo e os não-estacionários cujas medidas mudam. As figuras 3(a) e 3(b) são gráficos da amplitude pelo tempo de dois sinais, um estacionário outro não.



(a) Estacionário



(b) Não Estacionário

Normalmente podemos supor, em intervalos curtos de tempo, que o ruído e a voz são processos estacionários já que ambos não produzem mudanças bruscas na maior parte do tempo.

6 Correlação de processos aleatórios

A estatística tem um papel fundamental no processamento de sinais. Muitas das técnicas, inclusive as tratadas neste trabalho, necessitam de algumas informações do sinal original que normalmente não temos como conseguir, por isso precisam ser estimadas. Essa seção enuncia a correlação de processos aleatórios que representa um papel central no filtro de Wiener.

A correlação é uma medida de semelhança, ou dependência, entre dois processos e suas mudanças ao longo do tempo ou espaço, ou seja por um conjunto de realizações ou instâncias do sinal. Dois processos são independentes, ou não correlacionados, quando sua correlação é zero. A correlação cruzada entre os processos $x(m)$ e $y(m)$ é definido por:

$$\begin{aligned} r_{xy}(m_1, m_2) &= E[x(m_1), y(m_2)] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(m_1)y(m_2)f_{x(m_1),y(m_2)}(x(m_1), y(m_2))dx(m_1)dy(m_2) \end{aligned} \tag{9}$$

onde f é a fdp correspondente. A autocorrelação é a correlação cruzada de um processo com ele mesmo. Se for calculada ao longo do tempo ela pode ser entendida como uma medida de quão estacionário um sinal é, já que analisa a diferença entre um sinal e uma versão adiantada ou atrasada dele.

6.1 Processo Ergódicos

Como veremos mais adiante, calcular a correlação entre sinais digitais é difícil porque normalmente temos somente uma realização, deste processo então precisamos usar estimativas baseadas no processo ao longo do tempo e não ao longo do espaço, ou seja um conjunto de realizações do processo aleatório, o espaço amostral. Um processo estacionário é dito ergódico se ele exibe as mesmas características estatísticas ao longo do tempo em uma única amostra do processo do que de um conjunto de amostras. Ou seja para o cálculo da correlação, média e outros estimadores as dimensões espacial e temporal são intercambiáveis. Fazer essa restrição a um sinal de fala, ou ruído, é razoável se levarmos em conta um janela pequena onde o sinal se mantém num estado quase estacionário como comentado na seção 5.

7 Notação

A partir da próxima seção vamos utilizar as seguintes notações para facilitar a leitura:

- Vetores serão denotados por letras minúsculas em negrito, i. e. \mathbf{w} , \mathbf{r}_{yx} .
- Matrizes serão denotadas por letras maiúsculas, i. e. Y , R_{yy} .
- Números reais serão denotados por letras gregas minúsculas, i. e. α , λ .
- Funções serão denotadas por letras minúsculas, i. e. $f_{rr}(\mathbf{w})$, $\mathbf{y}(m)$.
- Elementos de um vetor serão denotados da seguinte forma: $x(m)$, representa o m^{esimo} elemento do vetor \mathbf{x} .

As exceções a essas regras serão deixadas claras no texto.

8 Filtro de Wiener

O Filtro de Wiener é um filtro de mínimos quadrados, ou seja, o seu objetivo seu é minimizar a esperança do erro quadrático entre o sinal de entrada e o sinal desejado. Como citado na introdução, filtros de mínimos quadrados, em particular os filtros de Wiener, são a base de diversas ramificações de processamento de sinal como predição linear, redução de ruído, cancelamento de eco entre outras. São filtros lineares e podem ser formulados tanto como um filtro FIR quanto como um filtro IIR. Nesse trabalho vamos analisar a versão FIR por ser mais simples e garantir a estabilidade do filtro como comentado na seção 4.3.

A equação abaixo representa a equação de filtro de Wiener:

$$\hat{x}(m) = \sum_{k=0}^{P-1} w_k y(m-k) \quad (10)$$

ou na sua forma vetorial $\hat{\mathbf{x}} = \mathbf{w}^t \mathbf{y}$ onde $\mathbf{w} = [w_0, w_1, \dots, w_{P-1}]$ representa o vetor de coeficientes do filtro e $\mathbf{y} = [y(m), y(m-1), \dots, y(m-P+1)]$ o vetor de entrada. Note que ela pode ser entendida como uma operação de convolução entre \mathbf{w} e \mathbf{y} , onde $y(m)$ indica o sinal recebido, w_k são os coeficientes do filtro e $\hat{x}(m)$ a estimativa de erro quadrático mínimo do sinal $x(m)$ original.

O erro do filtro de Wiener é dado pela diferença entre o sinal original, $x(m)$, e o sinal estimado pelo filtro $\hat{x}(m)$.

$$e(x) = x(m) - \hat{x}(m) = x(m) - \mathbf{w}^t \mathbf{y} \quad (11)$$

Repetindo a equação 11 em um segmento de N amostras obtemos o seguinte sistema linear:

$$\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(N-1) \end{bmatrix} = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} - \begin{bmatrix} y(0) & \cdots & y(1-P) \\ y(1) & \cdots & y(2-P) \\ \vdots & \ddots & \vdots \\ y(N-1) & \cdots & y(N-P) \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{P-1} \end{bmatrix} \quad (12)$$

A partir desta equação podemos tirar algumas conclusões sobre o tamanho P do filtro e sua relação com erro:

- Se temos $P = N$ então o sistema linear pode ser determinado e apresentar uma solução onde $e = 0$ e conseguiríamos restaurar o sinal por completo, ou o sistema é indeterminado. Esta escolha de P é inviável na prática porque, como N é grande e $P = N$, então P também seria grande e o custo computacional de resolver esta equação torna o processo muito custoso. Outro problema é que se o ruído não for linearmente separável do sinal original esse sistema pode não ter solução.
- Para $P > N$ temos um sistema indeterminado e não conseguiríamos obter uma solução única para $e = 0$, teríamos infinitas soluções para o sistema. Tem os mesmos problemas da situação anterior.
- Quando $P < N$ o sistema tem uma única solução, normalmente com erro não zero e tomamos \mathbf{w} tal que conseguimos minimizar $E^2[e(m)]$. Esse é o cenário que ocorre na prática e o que será considerado neste trabalho.

Primeiramente precisamos calcular $E[e^2(m)]$ onde $E[.]$ é a esperança estatística.

$$\begin{aligned} E[e^2(m)] &= E[(x(m) - \mathbf{w}^t \mathbf{y})^2] \\ &= E[x^2(m)] - 2\mathbf{w}^t E[\mathbf{y}x(m)] + \mathbf{w}^t E[\mathbf{y}\mathbf{y}^t] \mathbf{w} \\ &= r_{xx}(0) - 2\mathbf{w}^t \mathbf{r}_{\mathbf{y}\mathbf{x}} + \mathbf{w}^t R_{\mathbf{y}\mathbf{y}} \mathbf{w} \end{aligned} \quad (13)$$

onde R_{yy} é a matriz de autocorrelação de \mathbf{y} e \mathbf{r}_{yx} o vetor de correlação cruzada entre \mathbf{y} e \mathbf{x} . A equação 13 em \mathbf{w} é um parabolóide convexo, portanto seu mínimo local é o mínimo global. Para minimizar o erro podemos igualar o gradiente de 13 a zero e encontrar \mathbf{w} ótimo que minimiza o erro quadrático.

$$\frac{\partial}{\partial \mathbf{w}} E[e^2(m)] = -2\mathbf{r}_{yx} + 2\mathbf{w}^t R_{yy} \quad (14)$$

para $E[e^2(m)] = 0$ temos:

$$\mathbf{w} = R_{yy}^{-1} \mathbf{r}_{yx} \quad (15)$$

que é chamada de equação de Wiener-Hopf. Abrindo a equação vetorial temos o seguinte sistema:

$$\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{P-1} \end{bmatrix} = \begin{bmatrix} r_{yy}(0) & \cdots & r_{yy}(P-1) \\ r_{yy}(1) & \cdots & r_{yy}(P-2) \\ \vdots & \ddots & \vdots \\ r_{yy}(P-1) & \cdots & r_{yy}(0) \end{bmatrix} \begin{bmatrix} r_{yx}(0) \\ r_{yx}(1) \\ \vdots \\ r_{yx}(P-1) \end{bmatrix} \quad (16)$$

8.1 Calculando as correlações

A equação 15 mostra que para conseguirmos \mathbf{w} ótimo precisamos conseguir calcular a matriz de autocorrelação R_{yy} e o vetor de correlação cruzada \mathbf{r}_{yx} . Como foi dito na seção 6 para calcular a correlação entre dois processos aleatórios precisamos de diversas instâncias deste processo. Na prática só temos uma instância de cada sinal, a entrada \mathbf{y} e o sinal original \mathbf{x} . Como foi comentado na seção 6.1 podemos assumir que os sinais são ergodicamente correlacionados em blocos pequenos já que eles são quase estacionários dentro de um bloco. Assim conseguimos calcular as correlações utilizando a média pelo tempo em um bloco de tamanho N .

$$r_{yy}(k) = \sum_{m=0}^{N-1} y(m)y(m+k) \quad (17)$$

$$r_{yx}(k) = \sum_{m=0}^{N-1} y(m)x(m+k) \quad (18)$$

Calcular R_{yy} é simples, porque ele depende somente da entrada, mas \mathbf{r}_{yx} depende do sinal original \mathbf{x} que, normalmente, não temos. Como já foi dito

estamos considerando que o ruído \mathbf{r} é aditivo, ou seja, $y(m) = x(m) + r(m)$. Além disso assumimos que o ruído e o sinal não são correlacionados, portanto $\mathbf{r}_{\mathbf{xr}} = \mathbf{0}$. Com essas duas informações podemos calcular $\mathbf{r}_{\mathbf{yx}}$ da seguinte forma:

$$\begin{aligned}
 r_{yx}(m) &= E[y(m)x(m)] \\
 &= E[y(m)(y(m) - r(m))] \\
 &= E[y(m)y(m)] - E[(x(m) + r(m))r(m)] \\
 &= r_{yy}(m) - E[x(m)r(m)] - E[r(m)r(m)] \\
 &= r_{yy}(m) - r_{xr}(m) - r_{rr}(m) \\
 &= r_{yy}(m) - r_{rr}(m)
 \end{aligned} \tag{19}$$

$\mathbf{r}_{\mathbf{yy}}$ é a primeira coluna de R_{yy} . Como podemos assumir que o sinal ruidoso não muda muito de característica ao longo do tempo então podemos calcular $\mathbf{r}_{\mathbf{rr}}$ num período de silêncio do sinal \mathbf{y} onde somente o ruído é observado. Desta forma conseguimos calcular os coeficientes ótimos \mathbf{w} .

9 Filtros Adaptativos

Um dos problemas do filtro de Wiener é que ele assume que o sinal tratado é estacionário, o que não é verdade na maioria dos casos, inclusive sinais de fala. Para contornar esta limitação podemos utilizar alguma estratégia que recalcula periodicamente \mathbf{w} para refletir as mudanças no sinal. Nesta seção vamos apresentar três formulações adaptativas do filtro de Wiener: a adaptação por bloco, o RLS e o LMS com um pequeno comparativo das vantagens e desvantagens de cada método.

9.1 Adaptação por Bloco (BLS)

Adaptação por bloco, ou *Block Least Square*, se aproveita do fato de que sinais não-estacionários que mudam vagarosamente ao longo do tempo podem ser considerados estacionários dentro de uma janela de tempo pequena o suficiente. Com isso em mente podemos recalculá-los a cada nova janela de entrada e conseguimos um filtro adaptativo com poucas mudanças em relação ao filtro geral. Para isso precisamos de uma formulação um pouco diferente da apresentada na seção 8 porque não temos como calcular R_{yy} e $\mathbf{r}_{\mathbf{yx}}$ sem

o sinal inteiro. O filtro de Wiener em um bloco de tamanho N pode ser representado pela seguinte equação matricial:

$$\begin{bmatrix} \hat{x}(0) \\ \hat{x}(1) \\ \vdots \\ \hat{x}(N-1) \end{bmatrix} = \begin{bmatrix} y(0) & \cdots & y(1-P) \\ y(1) & \cdots & y(2-P) \\ \vdots & \ddots & \vdots \\ y(N-1) & \cdots & y(N-P) \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{P-1} \end{bmatrix} \quad (20)$$

ou em forma compacta como $\hat{\mathbf{x}} = Y\mathbf{w}$. O erro desta dentro desta janela é:

$$\mathbf{e} = \mathbf{x} - Y\mathbf{w} \quad (21)$$

o produto $\mathbf{e}^t\mathbf{e}$ representa o erro quadrático desse processo e é dado por:

$$\mathbf{e}^t\mathbf{e} = \mathbf{x}^t\mathbf{x} - \mathbf{x}^tY\mathbf{w} - \mathbf{w}^tY^t\mathbf{x} + \mathbf{w}^tY^tY\mathbf{w} \quad (22)$$

novamente tiramos o gradiente do erro quadrático em relação à w e igualamos a zero para encontrar \mathbf{w} que leva ao erro mínimo:

$$\frac{\partial \mathbf{e}^t\mathbf{e}}{\partial w} = -2\mathbf{x}^tY + 2\mathbf{w}^tY^tY = 0 \Leftrightarrow \mathbf{w} = (Y^tY)^{-1}Y^t\mathbf{x} \quad (23)$$

Observe que (Y^tY) , quando o sinal for considerado invariante durante a janela, é uma estimaco de R_{yy} calculado como mdia das amostras pelo tempo e $Y^t\mathbf{x}$ é uma estimativa de \mathbf{r}_{yx} com as mesmas condies. Portanto se considerarmos, novamente, que o processo é ergdico podemos ver que quando N tende ao tamanho do sinal original a formulao por blocos se aproxima da formulao original apresentada na seo 8. Novamente podemos usar a informao do rudo ser aditivo para calcular a estimativa de \mathbf{r}_{yx} fazendo:

$$Y^t\mathbf{x} = (\mathbf{y}_r - \mathbf{r}_{rr}) \quad (24)$$

onde \mathbf{y}_r é a primeira coluna de Y^tY , ou seja a estimativa de \mathbf{r}_{yy} .

9.2 RLS - Recursive Least Square

RLS é um filtro adaptativo por amostra que converge rapidamente para os coeficientes ótimos. Para sinais estacionários ele converge para os coeficientes ótimos do filtro de Wiener tradicional e para sinais não-estacionários ele se adapta as mudanas no sinal ao longo do tempo. A chave deste filtro é ser capaz de calcular $\mathbf{w}(m)$ a partir de $\mathbf{w}(m-1)$ eficientemente.

Para isso precisamos calcular $R_{yy}^{-1}(m)$ e $\mathbf{r}_{yx}(m)$ a partir de $R_{yy}^{-1}(m-1)$ e $\mathbf{r}_{yx}(m-1)$ respectivamente. Primeiro vamos considerar R_{yy} num bloco de N amostras que pode ser escrito como:

$$R_{yy}(m) = \sum_{m=0}^{N-1} \mathbf{y}(m)\mathbf{y}^t(m) = R_{yy}(m-1) + \mathbf{y}(m)\mathbf{y}^t(m) \quad (25)$$

Para incluir o fator de adaptabilidade é importante que o bloco atual tenha um peso maior do que o anterior então introduzimos um λ tal que $0 < \lambda < 1$ na equação resultando em:

$$R_{yy}(m) = \lambda R_{yy}(m-1) + \mathbf{y}(m)\mathbf{y}^t(m). \quad (26)$$

$\mathbf{r}_{yx}(m)$ pode ser calculado de forma similar resultando em:

$$\mathbf{r}_{yx}(m) = \lambda \mathbf{r}_{yx}(m-1) + \mathbf{y}(m)x(m) \quad (27)$$

Podemos escrever a equação acima em função de \mathbf{r}_{rr} lembrando que \mathbf{r}_{rr} muda pouco com o tempo então não precisa ser atualizado. Reescrevendo temos:

$$\mathbf{r}_{yx}(m) = \lambda \mathbf{r}_{yy}(m-1) + \mathbf{y}(m)y(m) - \mathbf{r}_{rr} \quad (28)$$

Para conseguir escrever $R_{yy}^{-1}(m)$ da mesma maneira precisamos usar o seguinte lema de inversão de matrizes:

Lema de Inversão de Matrizes

Seja A e B duas matrizes $P \times P$ positivas definidas relacionadas por:

$$A = B^{-1} + CD^{-1}C^t \quad (29)$$

onde D é uma matriz $N \times N$ positiva definida e C é uma matriz $P \times N$. A inversa de A , A^{-1} é definida por:

$$A^{-1} = B - BC(D + C^tBC)^{-1}C^tB \quad (30)$$

Ele pode ser provado multiplicando as equações 29 e 30.

Fazendo $A = R_{yy}(m)$, $B = \lambda^{-1}R_{yy}(m-1)^{-1}$, $C = \mathbf{y}(m)$ e $D = I$ onde I é a identidade temos que:

$$R_{yy}^{-1}(m) = \lambda^{-1}R_{yy}^{-1}(m-1) - \frac{\lambda^{-2}R_{yy}^{-1}(m-1)\mathbf{y}(m)\mathbf{y}^t(m)R_{yy}^{-1}(m-1)}{1 + \lambda^{-1}\mathbf{y}^t(m)R_{yy}^{-1}(m-1)\mathbf{y}(m)} \quad (31)$$

Multiplicando as equações 31 e 28 conseguimos escrever $\mathbf{w}(m)$ em função de $\mathbf{w}(m-1)$. Essa técnica tem uma taxa de convergência alta, mas tem uma complexidade de $O(NP^2)$ para processar um sinal de tamanho N , o que é muito lento para tempo real.

9.3 LMS - Least Mean Squared Error

Como mencionado na seção 8 a superfície do erro quadrático em relação a \mathbf{w} é um parabolóide com um ponto de mínimo que corresponde aos coeficientes ótimos do filtro de Wiener. Podemos usar esse fato para escolher um \mathbf{w} inicial e, para cada passo, introduzir um fator para que \mathbf{w} desça no parabolóide em direção ao novo valor ótimo, que muda porque o sinal não é estacionário. Isso é feito adicionando a cada passo um fator de $-\lambda \frac{\partial E[e^2(m)]}{\partial \mathbf{w}(m)}$ onde λ é o fator de adaptação para cada passo. Porém calcular a esperança do erro quadrático a cada iteração pode ser muito custoso então podemos sacrificar um pouco da convergência do método em favor da eficiência computacional formulando a adaptação da seguinte forma:

$$\mathbf{w}(m+1) = \mathbf{w}(m) - \lambda \left[\frac{\partial e^2(m)}{\partial \mathbf{w}(m)} \right] \quad (32)$$

Onde o erro $e(m)$ é dado por:

$$e(m) = x(m) - \mathbf{w}^t(m)\mathbf{y}(m) \quad (33)$$

Calculando o gradiente do erro quadrático instantâneo temos:

$$\frac{\partial e^2(m)}{\partial \mathbf{w}(m)} = -2\mathbf{y}(m)[x(m) - \mathbf{w}^t(m)\mathbf{y}(m)] = -2\mathbf{y}(m)e(m) \quad (34)$$

Substituindo na equação adaptativa temos:

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \lambda[\mathbf{y}(m)e(m)] \quad (35)$$

Uma vantagem clara deste filtro em relação ao RLS é a simplicidade dele e eficiência computacional, a cada passo a fase de adaptação toma $O(P)$ tempo e memória.

9.4 Comparação

Algumas vantagens e desvantagens dos métodos considerados já foram apresentadas, mas é interessante fazer um resumo comparativo entre eles:

- BLS: introduz poucas mudanças na formulação não-adaptativa dos filtros de Wiener; não funciona bem com sinais que variem muito rápido no tempo, ou seja, que sejam não-estacionários dentro da janela; computacionalmente caro, para cada janela é necessário resolver um sistema linear.

- RLS: converge rapidamente para o valor ótimo; é mais sensível a sinais com grande dinamicidade; muito custoso, não é indicado para aplicações em tempo real.
- LMS: mais simples de implementar; eficiente, toma tempo $O(P)$ onde P é o tamanho do filtro; tem uma taxa de convergência menor e pode ter uma diferença de resultado grande em relação ao RLS para sinais que mudem muito ou com muito ruído.

10 Alguns Parâmetros utilizados

Para analisar melhor a eficiência do filtro de Wiener no processo de redução de ruído em sinais de fala precisamos definir três parâmetros. O primeiro ajuda a entender a distorção ou a perda de clareza que é inerente ao processo do filtro de Wiener como foi citado na introdução enquanto o segundo parâmetro analisa a eficiência desse processo em termos de redução de ruído.

10.1 Índice de Distorção de Voz

Da equação 15 temos que $\hat{x}(m) = \mathbf{w}^t \mathbf{y}(m)$, podemos calcular então $E^2[\hat{x}(m)]$ para analisar como é o sinal restaurado:

$$E[\hat{x}^2(m)] = \mathbf{w}^t R_{yy} \mathbf{w} = \mathbf{w}^t R_{xx} \mathbf{w} + \mathbf{w}^t R_{rr} \mathbf{w} \quad (36)$$

ou seja conseguimos restaurar o sinal original, mas temos uma redução do sinal original num fator de $\mathbf{w}^t R_{rr} \mathbf{w}$ que representa o ruído residual. A presença desse fator implica numa atenuação de partes do sinal original devido ao processo de filtragem. Para medir melhor essa atenuação vamos definir a seguinte medida:

$$i_{dv}(\mathbf{w}) = \frac{E[(x(m) - \mathbf{w}^t \mathbf{x}(m))^2]}{E[r^2(m)]} \quad (37)$$

Onde $\mathbf{x}(m)$ é o equivalente de $\mathbf{y}(m)$ para o sinal original. Essa fórmula representa a relação entre a atenuação do sinal original, ou seja a distorção, em relação ao ruído presente no sinal de entrada. É dada em relação ao ruído para possibilitar a comparação em casos em que o ruído não é constante. Quanto maior o i_{dv} mais distorcido é o sinal. Ele pode ser entendido também como o erro de aplicar o filtro de Wiener usando o sinal original como sinal de entrada.

10.2 Fator de Redução de Ruído

O fator de redução de ruído é definido de maneira similar ao índice de distorção de voz e é uma normalização do erro quadrático pelo erro quadrático do ruído total:

$$f_{rr}(\mathbf{w}) = \frac{E[(x(m) - \mathbf{w}^t \mathbf{y}(m))^2]}{E[r^2(m)]} \quad (38)$$

Quanto menor f_{rr} mais o ruído foi reduzido.

11 Filtro de Wiener com Trade-off

Como foi mostrado na seção passada o filtro de Wiener ótimo leva a máxima distorção na clareza e existe uma relação direta entre o quanto de ruído conseguimos reduzir e o quanto a fala é distorcida. Com isso em mente a ideia de encontrar um filtro sub-ótimo, no sentido da redução de ruído, que ofereça o melhor *trade-off* entre redução de ruído e clareza é natural.

Seja o filtro sub-ótimo:

$$\mathbf{w}_s = \mathbf{u}_1(1 - \alpha) + \alpha \mathbf{w}_o \quad (39)$$

onde $\mathbf{u}_1 = [1, 0, 0, \dots, 0]$ é o filtro canônico que não altera a entrada. Podemos entender melhor o funcionamento desse filtro aplicando ele à entrada:

$$\begin{aligned} \mathbf{w}_s^t \mathbf{y} &= (1 - \alpha) \mathbf{u}_1^t \mathbf{y} + \alpha \mathbf{w}_o^t \mathbf{y} \\ &= (1 - \alpha) \mathbf{y} + \alpha \hat{\mathbf{x}} \end{aligned} \quad (40)$$

Como pode-se observar o sinal estimado pelo filtro sub-ótimo é formado por duas partes: uma inalterada igual a entrada, e outra com o ruído reduzido. Vamos agora calcular o erro quadrático do filtro:

$$E[(x(m) - \mathbf{w}_s^t \mathbf{y}(m))^2] = E[r^2(m)] - \alpha(2 - \alpha) \mathbf{r}_r^t R_{yy}^{-1} \mathbf{r}_r \quad (41)$$

Da equação 39 temos que $\mathbf{w}_s = \mathbf{w}_o$ se $\alpha = 1$ e para obter redução de ruído precisamos escolher α tal que $0 \leq \alpha \leq 2$ para garantir que $E[(x(m) - \mathbf{w}_s^t \mathbf{y}(m))^2] < E[r^2(m)]$.

Vamos calcular o i_{dv} para o filtro sub-ótimo:

$$\begin{aligned}
i_{dv}(\mathbf{w}_s) &= \frac{E[(x(m) - \mathbf{w}^t \mathbf{x}(m))^2]}{E[r^2(m)]} \\
&= \frac{E[(x(m) - (\mathbf{u}_1(1 - \alpha) + \alpha \mathbf{w})^t \mathbf{x}(m))^2]}{E[r^2(m)]} \\
&= \frac{E[(x(m) - \mathbf{u}_1^t \mathbf{x}(m) + \alpha \mathbf{x}(m) - \alpha \mathbf{w}^t \mathbf{x}(m))^2]}{E[r^2(m)]} \\
&= \frac{E[(\alpha(x(m) - \mathbf{w}^t \mathbf{x}(m)))^2]}{E[r^2(m)]} \\
&= \alpha^2 i_{dv}(\mathbf{w}_o)
\end{aligned} \tag{42}$$

como podemos ver a relação entre o $i_{dv}(\mathbf{w}_o)$ e $i_{dv}(\mathbf{w}_s)$ é dada somente por α . Isso implica que para $i_{dv}(\mathbf{w}_o) > i_{dv}(\mathbf{w}_s)$ precisamos de $-1 < \alpha < 1$, juntando com a restrição anterior temos que $0 < \alpha < 1$. Com $\alpha = 0$ temos que $\mathbf{w}_s = \mathbf{u}_1$ e não temos redução de ruído e com $\alpha = 1$ temos $\mathbf{w}_s = \mathbf{w}_o$ e máxima distorção.

Precisamos encontrar α que ofereça o melhor *trade-off* entre redução de ruído e distorção, para isso vamos analisar a seguinte relação:

$$\frac{1 - f_{rr}(\mathbf{w}_s)}{1 - f_{rr}(\mathbf{w}_o)} = 2\alpha - \alpha^2 \tag{43}$$

Usamos $1 - f_{rr}$ por conveniência, já que quando $1 - f_{rr} = 1$ temos a máxima redução de ruído. Esse formato ajuda a definir a seguinte função custo para encontrar α que maximize o *trade-off*:

$$c(\alpha) = \frac{1 - f_{rr}(\mathbf{w}_s)}{1 - f_{rr}(\mathbf{w}_o)} - \frac{i_{dv}(\mathbf{w}_s)}{i_{dv}(\mathbf{w}_o)} = 2(\alpha - \alpha^2) \tag{44}$$

$c(\alpha)$ é um polinômio do segundo grau em α e é fácil ver que seu ponto máximo é de $\alpha = \frac{1}{2}$.

12 Implementação

A implementação foi feita em C++ utilizando duas bibliotecas abertas auxiliares: o it++ [4], uma biblioteca de álgebra linear que traz uma implementação de vetores e matrizes parecidas com a sintaxe do Octave [5] facilitando a codificação e legibilidade. E a biblioteca libsndfile [6], uma biblioteca

para manipulação de arquivos .wav, formato de áudio utilizados nos experimentos.

Dentre as técnicas apresentadas foram escolhidas duas para serem comparadas, o filtro de Wiener tradicional e a versão adaptativa LMS. O filtro de Wiener foi escolhido como base de comparação e para demonstrar na prática a relação entre distorção e redução de ruído já que a análise foi feita em relação a essa versão do filtro. O LMS foi escolhido porque oferece uma boa taxa de convergência e baixo custo computacional, o baixo custo é muito importante já que a aplicação principal dessa análise é em telefonia e VoIP, e o custo do RLS seria proibitivo para tempo real, em suma ele oferece o melhor custo-benefício dentre os filtros adaptativos citados.

Para simplificar o código a aplicação não foi desenvolvida em tempo real, ela analisa o arquivo .wav de uma vez só. Outra simplificação foi em relação à estimativa do ruído, estimar o ruído através do sinal de entrada é um problema complicado então foi utilizado o próprio ruído como estimativa. A implementação segue as versões dadas em relação à r_{rr} que seria a informação estatística retirada do estimador de ruído.

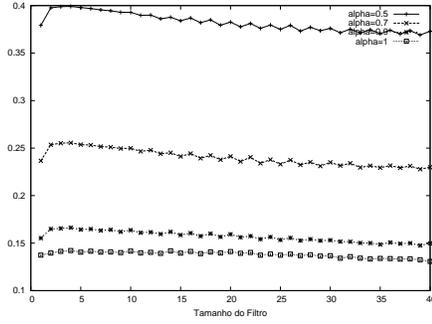
13 Experimentos

Os experimentos consistem numa comparação entre os índices f_{rr} e i_{dv} para o filtro de Wiener com *trade-off* para diversos α inclusive $\alpha = 1$ que corresponde ao filtro ótimo. O mesmo foi feito para os outros filtros com *trade-off*. Seguem os resultados nos gráficos 3, 4, 5 e 6.

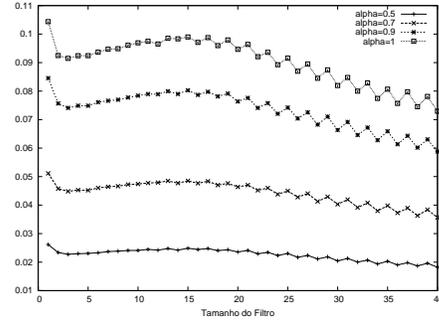
α	Wiener	LMS	BLS	RLS
10	0.167	0.246	0.251	0.930
20	0.283	0.340	0.350	3.382
30	0.330	0.379	0.555	7.926

Tabela 1: Tempo de Execução

Para o filtro de Wiener tradicional o resultado segue o esperado da análise feita na seção 11, quanto menor o α menor a distorção no sinal de entrada e menos ruído é reduzido. O filtro BLS segue a relação esperada entre os índices e o α , mas produz resultados muito ruins. Na prática para que se obtenha um resultado viável com esse filtro devemos ter um tamanho de bloco muito

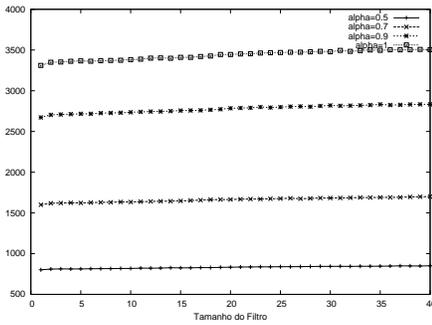


(c) Wiener - f_{rr}

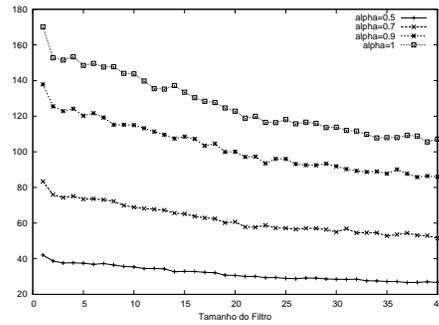


(d) Wiener - i_{dv}

Figura 3: Índices para Wiener, diversos α



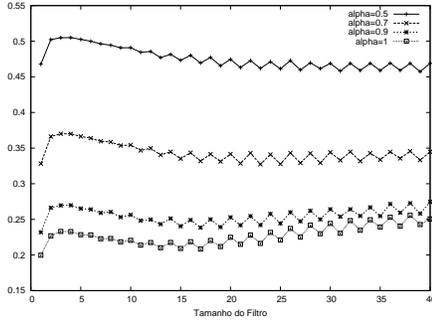
(a) BLS - f_{rr}



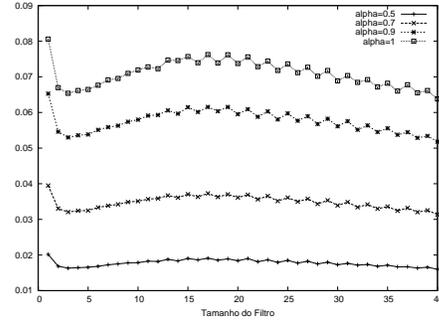
(b) BLS - i_{dv}

Figura 4: Índices para BLS, diversos α

elevado o que o transforma no filtro de Wiener convencional. O filtro RLS também seguiu o esperado em relação aos índices, mas não obteve resultados satisfatórios sendo comparável ao filtro convencional. Outra desvantagem já citada do filtro RLS é o tempo de execução. A tabela 1 traz uma comparação do tempo de execução dos filtros em relação à alguns valores de P . Como podemos observar o filtro RLS é muito menos eficiente que os outros. Os

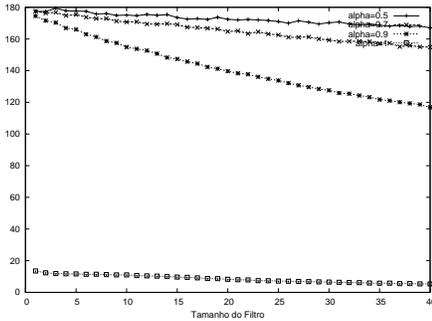


(a) RLS - f_{rr}

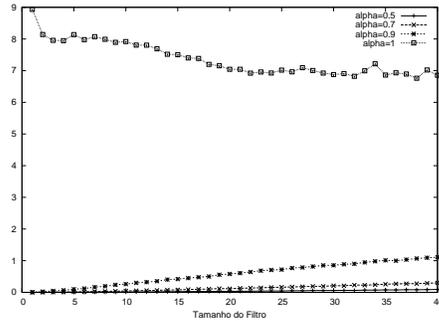


(b) RLS - i_{dv}

Figura 5: Índices para RLS, diversos α



(a) LMS - f_{rr}



(b) LMS - i_{dv}

Figura 6: Índices para LMS, diversos α

resultados do LMS são mais interessantes, porque dá pra ver que o impacto na convergência do filtro ao introduzir um *trade-off*, mesmo que de 0.9, é bem significativo. A distância dos valores para $\alpha = 1$ dos outros é bem grande, principalmente no f_{rr} , isso mostra que utilizar o *trade-off* junto com este filtro não é muito recomendável, porque a perda de eficiência na redução de ruído é muito drástica e não compensa a diminuição da distorção que é

comparável ao filtro de Wiener tradicional com *trade-off* de 0.7.

Parte II

Parte Subjetiva

14 Dificuldades e Frustrações

A grande dificuldade foi também a maior frustração, o fato de que não consegui uma maneira boa de estimar o ruído a partir do sinal de entrada. As técnicas apresentadas necessitam de uma estimativa boa de r_{rr} para serem eficazes, porém foi difícil encontrar uma bibliografia boa sobre esse tema, os poucos artigos que encontrei tratavam de técnicas muito complexas que não teria tempo de introduzir no trabalho. Encontrei algumas dificuldades na escolha da linguagem utilizada. A primeira tentativa foi com Java, mas não encontrei bibliotecas de álgebra linear satisfatórias. A alternativa foi migrar para o c++ por causa do it++ que é uma excelente biblioteca.

15 Matérias Relacionadas

15.1 MAC0122 - Princípios de Desenvolvimento de Algoritmos

Primeiro contato com C, noções de complexidade computacional e ver maneiras diferentes de abordar um problema. Esses foram os conceitos passados nessa disciplina mais relevantes para mim.

15.2 MAE0121 - Introdução a Probabilidade e a Estatística I

Conceitos básicos de estatística, como distribuições probabilísticas e correlação.

15.3 MAE0228 - Noções de Probabilidade e Processos Estocásticos

Muitas técnicas de processamento de som utilizam modelos marcovianos e estocásticos então essa matéria foi muito importante, além de reforçar alguns conceitos básicos.

15.4 MAT0221 - Cálculo Diferencial e Integral IV

Transformada de Fourier é parte chave do ferramental de processamento de sinais. Esse assunto foi abordado nessa matéria, mas só entendi ele melhor em MAC0337.

15.5 MAT0139 - Álgebra Linear para Computação

Sinais digitais são grandes vetores reais, portanto álgebra linear faz parte da análise de diversas técnicas.

15.6 MAC0300 - Métodos Numéricos da Álgebra Linear

Como foi dito sinais digitais são grandes vetores de números reais então saber trabalhar com matrizes e garantir estabilidade numérica foi muito útil.

15.7 MAC0337 - Computação Musical

Matéria mais importante para esse trabalho. Abordou conceitos como equações de filtro, transformada de Fourier, filtros FIR e IIR, digitalização de sinais, entre outros conceitos base em processamento de sinais.

16 Aplicação de Conceitos

Dentre as matérias que tiveram uma aplicação direta nesse trabalho eu quero destacar MAC0377, MAT300 e MAE0228 que foram essenciais e ter cursado elas facilitou muito fazer esse trabalho. Todas as matérias de MAC foram importantes, porque os diversos métodos e abordagens de como resolver um problema me ajudaram a entender os novos caminhos que podiam ser tomados.

Referências

- [1] <http://en.wikipedia.org/wiki/File:Pcm.svg>.
- [2] <http://en.wikipedia.org/wiki/File:AliasingSines.svg>.
- [3] http://en.wikipedia.org/wiki/Vocal_range.
- [4] <http://sourceforge.net/apps/wordpress/itpp/>.
- [5] <http://www.gnu.org/software/octave/>.
- [6] <http://www.mega-nerd.com/libsndfile/>.
- [7] Y. Ephraim e H. L. Van Trees. A signal subspace approach for speech enhancement. 1995.
- [8] Y. Huang e S. Doclo J. Chen, J. Benesty. New insights into the noise reduction wiener filter. 2006.
- [9] Yiteng Huang e Israel Cohen Jacob Benetsy, Jingdong Chen. *Noise Reductino in Speech Processing*. Springer, 2009.
- [10] F. Richard Moore. *Elements of Computer Music*. P T R Prentice Hall, 1990.
- [11] Saeed V. Vaseghi. *Multimedia Signal Processing*. John Wiley and Sons Ltd., 2007.
- [12] Saeed V. Vaseghi. *Advanced Signal Processing and Noise Reduction*. John Wiley and Sons Ltd., 4th edition, 2008.