



Universidade do Minho
Escola de Engenharia

*Mestrado Integrado em Engenharia Eletrónica Industrial e
Computadores*

2019/2020

Processamento Digital de Sinal

**DIMINUIÇÃO DA FREQUÊNCIA DE AMOSTRAGEM POR MEIOS
DIGITAIS EVITANDO A OCORRÊNCIA DE ALIASING**

FILTRO ELÍPTICO



Rafael Andrade Silva, A74437

Docente:

Professor Carlos Manuel Gregório Santos Lima



Índice

1 Introdução	3
2 Fundamentos Teóricos	4
2.1 Decimação	4
2.2 Filtro Passa-baixo	6
2.3 Filtros Digitais	6
2.4 Filtros FIR	7
2.5 Filtro Elíptico	8
3. Análise do Problema	11
4. Implementação do código	12
4.1 Gravação do Áudio	12
4.2 Especificações do Filtro	12
4.3 Criação do Filtro Elíptico	13
4.4 Aplicação do Filtro	13
4.5 Downsampling e Formação do Sinal Filtrado	14
5. Testes e Resultados da Implementação	15
5.1 Teste com fator de subamostragem =3	15
5.2 Teste com fator de subamostragem =6	17
5.3 Teste com fator de subamostragem =9	18
6. Conclusão	20
7. Índice de Figuras	21
9. Referências	22



1 Introdução

Este trabalho foi realizado no âmbito da unidade curricular de Processamento Digital de Sinal, sob a orientação do professor Carlos Lima com o intuito de permitir a diminuição da frequência de amostragem por meios digitais, evitando a ocorrência de “aliasing”.

Como tal, foi-nos proposto implementar em Matlab um módulo que permitisse diminuir a frequência de amostragem por meios digitais evitando a ocorrência de “aliasing”, processo normalmente designado por “Downsampling”. O propósito deste procedimento é compactar ao máximo o sinal de modo a ocupar menos memória e exigir menos cálculos em processamento de sinal como por exemplo, filtragem. Este processo de decimação obtido pela amostragem do sinal já digitalizado requer uma ação de filtragem passa-baixo para evitar o “aliasing”, processo que deve ser efetuado recorrendo a um dos métodos de filtragem estudados. Os métodos de filtragem de estudo foram: Butterworth, Chebyshev (tipo I ou II) Elíptico, Janela de kaiser, Parks-McClellan.

Para um melhor entendimento do objetivo da realização deste trabalho prático, inicialmente, este relatório contém uma explicação e consolidação de conceitos relevantes ao trabalho, uma análise pormenorizada ao trabalho requerido e uma descrição da implementação bem como testes realizados para uma melhor reflexão sobre o realizado.

Neste trabalho a filtragem é realizado por um filtro FIR, mais propriamente pelo filtro Elíptico. O programa utilizado na elaboração destes procedimentos, tal como referenciado anteriormente, é o MatLab.

2 Fundamentos Teóricos

2.1 Decimação

Suponha-se que se tem uma versão amostrada de um sinal analógico que teve uma frequência de amostragem muito alta. A decimação pode ser muito útil nesta situação. A decimação é um método para diminuir a frequência de amostragem por um fator inteiro (D), ou seja, e a operação correspondente a redução da frequência de amostragem de um sinal digital, sendo este um sinal amostrado a partir de um sinal contínuo, a uma determinada frequência de amostragem. Através do processo de decimação somos capazes de reduzir as amostras, sendo o sinal original recuperável visto que a decimação é um processo reversível.

Este processo é vantajoso para compactar os dados, de modo a ser ocupada menos memória para armazenamento do sinal, sendo que o processo de decimação tem que cumprir determinados requisitos.

Este dispositivo tem como sinal de entrada o sinal $x[n]$ (sinal amostrado do sinal contínuo), e um fator D , que é responsável por gerar um sinal de saída $y[m]$, com uma taxa de amostragem D vezes menor que a taxa de amostragem do sinal amostrado na entrada. No caso de tempo discreto, considerando x_d o sinal que já sofreu decimação, pode-se considerar o seguinte teorema:

$$x_d(n) = x(Dn)$$

O processo de decimação está dividido em duas etapas como ilustrado na figura abaixo. Pode ser explicada como um processo de 2 passos: primeiro, reduz-se as componentes do sinal com alta frequência através de um filtro passa-baixo; segundo, realiza-se “Downsample” no sinal filtrado.

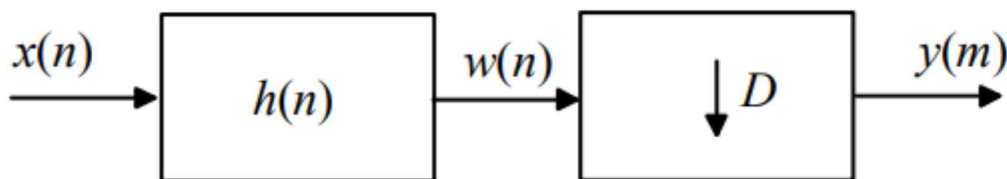


Figura 1- Modelo de Decimação

O processo de “Downsampling” por si só causa que componentes de sinal de alta frequência sejam sobrepostos por outros dados.

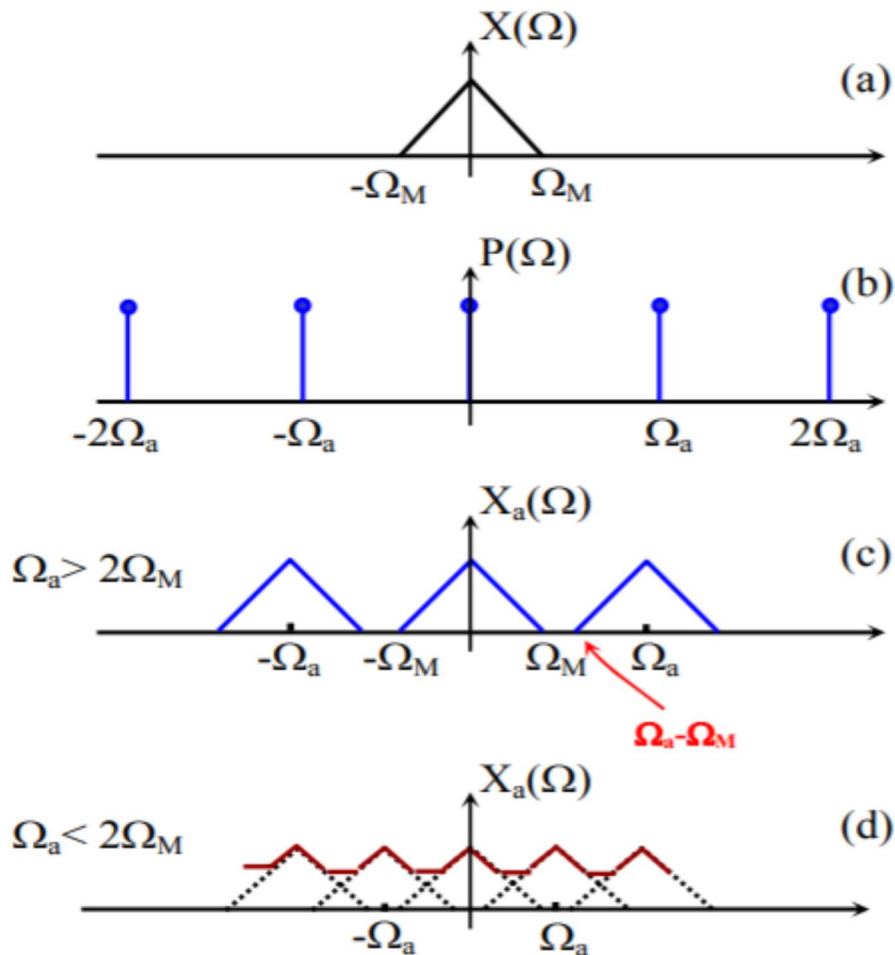


Figura 2- Exemplo de Decimação

Observando a figura anterior verificamos que reduzindo a frequência de amostragem através da seleção de valores $x[n]$ para múltiplos de D , as cópias do espectro de $x(n)$ podem sobrepor-se, causando a perda no formato do espectro original, este fenómeno tem o nome de “aliasing”. Desta forma, para evitar o “aliasing”, é necessário reduzir a largura de banda de $x[n]$ para:

$$\Omega_{m\acute{a}x} = \frac{\pi}{D}$$

Assim sendo o espectro original desaparece e não pode ser mais recuperado a partir do sinal amostrado, se as condições forem satisfeitas o espectro original é mantido e pode ser recuperado através de uma filtragem por um filtro passa-baixo.

Normalmente, a aplicação do filtro é realizada para suprimir este fenómeno garantidamente. Esse filtro pode ser designado por filtro anti-aliasing e é caracterizado pela resposta ao impulso $h[n]$ e uma resposta de frequência $H_d(\Omega)$ que satisfaz a seguinte condição:

$$H_D(\Omega) = \begin{cases} 1, & |\Omega| \leq \pi D \\ 0 & \text{demais casos} \end{cases}$$

2.2 Filtro Passa-baixo

Um filtro passa-baixo permite a passagem de sinais de baixas frequências e atenua ou reduz sinais com amplitude superior à frequência de corte.

Numa filtragem passa baixo, podemos identificar três bandas de frequências distintas:

- **Banda Passante:** corresponde às frequências do sinal de entrada que passam para a saída sem atenuação ou com atenuação ligeira.
- **Banda de corte:** corresponde à gama de frequências do sinal de entrada que são rejeitadas pelo filtro.
- **Banda de Transição:** zona intermédia entre as duas anteriores, nesta zona o comportamento do filtro varia entre a atenuação ligeira e a rejeição do sinal de entrada.

2.3 Filtros Digitais

Um filtro digital é um filtro que processa sinais digitais.

Utilizando um conversor analógico-digital (ADC), digitaliza-se o sinal a filtrar, em seguida este é processado por algum tipo de processador, onde está programado o filtro digital que usa esse processador digital para fazer os cálculos numéricos sobre os valores

de amostragem do sinal. Para estes cálculos utiliza-se a multiplicação dos valores de entrada por constantes e por soma de produtos.

Para se obter de novo um sinal analógico já filtrado, coloca-se um conversor digital-analógico (DAC).

Os filtros digitais podem ser divididos em duas classes, conforme a duração da sequência de saída, quando aplicado na sua entrada um sinal de impulsos.

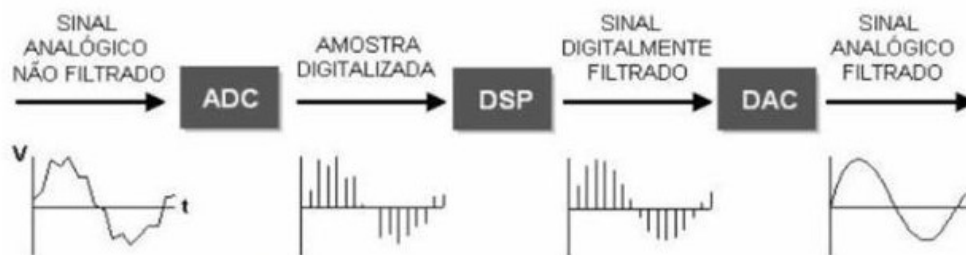


Figura 3- Esquema Filtro Digital

2.4 Filtros FIR

Um filtro FIR ou de resposta ao impulso finita é um tipo de filtro digital caracterizado por uma resposta ao impulso que se torna nula após um tempo finito, em contraste com os filtros IIR.

Os filtros FIR sempre são estáveis, e com fase linear (atraso no tempo)

Métodos comuns para o projeto de filtros FIR:

- Janelas, usando a resposta ao impulso dos filtros ideais
- Amostragem em frequência
- Projeto iterativo baseado em restrições ótimas

Para um sistema FIR de ordem M:

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k]$$

Com função de transferência:

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k}$$

E resposta ao impulso:

$$h[n] = \begin{cases} b_n, & 0 \leq n \leq M-1 \\ 0, & \text{caso contrario} \end{cases}$$

2.5 Filtro Elíptico

Para a elaboração do projeto será projetado um filtro Elíptico que apresenta uma vantagem relativamente a outros filtros:

- Possuem equiripple na banda passante e de corte
- Alcançam a mínima ordem N para uma dada especificação, ou seja, alcançam a mais estreita banda de transição para uma dada ordem N

Um **filtro elíptico** (também conhecido como um **filtro Cauer**) é um filtro de processamento de sinal com equalizada ondulação (equiripple) comportamento tanto na banda de passagem e a faixa de rejeição. A quantidade de ondulações em cada banda é ajustável de forma independente, e nenhum outro filtro de igual ordem pode ter uma transição rápida em ganho entre a faixa de passagem e a faixa de rejeição, para os valores dados de ondulação (quer a ondulação é igualada ou não).

O ganho de um passa-baixo do filtro elíptico como uma função de frequência angular ω é dada por:

$$G_n(\omega) = \frac{1}{\sqrt{1 + \epsilon^2 R_n^2(\xi, \frac{\omega}{\omega_0})}}$$

em que R_n é o n -th-fim função racional elíptica (algumas vezes conhecida como uma função racional Chebyshev) e:

- ω_0 representa a frequência de corte
- ϵ é o factor de ondulação
- ξ é o factor de selectividade

O valor do factor de ondulação especifica a ondulação da banda passante, enquanto que a combinação do factor de ondulação e o factor de selectividade especifica a ondulação de banda atenuada.

Propriedades:

- Na banda passante, a função racional elíptica varia entre zero e um. O ganho da banda passante, por conseguinte, irá variar entre 1 e $1/\sqrt{1 + \epsilon^2}$.
- Na faixa de rejeição, a função racional elíptica varia entre infinito e o factor de discriminação que é definida como: $L_n = R_n(\xi, \xi)$

O ganho da banda atenuada, por conseguinte, irá variar entre 0 e $1/\sqrt{1 + \epsilon^2 L_n^2}$.

- No limite da função racional elíptica torna-se um polinómio Chebyshev, e, por conseguinte, o filtro torna-se um filtro do tipo Chebyshev I, com um factor de ondulação $\epsilon \xi \rightarrow \infty$
- Uma vez que o filtro de Butterworth é uma forma limitante do filtro Chebyshev, segue-se que, no limite de, e de tal modo que o filtro se torna um filtro de Butterworth:

$$\omega_0 \rightarrow 0 \epsilon \rightarrow 0 \epsilon R_n(\xi, 1/\omega_0) = 1$$

- No limite de, e de tal modo que e, o filtro torna-se um filtro de tipo II Chebyshev com ganho:

$$\begin{aligned} \xi \rightarrow \infty \epsilon \rightarrow 0 \omega_0 \rightarrow 0 \xi \omega_0 &= 1 \\ \epsilon L_n &= \alpha \\ G(\omega) &= \frac{1}{\sqrt{1 + \frac{1}{\alpha^2 T_n^2(1/\omega)}}} \end{aligned}$$

No gráfico a baixo podemos observar um exemplo da resposta em frequência de um filtro passa baixo elíptico quarta ordem com $\epsilon = 0,5$ e $\xi = 1,05$. Também são mostrados o ganho mínimo na faixa de passagem e o ganho máximo na faixa de rejeição, e a região de transição entre a frequência normalizada e uma ξ .

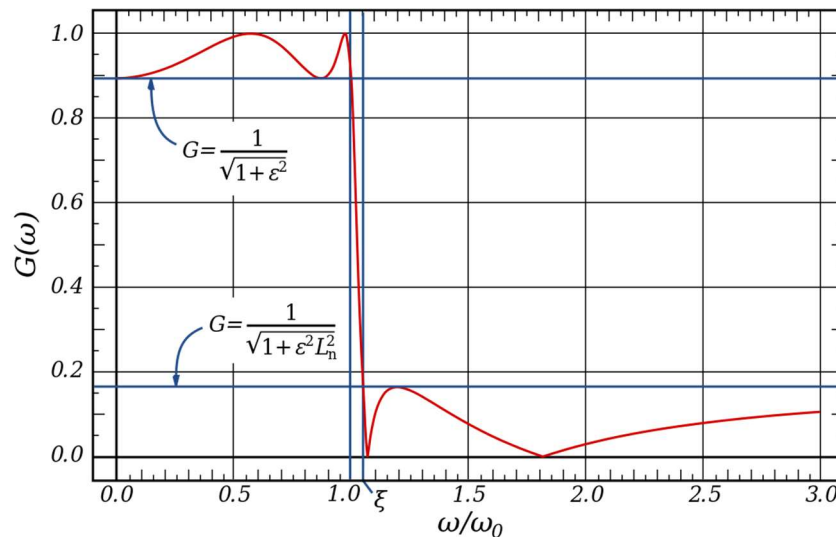


Figura 4- Resposta em frequência de um filtro passa baixo elíptico quarta ordem

Na imagem em baixo é mostrado o filtro elíptico ao lado de outro tipo comum de filtros obtidos com o mesmo número de coeficientes:

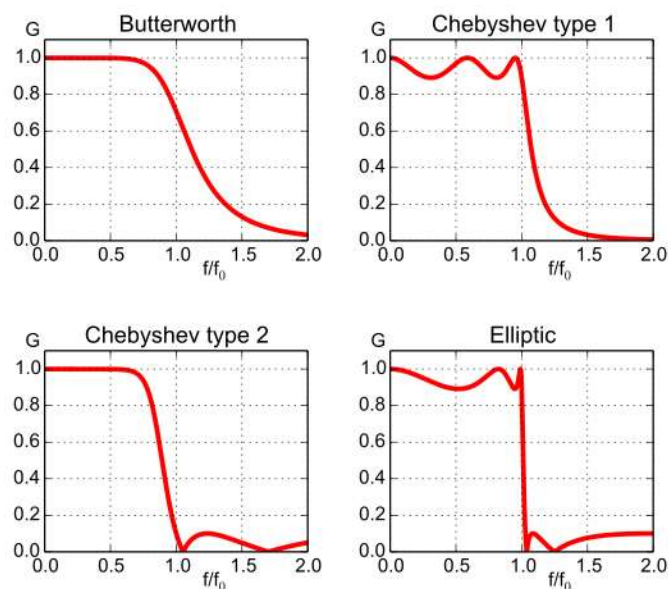


Figura 5- Comparação entre filtros usando o mesmo número de coeficientes

Como é evidente a partir da imagem, filtros elípticos são mais nítidos do que todos os outros, mas eles mostram ondulações em toda a largura de banda.

3. Análise do Problema

Como já descrito anteriormente, é necessário realizar uma decimação através de uma função que recebe como parâmetros de entrada um sinal de áudio amostrado a 8kHz e o fator de subamostragem. Esta função deve devolver um áudio amostrado a $\frac{F_s}{N}$.

Este trabalho será implementado usando o programa MATLAB.

Quanto ao filtro passa-baixo, o atribuído foi o filtro Elíptico com ripple na banda passante de 40 dB, ripple na banda de rejeição de 60dB e uma largura de banda de transição de 20% da banda passante.

Deve-se também chamar a função recursivamente para valores diferentes de N (ordem crescente), de modo a gerar os diferentes valores do coeficiente de subamostragem e verificar apenas o áudio para se poder identificar a perda sucessiva de conteúdo espectral do sinal.

4. Implementação do código

4.1 Gravação do Áudio

No início do programa foram executados os seguintes comandos de forma a gravar o áudio como se pode verificar na figura 6. Como resultado foi criado um ficheiro do tipo .wav, ficheiro este que consiste num sinal áudio de 4 segundos, o qual é transformada num array para posteriormente realizar o estudo do sinal.

```
%Criação do Som
gravacao=audiorecorder; %criar o Objecto
disp('Início da Gravação.')
recordblocking(gravacao,4); %gravação do audio por 4 segundos
disp('Fim da Gravação.')
play(gravacao); %reprodução do Objecto (gravação)
y=getaudiodata(gravacao); %conversão do Objecto da gravação para um Array
audiowrite('som.wav',y,8000); %escrita de um ficheiro .wav com o Array
[y,Fs]=audioread('som.wav'); %leitura do ficheiro .wav
```

Figura 6- Código em Matlab para a criação do Som

4.2 Especificações do Filtro

Depois do sinal já gravado, com os dados fornecidos define-se a frequência de amostragem, os valores da largura da banda de transição e por fim, especifica-se ripple da banda passante e a atenuação da banda de rejeição. Pode-se observar a declaração destes valores na figura 7.

```
%fator de subamostragem
N=10;

%frequencia de amostragem
%Fs=8000;

%Informações do som
info=audioinfo('som.wav');
disp(info);
[y,Fs]=audioread('som.wav'); %leitura do ficheiro .wav

%Características do filtro
Rp=20*log(1.1); %ripple na banda passante
Rs=60; %ripple na banda de rejeição
banda_pass=4000/N;
banda_rej=banda_pass+0.2*banda_pass; %valores da banda de transição
wp=(banda_pass)/(Fs/2); %frequencia de corte
ws=(banda_rej)/(Fs/2); %frequencia de banda
```

Figura 7- Código em Matlab para determinação dos parâmetros do Filtro

4.3 Criação do Filtro Elíptico

Obtendo-se os parâmetros do filtro desejado precedeu-se a sua criação. Considera-se para uma frequência de corte inversamente proporcional à medida que o fator cresce para que, deste modo, se garanta a não existência de “aliasing” no sinal.

É reproduzido numa figura o filtro criado.

```
[n,Wp]=ellipord(wp,ws,Rp,Rs);  
[b,a]=ellip(n,Rp,Rs,Wp); % Excerto de código para criação do filtro  
figure(1)  
freqz(b,a)  
title('Filtro Elíptico')
```

Figura 8- Código em Matlab para criação do Filtro

4.4 Aplicação do Filtro

Procedeu-se à transformação bilinear do filtro, para que, recorrendo à função “filter(b,a,y)”, se possa passar o filtro no sinal introduzido inicialmente. Esta função necessita dos coeficientes do numerador e de denominador, obtidos através da transformação bilinear, e do array que contém a data do sinal introduzido.

De seguida, criou-se uma figura que revela os dois gráficos. Nesta figura, é possível comparar o espectro dos dois sinais, o sinal original de entrada e o sinal filtrado.

```
%Aplicação do filtro  
sound_filtered=filter(b,a,y);  
figure(2)  
plot(y); %Gráfico com Sinal de Entrada  
title('Sinal Original e Sinal Filtrado')  
hold on %Mantém o plot atual  
plot(sound_filtered,'g'); %Gráfico com Sinal de filtrado a verde(g)  
legend('Sinal Original de Entrada','Sinal Filtrado');  
hold on %Mantém o plot atual
```

Figura 9- Código em Matlab para aplicação e utilização do Filtro

4.5 Downsampling e Formação do Sinal Filtrado

Por fim, procedeu-se ao “downsample” do sinal filtrado pelo fator de subamostragem, segundo passo no processo de decimação, como já foi mencionado anteriormente. Chama-se a função “downsample” e esta função recebe como parâmetros de entrada, o sinal filtrado e o fator de subamostragem.

Depois, cria-se uma figura e desenha-se o sinal filtrado, alterado por “downsample”.

```
result_sound=downsample(sound_filtered,N); %Downsample do som de entrada filtrado
figure(3)
plot(y);
hold on %Mantém o plot atual
legend('Som de entrada');
title('Sinal após Downsample')
plot(result_sound, 'r')
legend('Sinal Original de Entrada', 'Sinal Filtrado');
```

Figura 10- Janela com Sinal Original e Filtrado após o Downsample

5. Testes e Resultados da Implementação

Para testar o trabalho desenvolvido, realizaram-se 3 testes diferentes com o mesmo áudio, mas diferentes fatores de subamostragem para se poder verificar os diferentes resultados.

Inicialmente fez-se a gravação de um som de uma voz. O som foi gravado a uma frequência de 8kHz.

Para cada um dos testes realizados foram ilustrados os gráficos do sinal original, do sinal após filtragem e do sinal filtrado com aplicação do downsample.

5.1 Teste com fator de subamostragem =3

Para o primeiro teste, entendeu-se a utilização de um fator de subamostragem de valor 3.

Na figura 11, pode-se ver o filtro criado. É clara a existência de ripple na banda de rejeição de aproximadamente 60 dB pelo que a escala não permite uma melhor análise.

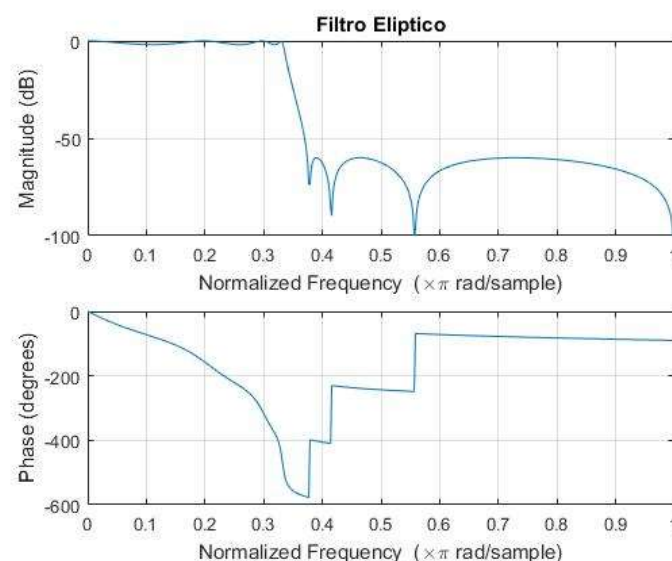


Figura 11- Filtro Criado com subamostragem=3

Na figura 12, é possível observar o sinal original de entrada e o sinal filtrado. Aqui pode-se verificar que, após a aplicação do filtro no sinal, se tem uma melhor percepção do conteúdo que foi filtrado e aquele que permaneceu no sinal. Daqui, verifica-se a correta aplicação do filtro no sinal.

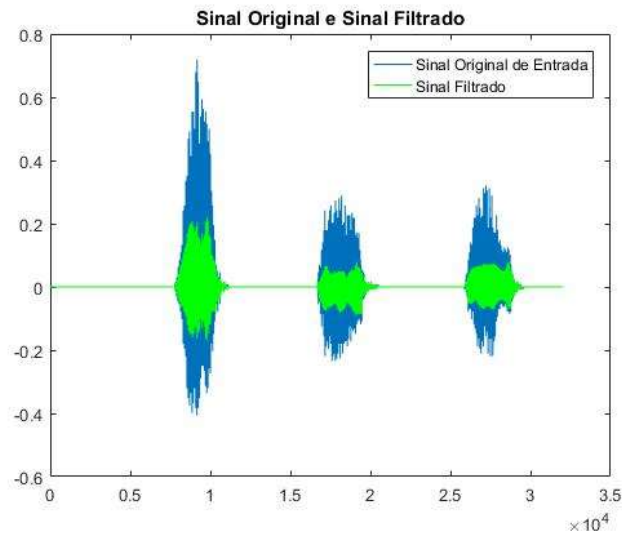


Figura 12- Aplicação do filtro

Na figura 13, tem-se uma melhor percepção do sinal filtrado. Através da escala do eixo dos yys, verifica-se uma redução na amplitude do sinal filtrado significativa.

Sendo que o sinal passa por um processo de “Downsample” em que o fator de subamostragem é de 3, pode-se verificar a ocorrência correta desse fenómeno. Como o sinal contém 3 segundos e é amostrado a 8000 Hz, o número total de amostras do sinal é de 24000.

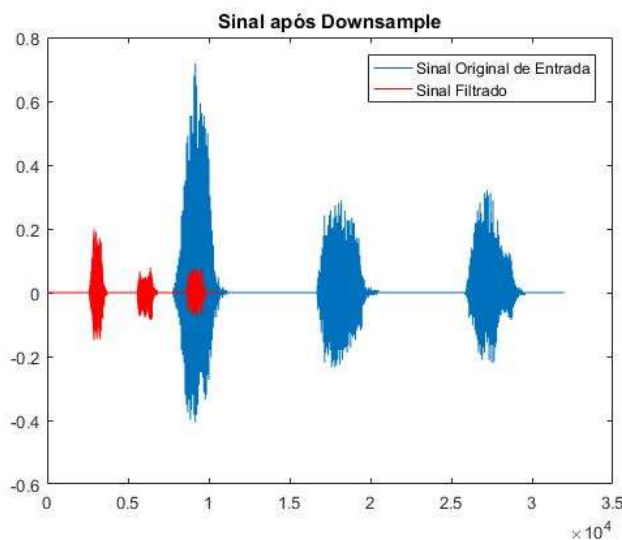


Figura 13- Sinal Filtrado depois de “Downsample”

5.2 Teste com fator de subamostragem =6

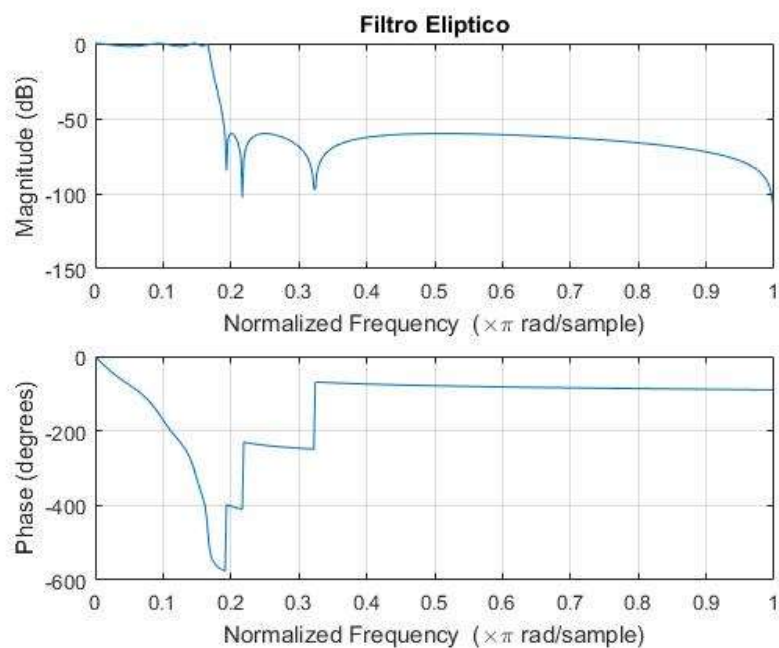


Figura 14- Filtro Criado com subamostragem=6

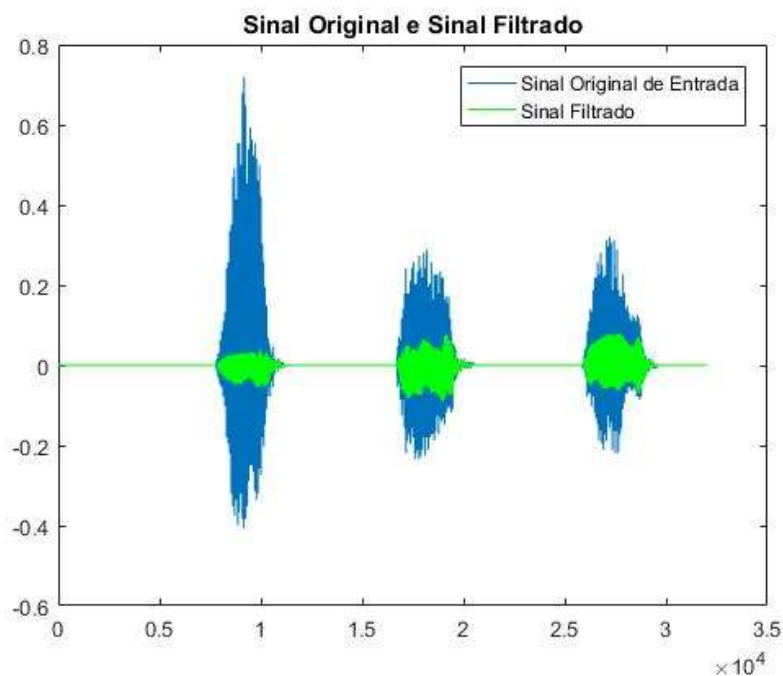


Figura 15- Aplicação do filtro (subamostragem=6)

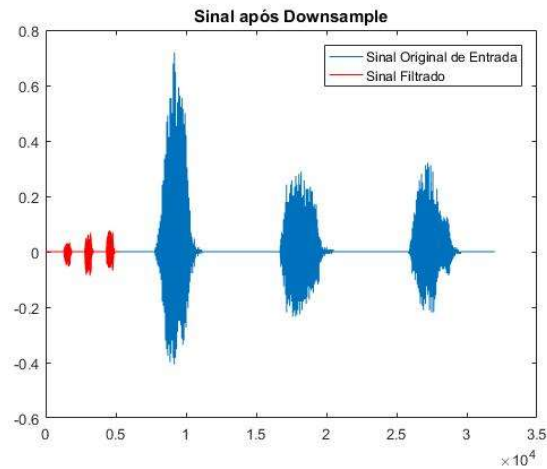


Figura 16- Sinal Filtrado depois de "Downsample" (subamostragem=6)

5.3 Teste com fator de subamostragem =9

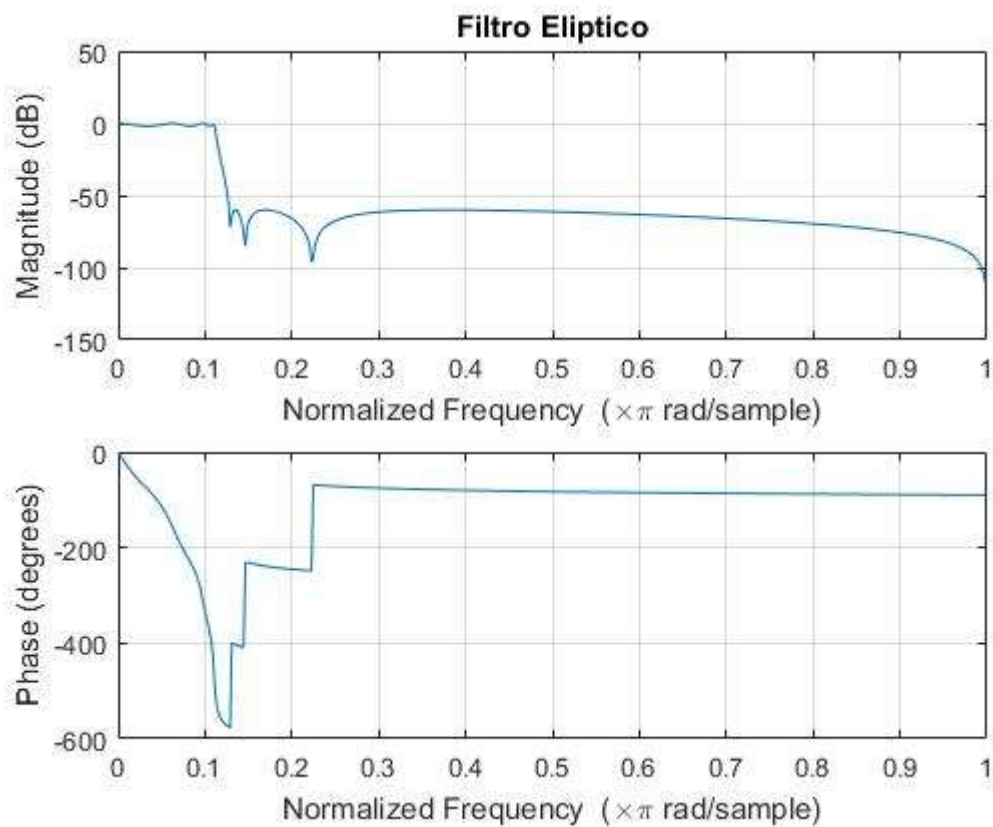


Figura 17- Filtro Criado com subamostragem=9

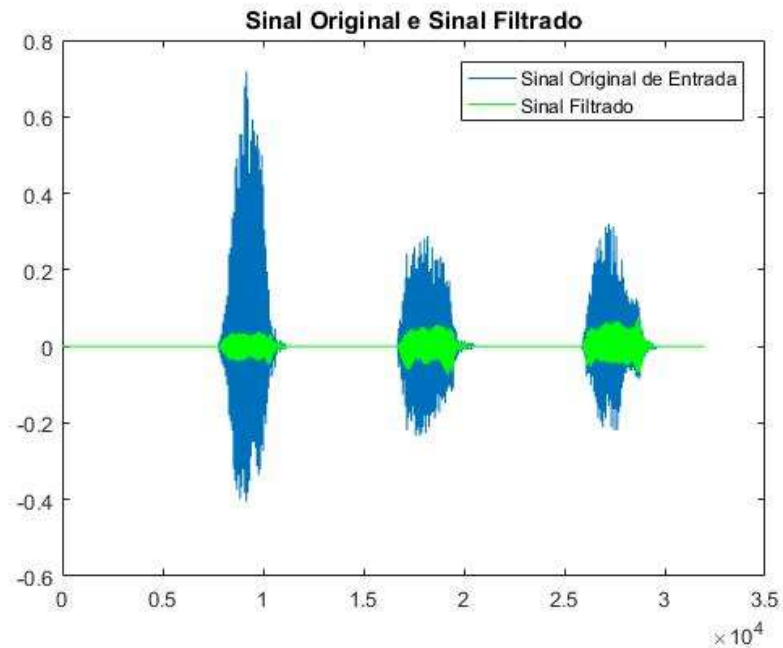


Figura 18- Aplicação do filtro (subamostragem=9)

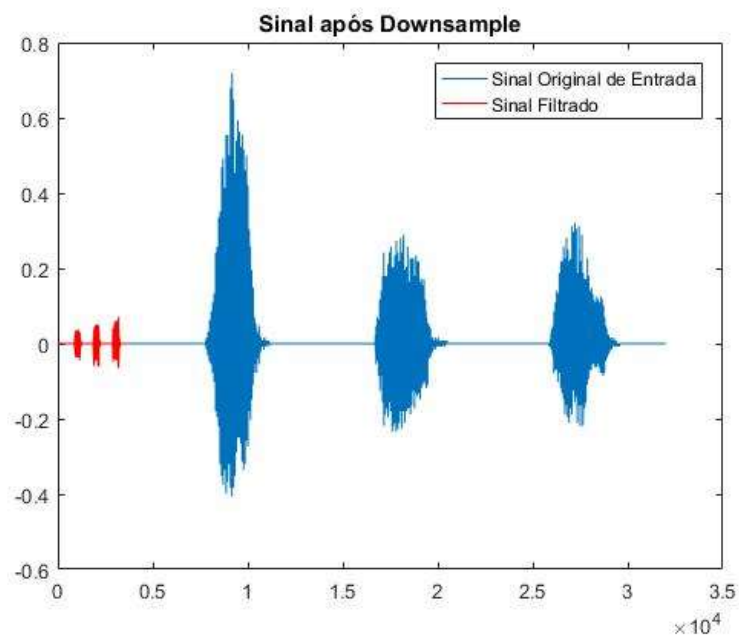


Figura 19- Sinal Filtrado depois de "Downsample" (subamostragem=9)

Com a observação dos gráficos anteriores, podemos observar que com $N=3$ o sinal fica com o seu tamanho reduzido em 3 vezes, com $N=6$ com o seu tamanho reduzido em 6 vezes e com $N=9$ com o seu tamanho reduzido em 9 vezes, tal como é esperado.

6. Conclusão

Com a realização deste trabalho foi possível desenvolver competências no software Matlab, bem como consolidação de conhecimentos relativamente a esta matéria abordados nas aulas da unidade curricular de Processamento Digital de Sinal.

Após os testes, conseguimos observar as diferenças do sinal inicial e do sinal após a decimação.

Assim, após a realização de todos os testes ao algoritmo desenvolvido, pode ser observado de uma melhor forma o seu funcionamento, consoante a gravação efetuada, previamente. Podendo-se, então, retirar as seguintes conclusões associados aos 2 passos que complementam a decimação:

- Quanto ao filtro, pode-se concluir que aquando da gravação do sinal, este é guardado no array com ruído e certos sons que não correspondem àquilo que se pretende gravar. A utilização do filtro neste array (som gravado) permite limpar todo aquele som que não é pertinente guardar como ruídos. Só com a utilização do filtro é possível verificar-se uma melhor clareza do som que foi gravado.
- Quanto ao “Downsampling”, à medida que o aumenta, a frequência de amostragem de o sinal e a frequência de corte do filtro diminuem. Esta diminuição provoca uma compactação do sinal, fazendo com que este ocupe menos memória e que os cálculos efetuados com o mesmo sejam mais “leves”.

Em suma, foi possível verificar com a realização deste trabalho todos os conceitos aprendidos em contexto de sala de aula, sendo que, com a utilização do software Matlab, os conteúdos esclarecidos.



7. Índice de Figuras

Figura 1- Modelo de Decimação	4
Figura 2- Exemplo de Decimação	5
Figura 3- Esquema Filtro Digital	7
Figura 4- Resposta em frequência de um filtro passa baixo elíptico quarta ordem	10
Figura 5- Comparação entre filtros usando o mesmo número de coeficientes	10
Figura 6- Código em Matlab para a criação do Som	12
Figura 7- Código em Matlab para determinação dos parâmetros do Filtro	12
Figura 8- Código em Matlab para criação do Filtro.....	13
Figura 9- Código em Matlab para aplicação e utilização do Filtro	13
Figura 10- Janela com Sinal Original e Filtrado após o Downsample.....	14
Figura 11- Filtro Criado com subamostragem=3.....	15
Figura 12- Aplicação do filtro	16
Figura 13- Sinal Filtrado depois de “Downsample”	16
Figura 14- Filtro Criado com subamostragem=6.....	17
Figura 15- Aplicação do filtro (subamostragem=6).....	17
Figura 16- Sinal Filtrado depois de “Downsample” (subamostragem=6)	18
Figura 17- Filtro Criado com subamostragem=9.....	18
Figura 18- Aplicação do filtro (subamostragem=9).....	19
Figura 19- Sinal Filtrado depois de “Downsample” (subamostragem=9)	19



9. Referências

- Aulas e apontamentos de Processamento digital de sinal ano letivo 2019/2020;
- <http://www.mathworks.com>
- http://cadeiras.iscte.pt/cse/Folhas/Filtros/Filtros.htm#_Toc11854319
- <http://w3.ualg.pt/~sjesus/aulas/pds/node31.html>
- https://pt.wikipedia.org/wiki/Filtro_digital
- <http://www.cear.ufpb.br/juan/wp-content/uploads/2016/08/Introdu%C3%A7%C3%A3o-a-Filtros-Digitais.pdf>
- https://pt.qwe.wiki/wiki/Elliptic_filter