PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO

# Trabalho 4

Sinais e Sistemas - ENG1400


Alunos:
Leo Land Bairos Lomardo - 2020201
Lucca Vieira Rocha - 2011342


Professor: Guilherme Torelly

Rio de Janeiro, RJ
Dezembro, 2023
Período 2023.2

# Conteúdo



# Lista de Figuras

# 1 Introdução

Neste trabalho, abordaremos o som como um sinal de onda sonoras, no tempo contínuo com frequência na faixa de $20Hz$ a $20kHz$, onde esse intervalo, é o intervalo audível para o ser humano. Analisaremos os processos de captura desse sinal, conversão do sinal elétrico para sua forma digital.

Já no formato digital, a música se torna um sinal com valores discretos de intensidade, e sua fidelidade sonora ao ser reproduzida em diferentes meios, está relacionada a sua taxa de amostragem e de bits.

Importante dissecar a estrutura da música, para entender os fatores a serem analizados. A música se propaga pelo meio através de uma onda mecânica, essa onda mecância é caracterizada por várias propriedades, sendo uma delas a frequência.

Em uma música com diversas frequências, cada frequência representa uma nota musical, um som.

# 2 Desenvolvimento

## 2.1 Sistema de Digitalização de Áudio

Em um sistema de digitalização de áudio, para que a captura das características temporais do som sejam capturadas com precisão, a frequência de amostragem, também conhecida como taxa de amostragem, é essencial. Modificar a taxa de amostragem, pode acarretar em mudanças bruscas na qualidade do som digitalizado.

A frequência de amostragem determina quantas amostras por segundo são retiradas de um sinal contínuo. De acordo com o Teorema de Nyquist-Shannon, a frequência de amostragem deve ser pelo menos o dobro da frequência mais alta presente no sinal, afim de evitar a perda de informações.

## 2.2 Teorema de Nyquist-Shannon

O Teorema de Amostragem de Nyquist(Teorema de Nyquist-Shannon), é um princípio fundamental na teoria da comunicação e processamento de sinais. O teorema estabelece as condições necessárias para a amostragem correta de um sinal analógico contínuo, garantindo que possa ser perfeitamente reconstruído a partir de suas amostras. Quando as condições são satisfeitas, podemos transformar um sinal analógico para digital, e digital para analógico, sem que hajam perdas significativas de informações. Tais condições estão resumidas abaixo:

- **Teorema Fundamental:** Um sinal contínuo, onde não possui frequências significativas acima de uma frequência máxima(limite de banda), pode ser completamente reconstruído a partir de suas amostras, desde que a taxa de amostragem seja pelo menos o dobro da frequência máxima presente no sinal.

$$X(f) = 0 \quad \text{para } |f| > \frac{1}{2T}$$

- **Frequência de Nyquist:** A taxa de amostragem mínima($f_s$) necessária (frequência de Nyquist), é igual ao dobro da frequência máxima($f_c$) do sinal. Qualquer taxa de amostragemabaixo dessa frequência resultará em *aliasing*, onde frequências do sinal são erroneamente representadas como frequências mais baixas.

$$f_s > 2f_c$$

## 2.3 Alteração da Frequência

Considerando um arquivo inicial, amostrado em uma frequência $F$, caso modifiquemos ele para que seja amostrado nas frequência $2F$ e $8F$, acontecerão diversas mudanças, algumas podendo interferir diretamente na qualidade do som e no tamanho do arquivo. Além da alteração da taxa de amostragem, também limitamos o intervalo de análise do sinal em função do tempo para $[0, 40]$, com o objetivo de tornar a análise dos gráficos mais detalhada. Antes de analisar o resultado das alterações, devemos analisar o arquivo inicial:

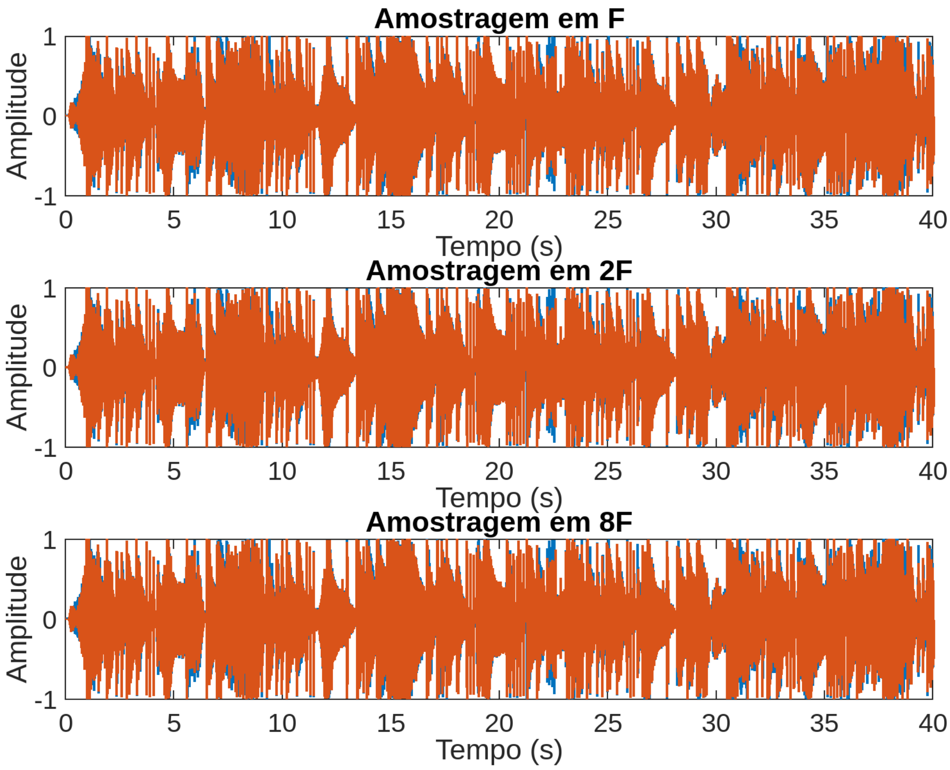


Figura 1: Comparação entre $F$'$s$ maiores

Analisando o primeiro caso, onde mudaremos a frequência para $2F$, podemos afirmar que o arquivo poderá sofrer mudanças significativas, caso a frequência de amostragem esteja abaixo da taxa de Nyquist. Tais mudanças podem acarretar em um arquivo com tamanho maior, visto que terão mais amostras por segundo, ocupando mais espaço de armazenamento.

Pelo gráfico é visível o aumento das amplitudes que tornam os gráficos com um $F$ maior com uma aparência mais "densa"o que implica em um arquivo maior. Esse é o motivo de que arquivos com $F$ maior são mais pesados pois em teoria possuem uma resolução mais detalhada do sinal analógico.

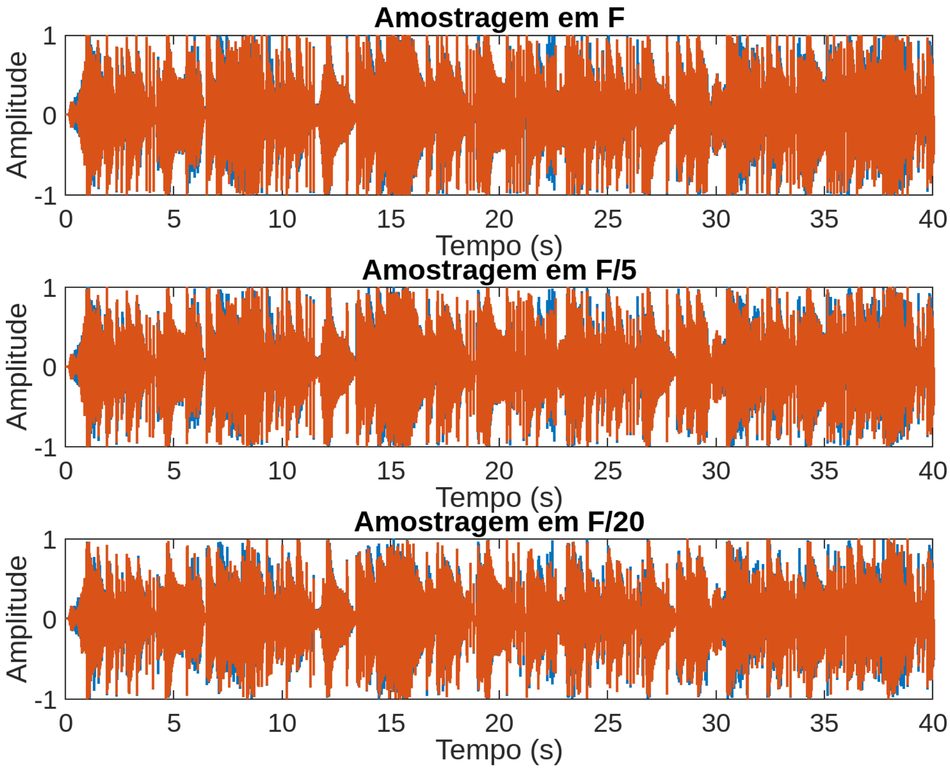


Figura 2: Comparação entre $F$'s menores

Já neste gráfico acontece justamente o contrário, notamos uma diminuição nas amplitudes, resultando em gráficos menos "densos"quando a frequência é mantida em F. Isso implica em arquivos menores, pois possuem menos amostras, teoricamente resultando em uma resolução menos detalhada do sinal analógico.

Essa é a razão pela qual arquivos com F menor são mais leves, já que, em teoria, apresentam uma resolução menos detalhada do sinal, refletindo-se em gráficos menos densos e, consequentemente, em um tamanho de arquivo menor.

Segue abaixo os gráficos e as informações dos arquivos originais e dos modificados para as respectivas músicas:

| *DontSopMe* | **Original** | **2F** | **8F** | **F/5** | **5/20** |
|---|---|---|---|---|---|
| **Tamanho do Arquivo (Mb)** | 26,5468 | 53,0934 | 212,3735 | 5,3094 | 1,3274 |
| **Taxa de Amostragem (Hz)** | 44100 | 88200 | 352800 | 8820 | 2205 |

## 2.4 Código MATLAB

```
limite_tempo = [0, 40];
filename = '/MATLAB Drive/Trabalho4/dont-stop-me-abstract-
    future-bass-162753.mp3';
[x, Fs] = audioread(filename);
```

No trecho do código acima, importamos o arquivo da música no formato `.wav` e limitamos o intervalo de tempo a ser analisado. Tal limitação foi implementada, com o objetivo de tornar a visualização dos gráficos menos poluída.

```
t_F = (0:length(x)-1) / Fs;
originalInfo = dir(filename);
disp('Informacoes sobre o arquivo original:');
disp(['Tamanho do arquivo original: ' num2str(originalInfo.
    bytes/1e6) ' MB']);
disp(['Taxa de amostragem original: ' num2str(Fs) ' Hz']);
```

Aqui utilizamos a função *dir(arquivo.wav)* para coletar informações pertinentes, como a *Taxa de Amostragem* e o *Tamanho do Arquivo*, para poder comparar os efeitos das transformações.

```
Fs_2F = 2 * Fs;
x_2F = resample(x, Fs_2F, Fs, 0, 0);
outputFilename_2F = '/MATLAB Drive/Trabalho4/
    DontStopMe_amostragem_2F.wav';
audiowrite(outputFilename_2F, x_2F, Fs_2F);
info_2F = dir(outputFilename_2F);
disp('Informacoes sobre o arquivo amostrado em 2F:');
disp(['Tamanho do arquivo amostrado em 2F: ' num2str(info_2F.
    bytes/1e6) ' MB']);
disp(['Taxa de amostragem amostrada em 2F: ' num2str(Fs_2F) '
    Hz']);
```

Para realizar alterações na taxa de amostragem da faixa de áudio, utilizamos a função `resample` do *MATLAB*. Importante destacar, que a função `resample`, vem automaticamente com um filtro *anti-aliasing*. Esse filtro, deve ser removido, para que possamos perceber os efeitos da transformação com clareza.

Para remover o filtro, basta zerar os dois últimos parâmetros da função.

```
Fs_8F = 8 * Fs;
x_8F = resample(x, Fs_8F, Fs, 0, 0);
outputFilename_8F = '/MATLAB Drive/Trabalho4/DontStopMe_amostragem_8F.wav';
audiowrite(outputFilename_8F, x_8F, Fs_8F);
info_8F = dir(outputFilename_8F);
disp('Informacoes sobre o arquivo amostrado em 8F:');
disp(['Tamanho do arquivo amostrado em 8F: ' num2str(info_8F.bytes/1e6) ' MB']);
disp(['Taxa de amostragem amostrada em 8F: ' num2str(Fs_8F) ' Hz']);
Fs_1_5F =  Fs/5;
x_1_5F = resample(x, Fs_1_5F, Fs, 0, 0);
outputFilename_1_5F = '/MATLAB Drive/Trabalho4/DontStopMe_amostragem_1_5F.wav';
audiowrite(outputFilename_1_5F, x_1_5F, Fs_1_5F);
info_1_5F = dir(outputFilename_1_5F);
disp('Informacoes sobre o arquivo amostrado em F/5:');
disp(['Tamanho do arquivo amostrado em F/5: ' num2str(info_1_5F.bytes/1e6) ' MB']);
disp(['Taxa de amostragem amostrada em F/5: ' num2str(Fs_1_5F) ' Hz']);
Fs_1_20F = Fs/20;
x_1_20F = resample(x, Fs_1_20F, Fs, 0, 0);
outputFilename_1_20F = '/MATLAB Drive/Trabalho4/DontStopMe_amostragem_1_20F.wav';
audiowrite(outputFilename_1_20F, x_1_20F, Fs_1_20F);
info_1_20F = dir(outputFilename_1_20F);
disp('Informacoes sobre o arquivo amostrado em F/20:');
disp(['Tamanho do arquivo amostrado em F/20: ' num2str(info_1_20F.bytes/1e6) ' MB']);
disp(['Taxa de amostragem amostrada em F/20: ' num2str(Fs_1_20F) ' Hz']);
```

```
figure;
subplot(3,1,1);
plot(t_F, x, 'LineWidth', 1);
title('Amostragem em F');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
subplot(3,1,2);
plot((0:length(x_2F)-1) / Fs_2F, x_2F, 'LineWidth', 1);
title('Amostragem em 2F');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
subplot(3,1,3);
plot((0:length(x_8F)-1) /Fs_8F, x_8F, 'LineWidth', 1);
title('Amostragem em 8F');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
```

Optamos por dividir os gráficos do trabalho em duas partes, onde na primeira comparamos o áudio original, com os áudios com a taxa de amostragem $2F$ e $8F$.

```
figure;
subplot(3,1,1);
plot(t_F, x, 'LineWidth', 1);
title('Amostragem em F');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
subplot(3,1,2);
plot((0:length(x_1_5F)-1) / Fs_1_5F, x_1_5F, 'LineWidth', 1);
title('Amostragem em F/5');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
subplot(3,1,3);
plot((0:length(x_1_20F)-1)/Fs_1_20F, x_1_20F, 'LineWidth',1);
title('Amostragem em F/20');
xlim(limite_tempo);
xlabel('Tempo (s)');
ylabel('Amplitude');
```

Nesta parte final do código, realizamos a construção dos gráficos referentes as taxas de amostragem $F$, $\frac{F}{5}$ e $\frac{F}{20}$.

# 3 Conclusão

Neste projeto aprendemos como de fato é feita a conversão de um som analógico em um sinal digital e além disso como a compressão desse sinal digital para economizar espaço. Além disso os gráficos criados no *MATLAB* nos auxiliaram a visualizar este processo tornando o mais claro e auxiliando na compreensão da matéria.

Ao longo do projeto o Teorema de Nyquist-Shannon desempenha um papel fundamental na conversão de sinais sonoros analógicos em representações digitais, sendo essencial para a preservação da qualidade do áudio. Ao entendermos a teoria por trás desse teorema e sua aplicação prática, percebemos como ele é crucial no processo de compressão de arquivos de áudio. A compressão, por sua vez, torna-se uma ferramenta valiosa para otimizar o armazenamento e a transmissão de dados, resultando em arquivos mais compactos sem sacrificar significativamente a qualidade perceptível do som.

As visualizações gráficas geradas pelo *MATLAB* oferecem uma perspectiva visual clara desse processo, contribuindo para uma compreensão mais profunda e eficaz dos princípios subjacentes à digitalização e à compressão de sinais de áudio. Dessa forma, o Teorema de Nyquist-Shannon não apenas viabiliza a representação digital do som, mas também influencia diretamente as estratégias de compressão que são essenciais na gestão eficiente de dados de áudio.

# 4 Referências


[1] *Documentação LaTeX*. 2023. URL: https://www.latex-project.org/help/documentation/.

[2] *Documentação MATLAB*. 2023. URL: https://www.mathworks.com/help/?s_tid=mlh_sn_help.

[3] Alan V. Oppenheim, Alan S. Willsky e S. Nawab Hamid. *Signals and Systems*. 2ª ed. Upper Saddle River, NJ: Prentice Hall, 1997. ISBN: 0-13–814757–4.

[4] *Resampling*. 2023. URL: https://www.mathworks.com/help/signal/ug/resampling.html.


Também utilizamos a tecnologia *ChatGPT* para nos auxiliar no desenvolvimento do relatório e do código, sanando dúvidas exclusivamente relacionadas a sintaxe da linguagem do LaTeX.