

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
ENGENHARIA DE COMPUTAÇÃO

Wagner Spinato Chittó

1º TRABALHO PRÁTICO
Espectro de Sinais

Santa Maria, RS
11/08/2025

Objetivo:

Obtenção do espectro de sinal usando o algoritmo da Transformada Rápida de Fourier (FFT – Fast Fourier Transform).

Fundamentação:

Arquivo de simulação “aula1.m”:

```
clear all;close all;clc;

% Exemplo de analise de sinais amostrados
td=1/512; % intervalo entre os pontos do sinal "analogico"
t=td:td:1; % intervalo de 1 segundo
%xsig=sin(2*pi*t)+sin(2*3*pi*t); % seno de 1Hz + 3Hz;
xsig=2*sin(2*30*pi*t)+2*sin(2*20*pi*t); % seno de 30Hz;
%xsig = square(2*pi*3*t); % onda quadrada
%xsig=rectpuls(t*3); % pulso quadrado
Lsig=length(xsig);

figure(1);
subplot(211); sfig1a=plot(t,xsig,'k');
set(sfig1a,'LineWidth',2);
xlabel('tempo,segundos');
%axis([0 1 -2 2]);
title('sinal {\it g} ({\it t})');

% calcula a transformada de Fourier
Lfft=2^ceil(log2(Lsig));
Fmax=1/(2*td); % Nyquist!
Faxis=linspace(-Fmax,Fmax,Lfft+1);
Xsig=fftshift(fft(xsig,Lfft)/Lfft);

subplot(212); sfig1b=stem(Faxis(1:Lfft),abs(Xsig));
set(sfig1b,'LineWidth',1);
xlabel('frequencia (Hz)');
axis([-50 50 0 2]);
title('Espectro de {\it g} ({\it t})');
```

Procedimento:

Do arquivo .m acima, foi montado o seguinte código em python para obter os 3 primeiros gráficos:

```
import matplotlib.pyplot as plt
from scipy import signal
import numpy as np

td=1/512
t=np.linspace(0, 1, 512)

#a) SINAL SENOIDAL:
xsig=np.sin(2*np.pi*1*t)+np.sin(2*np.pi*3*t)
#b) PULSO QUADRADO:
xsig=np.square(2*np.pi*3*t)
#c) ONDA QUADRADA:
xsig=signal.square(t*3)

Lsig=len(xsig)

Lfft=pow(2, np.ceil(np.log2(Lsig)))
Fmax=1/(2*td)
Faxis=np.linspace(-Fmax,Fmax,int(Lfft+1))
Xsig=np.fft.fftshift(np.fft.fft(xsig,Lfft)/Lfft)

plt.figure(figsize=(16,12))

#Plot Sinal
plt.subplot(2, 1, 1)
plt.plot(t, xsig)
plt.title('Sinal')

#Plot Espectro
plt.subplot(2, 1, 2)
plt.vlines(Faxis[0:int(Lfft)], abs(Xsig), np.zeros(len(Xsig)))
plt.title('Espectro')

plt.show()
```

Para o sinal de áudio obtido pela gravação do microfone (função sound do Octave), foi feito o seguinte código no Octave:

```
clear all;close all;clc;

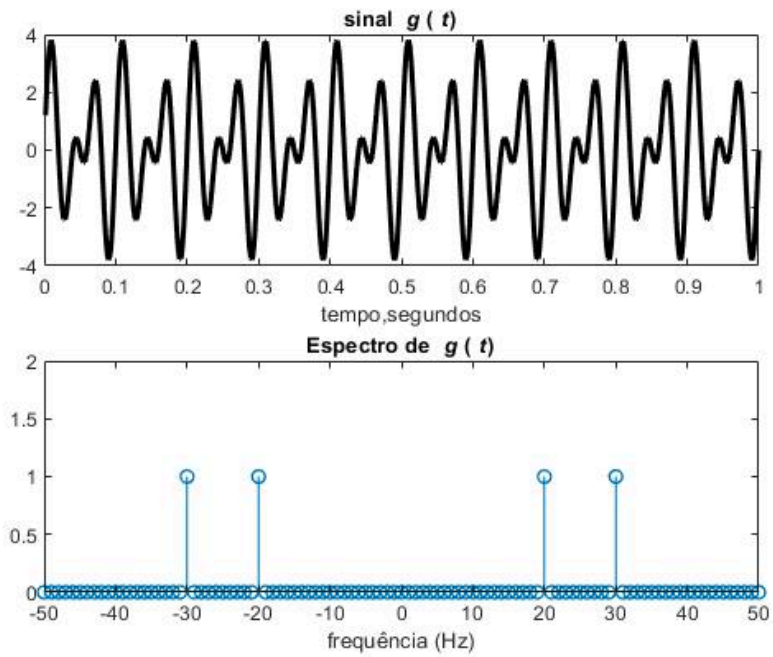
[y,fs] = audioread('<AUDIOFILE>'); N = length(y);
Y = fft(y,N);
F = ((0:1/N:1-1/N)*fs);
w = 2*pi*F; magnitudeY = abs(Y);
phaseY = unwrap(angle(Y)); figure (1);
subplot(2,1,1);
plot (F, phaseY); grid on;
set(gca, 'FontName', 'Times New Roman', 'FontSize', 14);
xlabel('Frequency(Hz)');
ylabel('Phase angle(rad)'); title('Sinal (Phase spectrum)');

subplot(2,1,2); plot(F, magnitudeY); grid on;
set(gca, 'FontName', 'Times New Roman', 'FontSize', 14); xlabel('Frequency
(Hz)');
ylabel('Magnitude (dB)'); title('Magnitude spectrum');
```

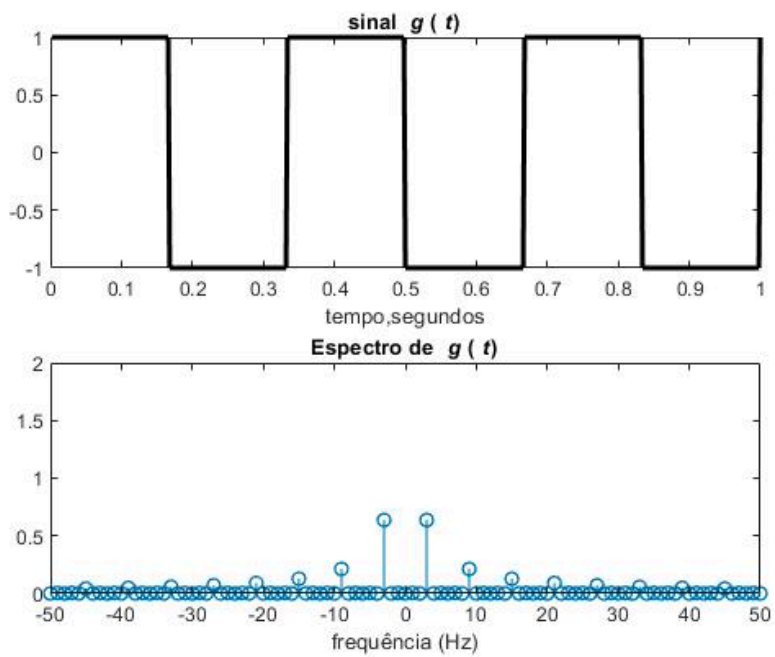
Sendo <AUDIOFILE> dois arquivos baixados no site <https://www.pacdv.com/>

Resultados:

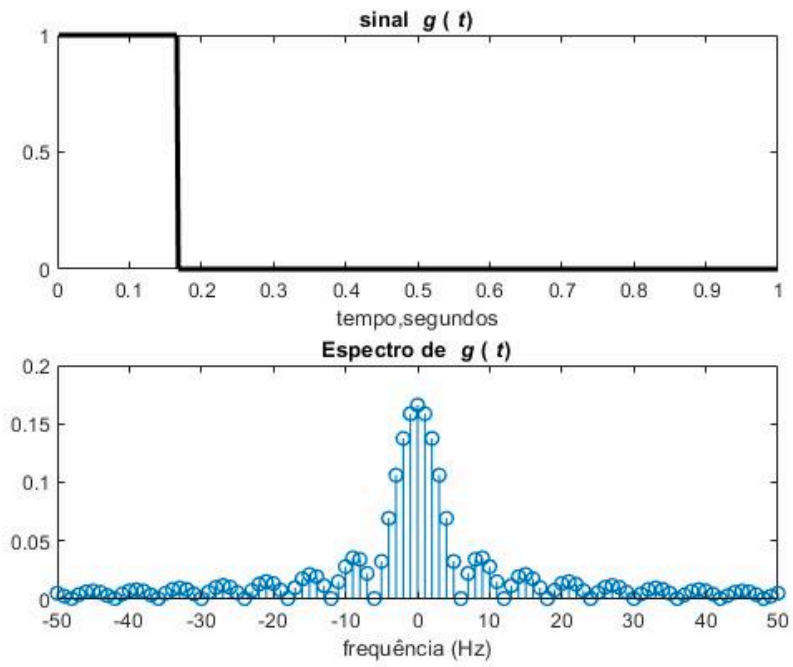
1.1 - Sinal Senoidal



1.2 - Onda Quadrada

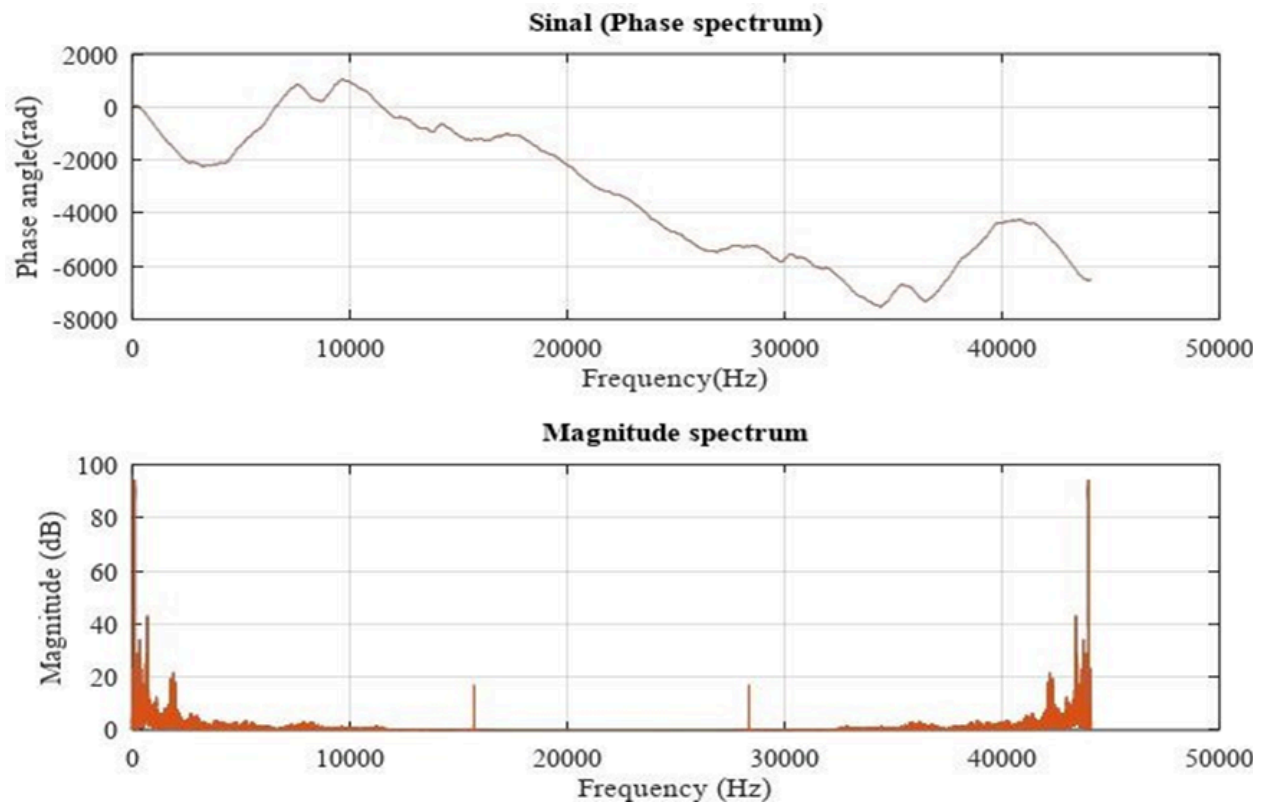


1.3 - Pulso Quadrado

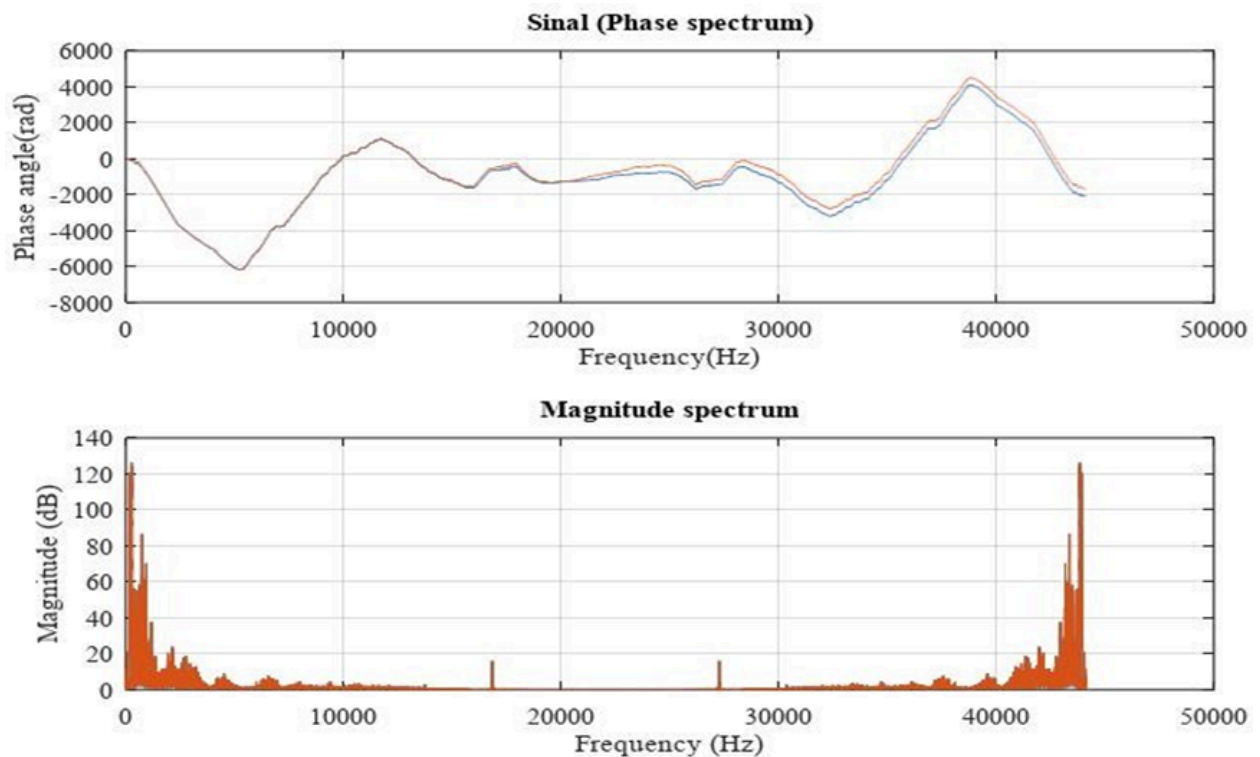


2. Sinal de áudio obtido pela gravação do microfone (função record do Octave)

2.1. Gráficos para voz masculina na pronúncia de “ænd”:



2.1. Gráficos para voz feminina na pronúncia de “ænd”:



Conclusão:

Os resultados obtidos são coerentes com os esperados.

Bibliografia:

- Algoritmo aula1.m: https://ead06.proj.ufsm.br/pluginfile.php/5461130/mod_resource/content/2/aula1.m
- Gravação Masculina: <https://www.pacdv.com/sounds/voices/and-2.wav>
- Gravação Feminina: <https://www.pacdv.com/sounds/voices/and-1.wav>
- <https://www.mathworks.com>