**FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO**

**TEORIA DO TRATAMENTO DE SINAL**

**M.EMG**

# Trabalho Final - Versão Provisória

*Estudantes:*
João SÁ PEREIRA

*Docentes:*
Prof. Jorge CARVALHO
Prof. Teresa LAJINHA

13 de Abril de 2024

# Conteúdo

# Lista de Figuras

# Código

# 1. Fourier Theory Made Easy (?)

## 1.1. Uma onda sinusoidal

Começaremos por reproduzir uma onda sinusoidal. A sua forma mais básica, como função do tempo, é dada pela seguinte expressão:

$$y(t) = A \cdot \sin(2\pi ft + \phi) \tag{1.1}$$

Para o primeiro exemplo, iremos representar em MATLAB uma onda sinusoidal com amplitude 5 e frequência de 4 Hz. Para isso, é necessário definir o eixo do tempo. Neste caso, o eixo do tempo começa em zero e acaba em 1 (segundos), com incrementos de 0.01. O código que fornece a imagem é o seguinte:

**Código 1.1:** *Slide 2.*

```
dt = 0.01;
t = 0:dt:1;
F = 4;
A = 5;
y = A * sin(2 * pi * F * t);
plot(t,y);
xlabel('Segundos');
```

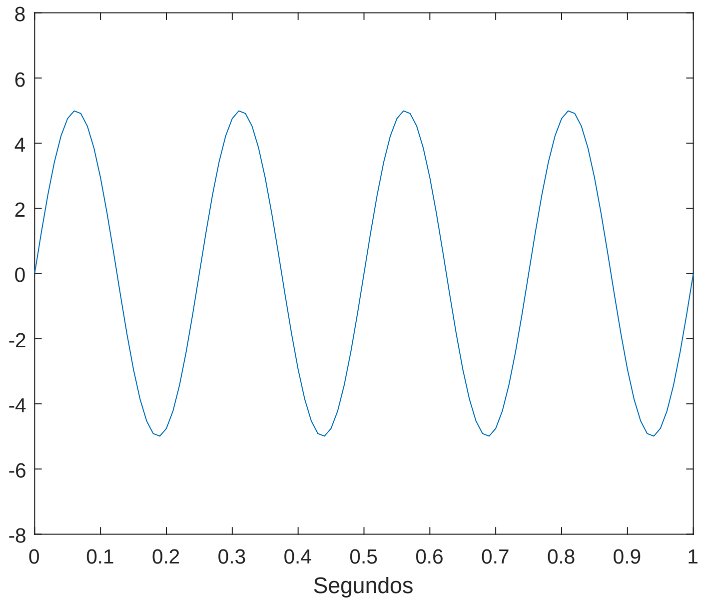


**Figura 1.1:** *Onda sinusoidal.*

Se definirmos a taxa de amostragem (*sampling rate*) como sendo igual a 256 e a duração da amostragem (*sampling duration*) como sendo igual a 1, podemos definir a variável dt como sendo o quociente entre a *sampling duration* e o *sampling rate*. E se representarmos a figura com pontos, podemos verificar que por cada segundo, temos 256 pontos.

**Código 1.2:** *Slide 3.*

```
sampling_rate = 256;
sampling_duration = 1;
dt = sampling_duration/sampling_rate;
t = 0:dt:1;
F = 4;
A = 5;
y = A * sin(2 * pi * F * t);
plot(t, y, 'o', 'MarkerSize',4);
xlabel('Segundos')
```

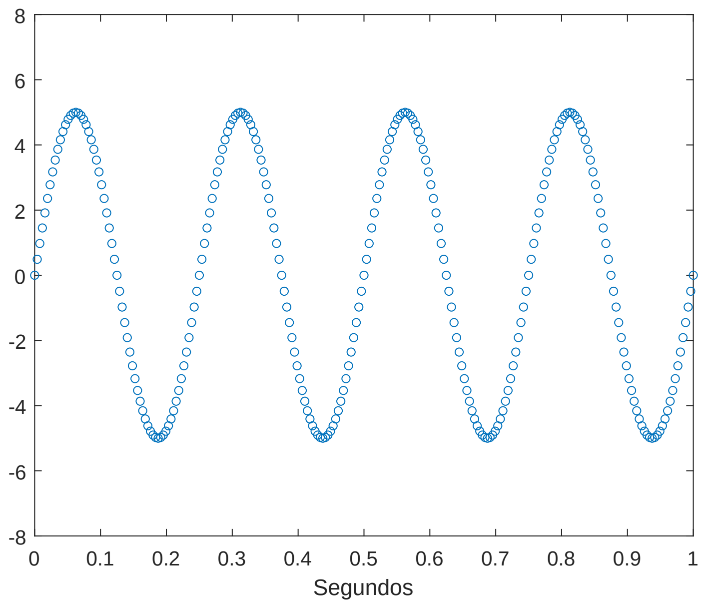


**Figura 1.2:** *Taxa de amostragem.*

Agora uma função sinusoidal com frequência de 8 Hz e com uma determinada taxa de amostragem. Quando se altera o valor do passo de amostragem, para 8.5 Hz, obtêm-se um sinal com frequência de 0.5 Hz. Este exercicio, recriado em MATLAB no código da página seguinte, mostra a importância da escolha do passo de amostragem.

**Código 1.3:** *Slide 4.*

```
sampling_rate = 8.5;
sampling_duration = 1;
dt1 = 1/256;
dt2 = sampling_duration/sampling_rate;
% Vetor tempo e função seno 1
t1 = 0:dt1:2;
y1 = sin(2 * pi * 8 * t1);
% Vetor tempor e função seno 2
t2 = 0:dt2:2;
y2 = sin(2 * pi * 8 * t2);
% Plots
hold on
plot(t1,y1)
plot(t2,y2, 'o', 'MarkerFaceColor', 'red', 'MarkerSize', 4)
legend('Sinal', 'Sinal subamostrado')
xlabel('Segundos')
hold off
axis([0 2 -2 2])
```

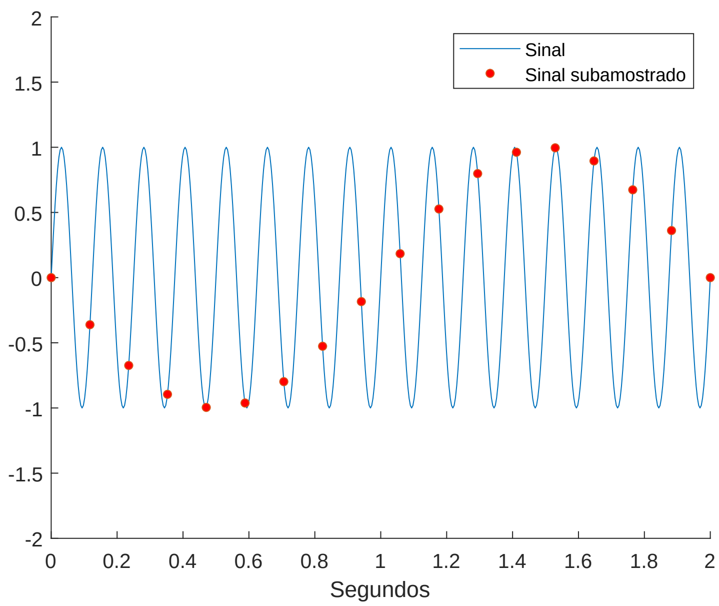


**Figura 1.3:** *Sinal subamostrado.*

## 1.2. Transformadas de Fourier Famosas

Uma transformada pega numa função (ou sinal) e transforma-a numa outra função (ou sinal). A transformada discreta de Fourier é dada pela seguinte expressão:

$$H(f) = \int_{-\infty}^{+\infty} h(t) \cdot e^{2\pi i f t} \cdot dt \tag{1.2}$$

A transformada rápida de Fourier (do inglês: *Fast Fourier Transform*, abreviado FFT), é um algoritmo que calcula a transformada discreta de Fourier. A análise de Fourier converte um sinal do domínio original (tempo) para uma representação no domínio das frequências e vice-versa.

### 1.2.1. Função Seno $\rightarrow$ Função Delta

A transformada de Fourier de uma função sinusoidal é uma função delta de Dirac, pelo menos em parte. Quando aplicamos a transformada de Fourier no MATLAB (função *fft*) a uma função sinusoidal, o resultado é uma combinação de deltas de Dirac localizados nas frequências da função sinusoidal. A transformada de Fourier da função é zero em todas as frequências exceto na frequência da função seno, onde há um Dirac localizado.

**Código 1.4:** *Slide 12.*

```
% Criar os incrementos dt
sampling_duration = 1;
sampling_rate = 250;
dt = sampling_duration/sampling_rate;

% Vetor tempo e função Seno
t = 0:dt:2;
A = 1;
F = 8;
y = sin(2 * pi * F * t);
figure();
plot(t,y);
title('Função Seno');
axis([0 2 -2 2]);

% Função Delta
ft_y = fft(y);
df = (1/dt)/length(ft_y);
freq = (0:length(ft_y)-1)*df;
figure();
plot(freq, abs(ft_y));
title('Função Delta');
axis([0 125 0 300]);
```

É possível verificar pelos gráficos produzidos pelo código anterior, que o Dirac ocorre na frequência $7.98403 \approx 8$, que é a frequência da função sinusoidal.

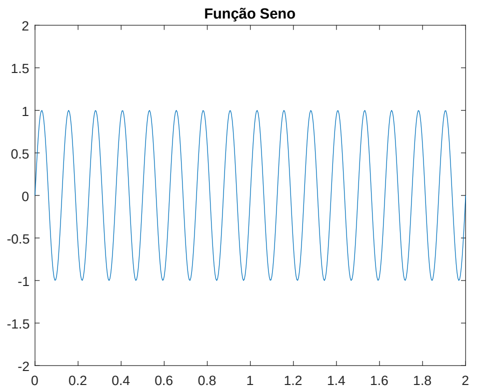


**Figura 1.4:** *Função Seno.*

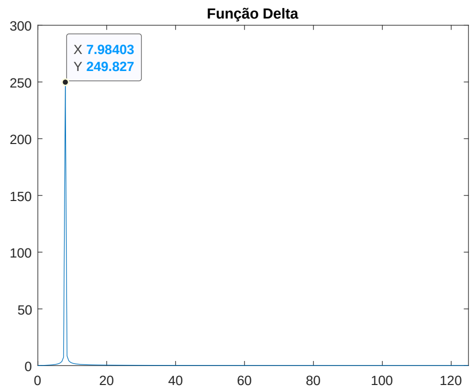


**Figura 1.5:** *Função Delta.*

### 1.2.2. Função Gaussiana

A transformada de Fourier de uma função gaussiana é também uma função gaussiana mas com parâmetros diferentes.

**Código 1.5:** *Slide 13.*

```
% Parâmetros e função Gaussiana
media = 25;
std = 1;
A = 0.4;
dt=0.1;
t=0:dt:50;
s = A * exp(-0.5 * ((t - media)/std).^2);
figure();
plot(t,s);
title('Função Gaussiana');
axis([0 50 0 0.5]);

% Transformada de Fourier
S = fftshift(abs(0.5*fft(s))); % Multiplicação por 0.5 para igualar ao apresentado no ppt
f = 0:0.5:250;
figure();
plot(f,S, Color='#A2142F');
title('Função Gaussiana - TF')
```

Neste caso, a amplitude da transformada de Fourier é 5 e a média é 125. As figuras da página seguinte mostram a resposta do código apresentado.

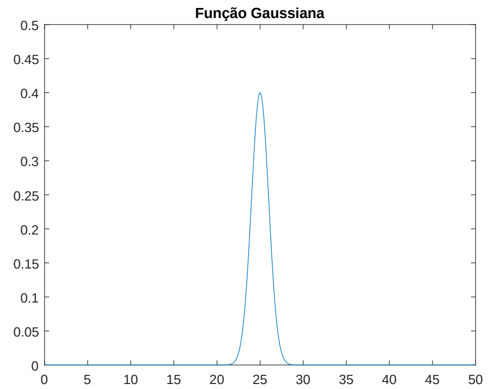


**Figura 1.6:** *Função Gaussiana.*

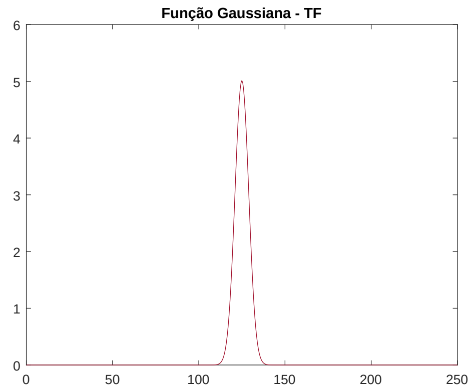


**Figura 1.7:** *Função Gaussiana - TF.*

### 1.2.3. Função Seno Cardinal $\rightarrow$ *Square Wave*

**Código 1.6:** *Slides 14 e 15.*

```
% Parâmetros do sinal Sinc
F = 1000;
T = 1;
f0 = 50;
t = -T:1/F:T;   % Vetor de tempo
s = sinc(f0 * t); % Sinal Sinc

S = abs(0.25*fftshift(fft(s))); % Calcula a TF do sinal
N = length(s);
f = (-N/2 : N/2-1) * (F/N); % Vetor de frequências correspondentes

w = hann(length(s))'; % Função de Hann
sw = s .* w; % Sinal Sinc 'filtro'
Sw = 0.25*fftshift(fft(sw));
f = (-N/2 : N/2-1) * (F/N); % Frequências correspondentes

figure();
plot(t, s);
title('Sinal Sinc filtrado');
ylim([-0.5 1.5])
figure();
plot(f, abs(Sw));
title('Espectro de Frequência (Sinal filtrado)');
xlim([-125 125]);
figure();
plot(f, abs(S));
title('Square wave');
xlim([-125 125]);
```

A transformada de Fourier da função seno cardinal (sinc) resulta numa *square wave* (pedestal).

De forma intuitiva, a função seno cardinal tem oscilações até ao infinito e quando a sua transformada de Fourier é calculada, envolve a soma de componentes sinusoidais infinitas, o que resulta numa onda quadrada.

Essencialmente, a transformada de Fourier da função seno cardinal capta a propriedade fundamental desta função, nomeadamente o seu comportamento oscilatório, que no eixo das frequências, resulta numa *square wave*.

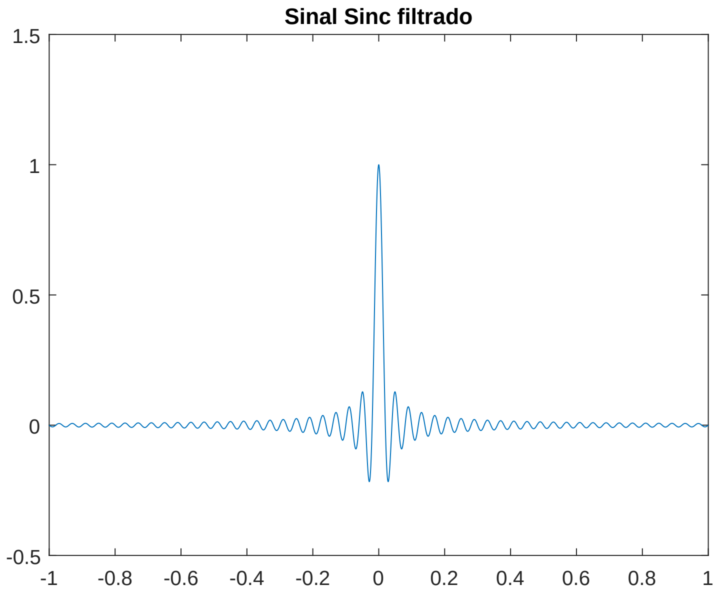


**Figura 1.8:** *Função Sinc.*

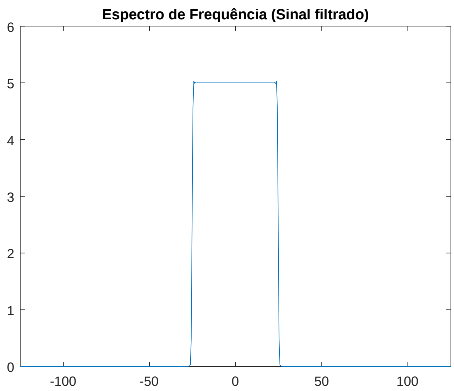


**Figura 1.9:** *Espectro de Frequências.*

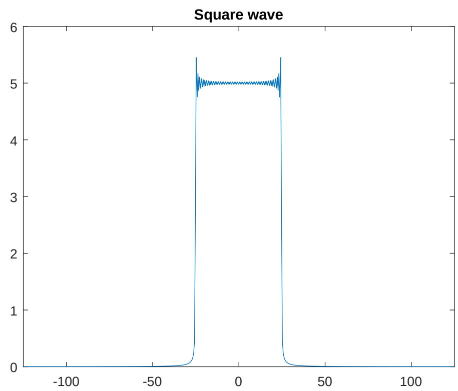


**Figura 1.10:** *Square Wave.*

### 1.2.4. Função Exponencial $\longrightarrow$ Lorentziana

**Código 1.7:** *Slide 16.*

```
dt = 0.1;
t = 0:dt:10;
% Função Exponencial
s = exp(-t);
S = fft(s);
Sr = real(S);
Si = imag(S);
Sabs = abs(S);

figure();
plot(t,s); title('Sinal Exponencial');
figure();
plot(fftshift(Sabs)); title('Transformada de Fourier Lorentzian')
xlim([0, length(Sabs)]);
```

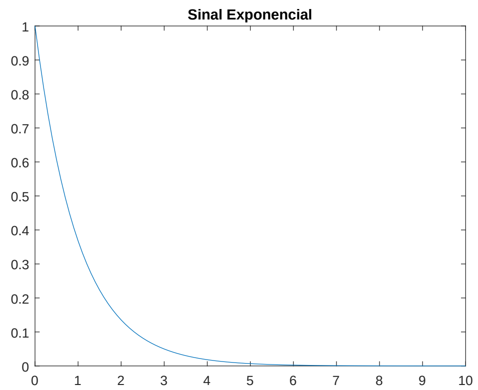


**Figura 1.11:** *Função Exponencial.*

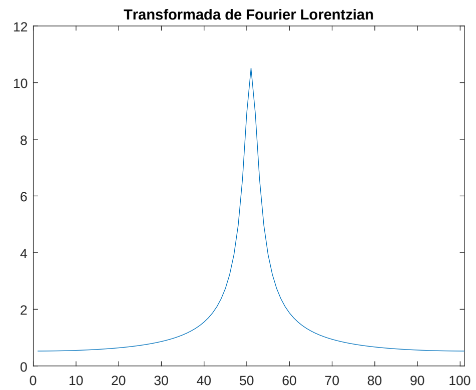


**Figura 1.12:** *Lorentziana.*

## 1.3. FFT ou FID