INSTITUTO FEDERAL
SANTA CATARINA
Campus São José

Relatório 1

Aluno: Guilherme Medeiros

Disciplina: Sistemas de Comunicação I

Professor: Mario Noronha

São José, 23 de Agosto de 2019

Seção I

Aqui serão abordados geração sinais, criação de filtros reais simulados de magnitudes e tipos distintos, sinais aleatórios, potência de um sinal e sua densidade espectral de potência e análise de sinais no domínio do tempo e frequência.

Seção II

Características de um sinal Senoidal:

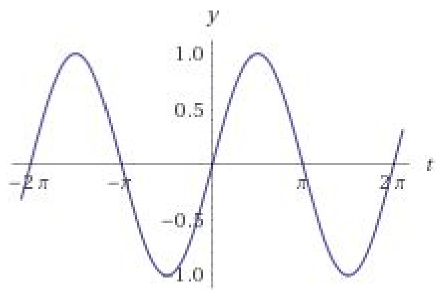
Descrição matemática periódica e contínua de uma oscilação com frequência(s) constante(s). Funções *Sen(x)* e *Cos(x)* descrevem as senóides mais simples que existem, com apenas uma frequência de oscilação e um comprimento de onda com valor de 2π.

Senóides mais complexas podem ser criadas com a soma de duas ou mais funções senoidais em frequências diferentes, dando origem a *n* oscilações na imagem da função final, sendo n o número de frequências diferentes que foram somadas ao sinal.

Aqui, serão tratadas funções senoidais que possuem como domínio o tempo, ou seja, sinais que tem seu valor (imagem) oscilando entre um valor A e -A, este valor é conhecido como amplitude.

Sinais deste tipo possuem então, uma amplitude A, uma frequência f e uma fase θ, que diz a origem do valor de uma senóide e realiza uma translação no eixo do tempo na função. A expressão matemática que representa este sinal pode ser visto na equação 1 e o gráfico realizado por essa oscilação pode ser visto na figura 1.

$$x(t) = A.sin(f.t + \theta) \qquad (1)$$



(Figura 1 - sin(t))

Onde A.cos(f.t) = A.sin(f.t + 90).

Senos e cossenos podem ser escritos em termos de funções exponenciais. Esta relação é denominada *relação de Euler* e pode ser vista na equação 2.

$$e^{it} = cos(t) + i.sin(t) \qquad (2)$$

Outra forma de observar a relação vem do desenvolvimento da relação de Euler para o Cosseno e o Seno, que podem ser verificadas na equação 3 e 4, respectivamente.

$$cos(t) = \frac{e^{it} + e^{-it}}{2} \quad (3)$$

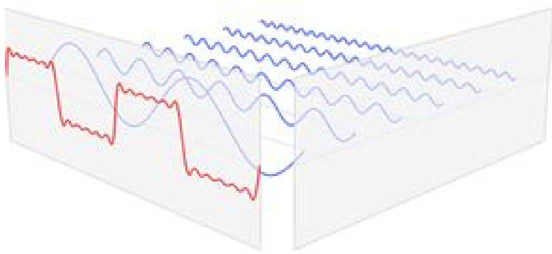
$$sen(t) = \frac{e^{it} - e^{-it}}{2i} \quad (4)$$

A transformada de Fourier:

Em análise de sinais e sistemas lineares, é de grande interesse conhecer as frequências que compõem um sinal periódico. Foi notado por Jean-Baptiste Joseph Fourier no século XIX que um sinal periódico pode ser descrito como a soma das frequências que o compõe. Considerando que sinais senoidais possuem apenas uma frequência de oscilação específica e que sinais senoidais podem ser escritos como a soma de exponenciais complexas (equação 3 e 4), conclui-se que sinais periódicos podem ser descritos como a soma de várias exponenciais complexas. Daqui vem a definição da série de Fourier, que realiza esta transformação (equação 5), onde Cn é um coeficiente à determinar para cada sinal.

$$f(t) = \sum_{-\infty}^{\infty} Cn.e^{-i\omega t} \quad (5)$$

O gráfico exibido na figura 2 ajuda a compreender como um sinal periódico pode ser gerado pela soma das frequências que compõe o sinal original.

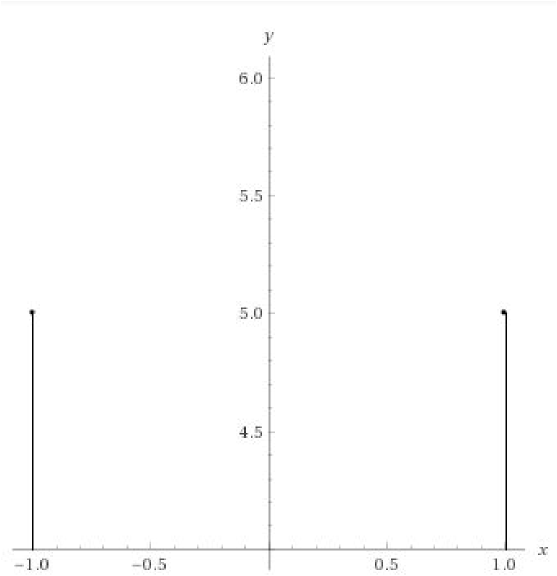


(Figura 2 - Sinal como soma de frequências)

Conhecer as frequências que compõem o sinal torna-se então interessante, e generalizando a série de Fourier para uma soma de infinitos termos, define-se então uma soma de Riemann com limite superior igual a infinito e limite inferior igual a menos infinito, ou seja, uma integral imprópria, conhecida como transformada de Fourier, que tem sua definição na equação 6, onde F(ω) é o sinal no domínio da frequência e f(t) o sinal no tempo.

$$F(\omega) = \int_{-\infty}^{\infty} f(t).e^{-\omega t}dt \quad (6)$$

A transformada de Fourier leva uma função do domínio do tempo ao domínio da frequência, tornando possível então entendermos características do sinal impossíveis de entender analisando apenas no tempo. Tratar de sinais na frequência pode tornar algumas operações e observações muito mais simples, a filtragem de frequências é uma dessas operações.



(Figura 3 - Cossenos com amplitude 5 e frequência de 1Hz na frequência)

Senóides, por exemplo, possuem apenas uma frequência de oscilação, tendo como resultado da transformada de Fourier um impulso com a amplitude da senóide específica na frequência do sinal.

Potência média de um Sinal:

A potência média de um sinal periódico com período T pode ser calculada pela equação 7.

$$\frac{1}{T} \cdot \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)^2 dt \qquad (7)$$

Sinais não periódicos tem período T igual a infinito, tornando a sua potência média igual a zero. Sinais dessa forma são classificados como sinais de energia, que não serão tratados neste relatório.

Autocorrelação de um sinal periódico:

A autocorrelação de um sinal periódico mede quão próximo um sinal está de sí mesmo deslocado no tempo e pode ser calculado pela equação 8.

$$R_x(\tau) = \frac{1}{To} \int_{-\frac{T}{2}}^{\frac{T}{2}} x(t).x(t-\tau)dt \qquad (8)$$

Densidade espectral de potência:

A densidade espectral de um sinal periódico dá a ideia de onde, na frequência, está presente a energia em um período de um sinal periódico, e é dada pela transformada de fourier da autocorrelação deste sinal (equação 9).

$$S_x = F\{R_x(\tau)\} \qquad (9)$$

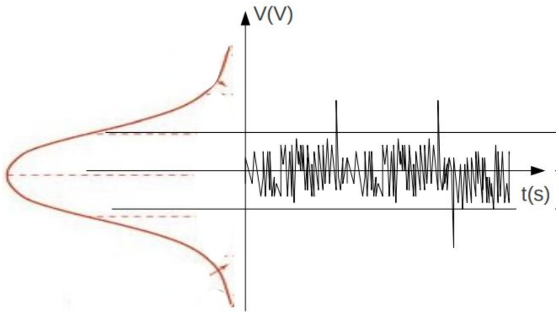
Também pode ser calculada utilizando o sinal no domínio da frequência, observável na equação 10.

$$S_x = \frac{1}{To} \cdot \int_{-\frac{T}{2}}^{\frac{T}{2}} |X(f)|^2.df \qquad (10)$$

Ruído branco Gaussiano

Um ruído, em sistemas digitais, é um sinal que não carrega informação e é oriundo de fontes não propositais, pode ter origem em interferência eletromagnética, térmica, entre outras.

Um ruído branco possui amplitude em todos os valores de frequência do espectro, daí o nome branco. Quando o ruído branco é dito gaussiano ele tem valores de amplitude se comportando como uma gaussiana de média zero e desvio padrão 1.



(Figura 4 - Ruído branco gaussiano)

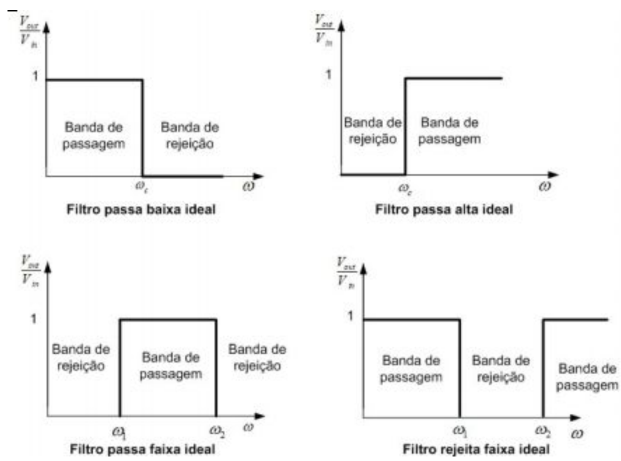
Filtros em frequência:

Filtros em frequência são sinais que ao serem convolucionados com outros sinais, eliminam componentes específicas de frequências normalmente não desejadas.

Um filtro é representado na frequência como uma janela com amplitude conhecida em uma zona de frequência e valor igual a zero no restante do espectro. Ao fazer a convolução no tempo, os sinais são multiplicados na frequência, esta multiplicação de uma janela por um espectro elimina tudo o que estiver na região onde a filtro tem valor igual a zero, limpando o sinal e mantendo as características originais desejadas.

De acordo com o posicionamento desta janela, tem-se um filtro de tipo diferente, eles podem ser:

- **Filtro passa baixa:** Possui uma janela com valor diferente de zero logo no começo do espectro, que decai à zero em uma frequência de corte conhecida. Assim, mantendo as características do sinal em baixa frequência e eliminando frequências maiores que a frequência de corte.
- **Filtro passa faixa:** Este filtro possui uma janela no meio do espectro, entre uma frequência ω_1 e uma frequência ω_2. Frequências dentro da janela são mantidas e valores fora da janela são eliminados.
- **Filtro rejeita faixa:** Funcionam muito parecidos com filtros passa faixa, entretanto este filtro elimina frequências entre as frequências de funcionamento e mantém as frequências fora da janela.
- **Filtro passa alta:** Um filtro passa alta possui janela que vem do infinito até uma frequência de corte, mantendo frequências altas e eliminando frequências abaixo da frequência de corte.



(Figura 5 - Tipos de filtro)

Filtros reais não conseguem eliminar regiões de frequência totalmente, conseguindo apenas atenuar essas regiões em níveis bem aceitáveis, além de não possuírem uma frequência de corte tão definida, mas sim uma tímida curva de atenuação próxima a frequência de corte, considerada onde a atenuação é maior do que -3DB do sinal original.

Seção III

Toda a resolução dos exercícios pode ser observada na seção V, onde estão todos os códigos em matlab.

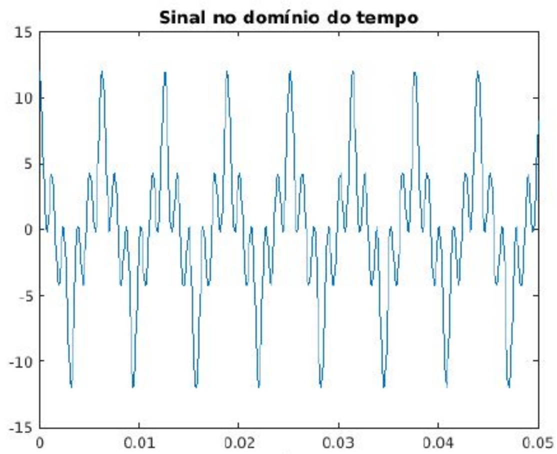
Exercício 1:

1. Gerar um sinal s(t) composto pela somatória de 3 senos com amplitudes de 6 V, 2 V e 4 V e frequências de 1 3 e 5 kHz, respectivamente.

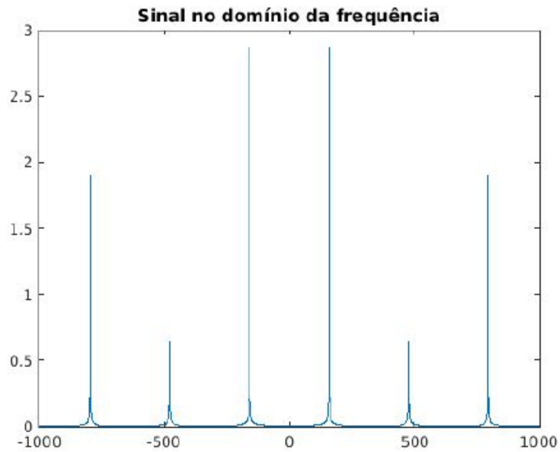
- Os sinais foram criados utilizando o matlab.

2. Plotar em uma figura os três cossenos e o sinal 's no domínio do tempo e da frequência.

- Sinais plotados nas figuras 6 e 7.



(Figura 6 - Exercício 1.2a)



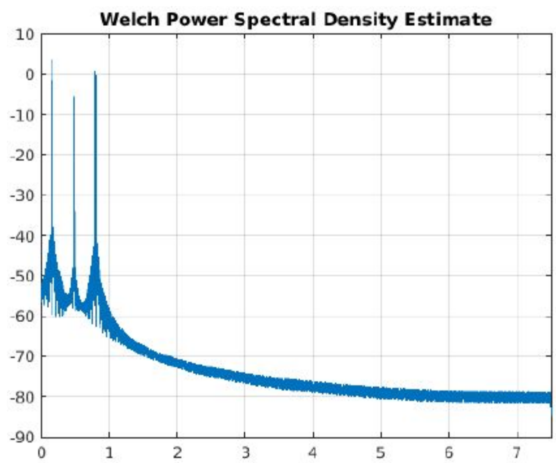
(Figura 7 - Exercício 1.2b)

3. Utilizando a função norm determine a potência média do sinal s.

- A potência média do sinal foi calculada, tendo como resultado: 28.013511.

4. Utilizando a função pwelch plote a Densidade Espectral de Potência do sinal

- Utilizando a função pwelch, foi plotada a densidade espectral de potência do sinal na figura 8.



(Figura 8 - Densidade espectral de potência: Exercício 1.4)

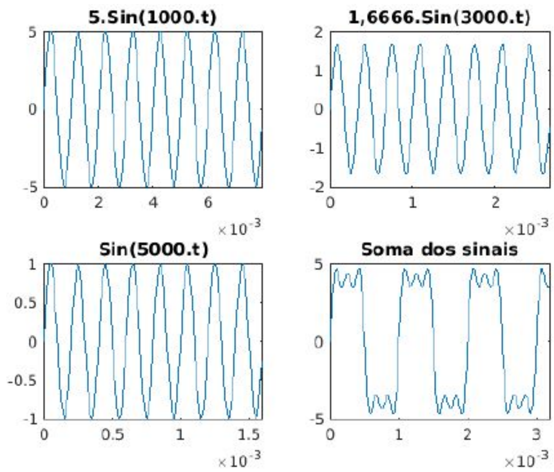
Exercício 2:

1. Gerar um sinal s(t) composto pela somatória de 3 senos com amplitudes de 5 V, 5 3 V e 1 V e frequências de 1 3 e 5 kHz, respectivamente.

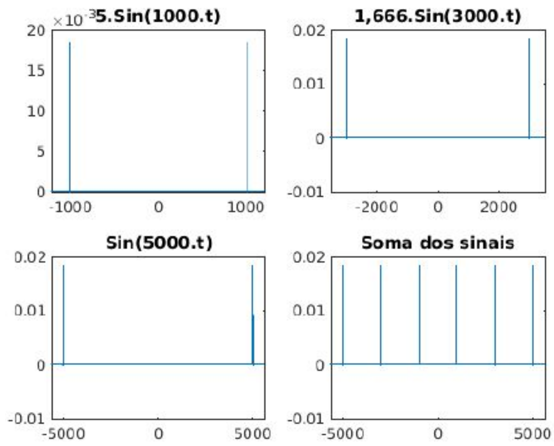
- Os sinais foram gerados como pode ser visto na seção V

2. Plotar em uma figura os três cossenos e o sinal 's no domínio do tempo e da frequência.

- Os sinais foram plotados e podem ser vistos nas figuras 9 e 10.



(Figura 9 - Sinais no domínio do tempo: Exercício 2.2a)



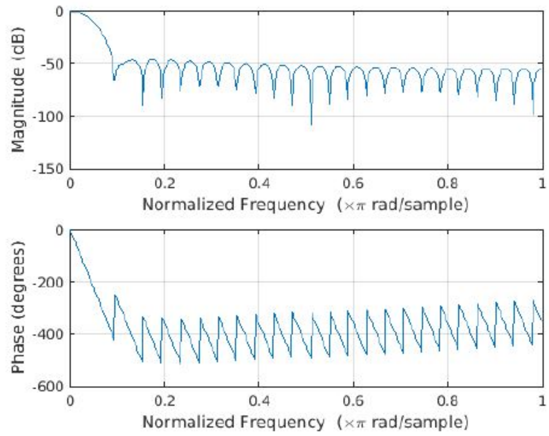
(Figura 10 - Sinais no domínio da frequência: Exercício 2.2b)

3. Gerar 3 filtros ideais:
• Passa baixa (frequência de corte em 2 kHz)
•Passa alta (banda de passagem acima de 4 kHz
•Passa faixa (banda de passagem entre 2 e 4 kHz

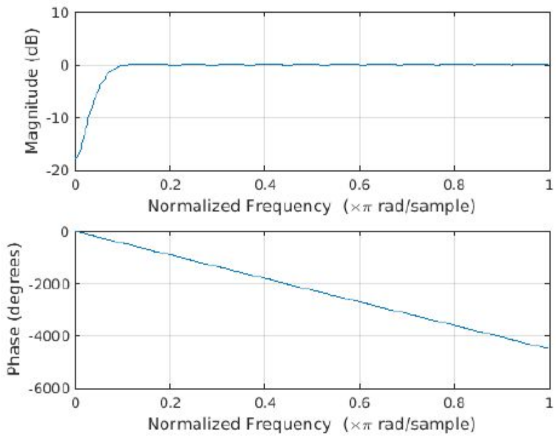
- Sinais foram gerados, como pode ser visto na seção V.

4.Plotar em uma figura a resposta em frequência dos 3 filtros

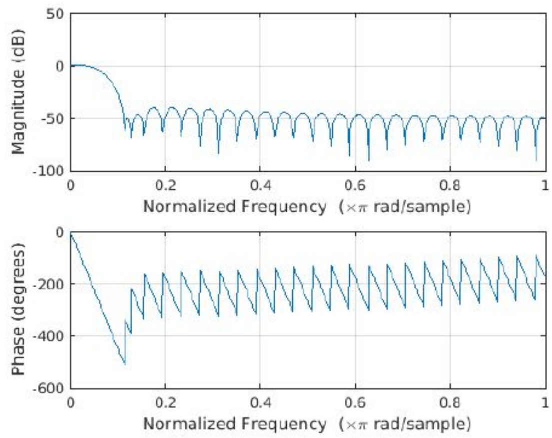
- A resposta em frequência de cada filtro foi plotada utilizando a função freqz e pode ser observada nas figuras 11, 12 e 13.



(Figura 11 - Resposta em frequência do filtro passa baixa: Exercicio 2.4a)



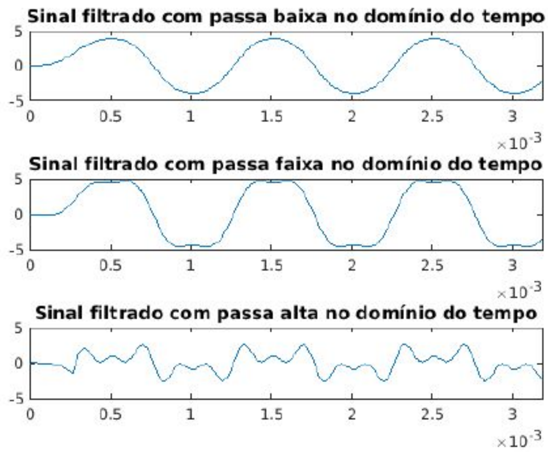
(Figura 12 - Resposta em frequência do filtro passa alta: Exercício 2.4b)



(Figura 13 - Resposta em frequência do filtro passa faixa: Exercício 2.4c)

5. Passar o sinal s(t) através dos 3 filtros e plotar as saídas, no domínio do tempo e da frequência para os 3 casos.

- Os 3 filtros foram passados no sinal s(t), composto pela soma de três senos. Isto pode ser observado na figura 14.



(Figura 14 - Sinal filtrado: Exercício 2.5)

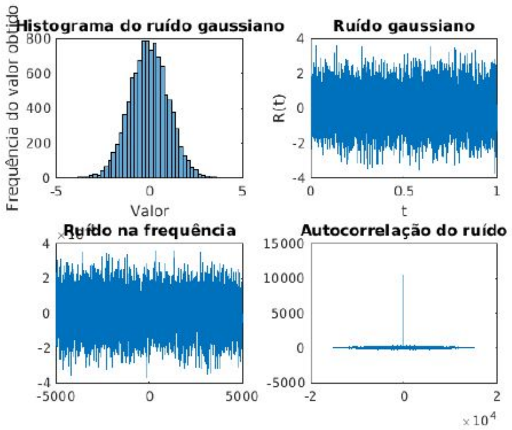
Exercício 3:

1. Gerar um vetor representando um ruído com distribuição normal utilizando a função 'randn' do matlab. Gere 1 segundo de ruído considerando um tempo de amostragem de 1/10k.

- O ruído foi realizado, como pode ser visto na seção V.

2. Plotar o histograma do ruído para observar a distribuição Gaussiana. Utilizar a função 'histogram'

- O histograma foi realizado, como pode ser visto na figura 15.



(*Figura 15 - Ruído Gaussiano: Exercícios 3.2, 3.3 e 3.4)*

3. Plotar o ruído no domínio do tempo e da frequência

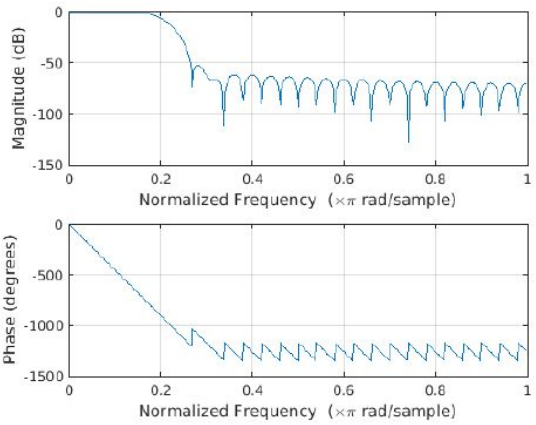
- O ruído foi plotado como pode ser visto na figura 15.

4. Utilizando a função 'xcorr', plote a função de autocorrelação
do ruído.

- A autocorrelação do ruído gaussiano foi calculada e plotada, como pode ser visto na figura 15.

5.Utilizando a função 'filtro=fir1(50,(1000*2)/fs)', realize uma operação de filtragem passa baixa do ruído. Para visualizar a resposta em frequência do filtro projetado, utilize a função 'freqz'.

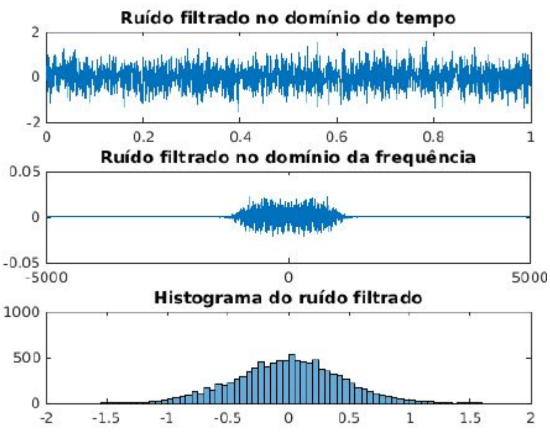
- A filtragem foi realizada, a resposta em frequência do filtro pode ser vista na figura 16.



(Figura 16 - Resposta em frequência do filtro: Exercício 3.5)

6. Plote, no domínio do tempo e da frequência, a saída do filtro e o histograma do sinal filtrado.

- Os gráficos foram plotados e podem ser observados na figura 17.



(Figura 17 - Ruído branco filtrado: Exercício 3.6)

Seção IV

Pode ser observado nos exercícios a importância da transformada de Fourier e como ela descreve perfeitamente o sinal no domínio da frequência. Isto pode ser observado na Seção III, em qualquer um dos exercícios em que acontece filtragem. Só foi possível localizar a região de filtragem após o estudo do sinal na frequência e este tipo de operação tem aplicações diversas em muitas áreas da comunicação, como a modulação, por exemplo.

Percebe-se a excelência da ferramenta do matlab para o estudo e manipulação de sinais. Desde a geração das funções, como do sinal aleatório, como dos filtros, assim como a realização das transformadas como a plotagem das figuras se mostraram bastante satisfatórios.

É perceptível a importância do estudo desta área para o desenvolvimento de sistemas de comunicação, já que tecnologias deste tipo manipulam sinais como foi feito neste experimento em qualquer tipo de comunicação.

Seção V

Toda a seção de códigos no matlab seguem nas próximas páginas, sendo divididas entre os três exercícios propostos:

Table of Contents


```
clc;
clear all;
close all;
```

Exercício 1:

Criando as variáveis de frequência e tempo:

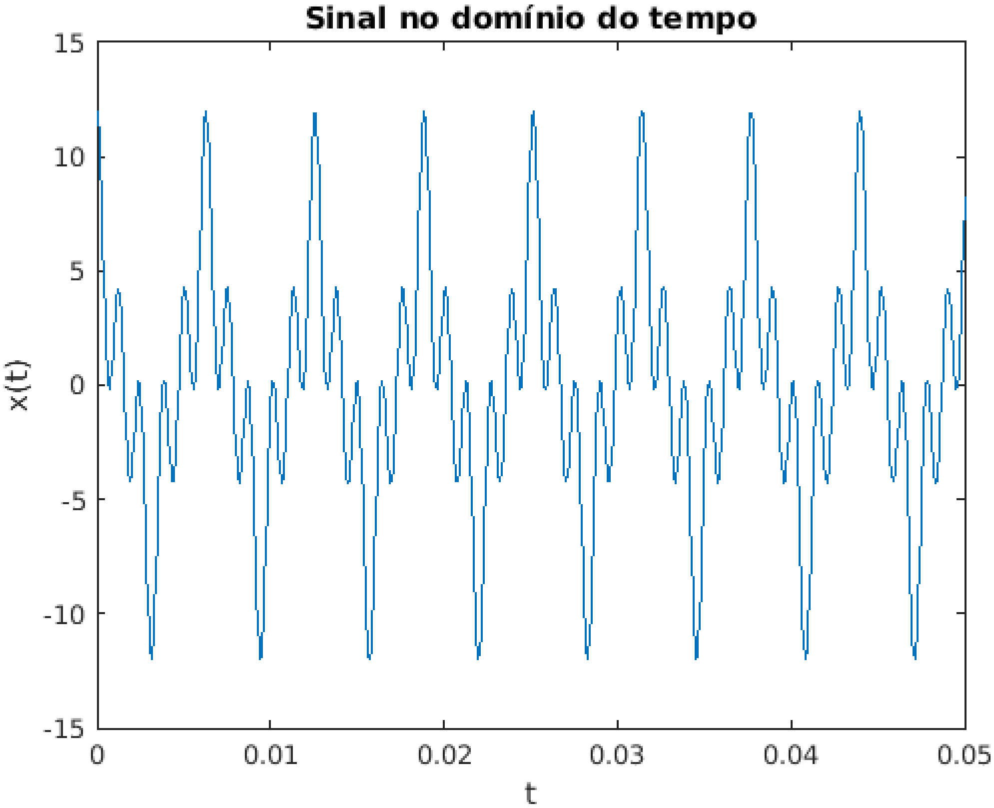
```
f1 = 1000;
f2 = 3000;
f3 = 5000;
fs = 15000;

t = 0:1/fs:1;
```

Criando as senóides no tempo:

```
x1 = 6*cos(f1*t);
x2 = 2*cos(f2*t);
x3 = 4*cos(f3*t);

xt = x1 + x2 + x3;
```

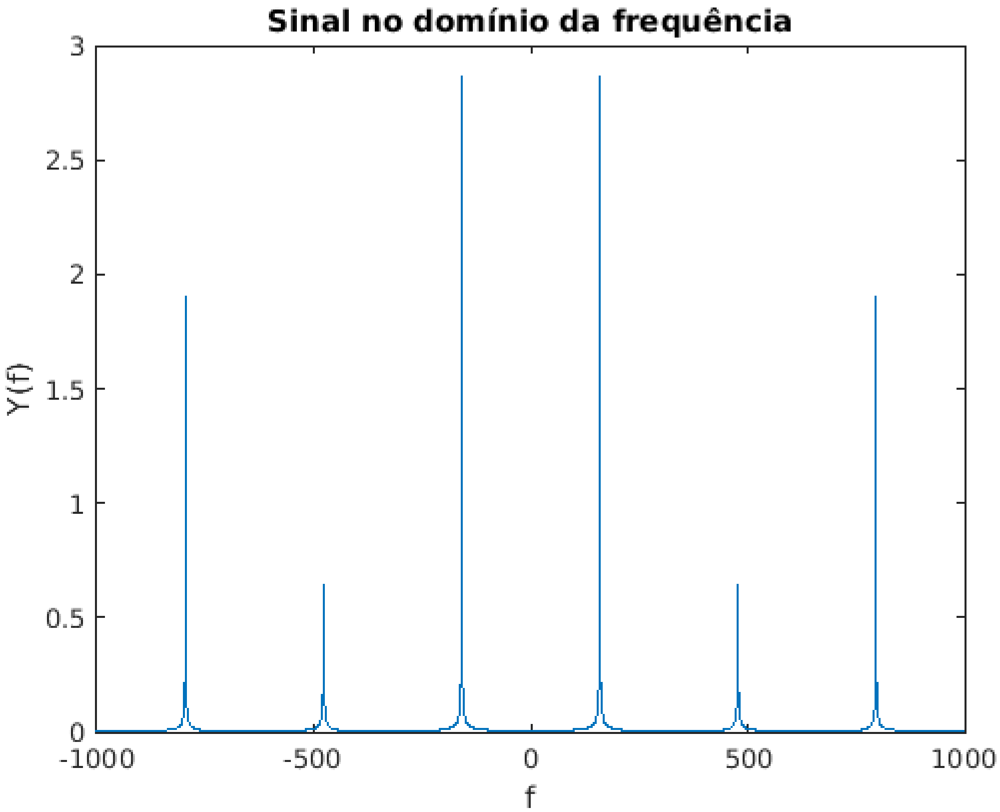
Plotando as funções no tempo:

```
figure(1);
plot(t,xt);
xlim([0 0.05]);
title('Sinal no domínio do tempo');
xlabel('t');
ylabel('x(t)');


Yaux = fft(xt);
Yw = fftshift(Yaux)/length(Yaux);
f = [-fs/2:1:fs/2];
```



Plotando os sinais no domínio da frequência:

```
figure(2)
plot(f, abs(Yw));
xlim([-1000 1000]);
title('Sinal no domínio da frequência');
xlabel('f');
ylabel('Y(f)');
```



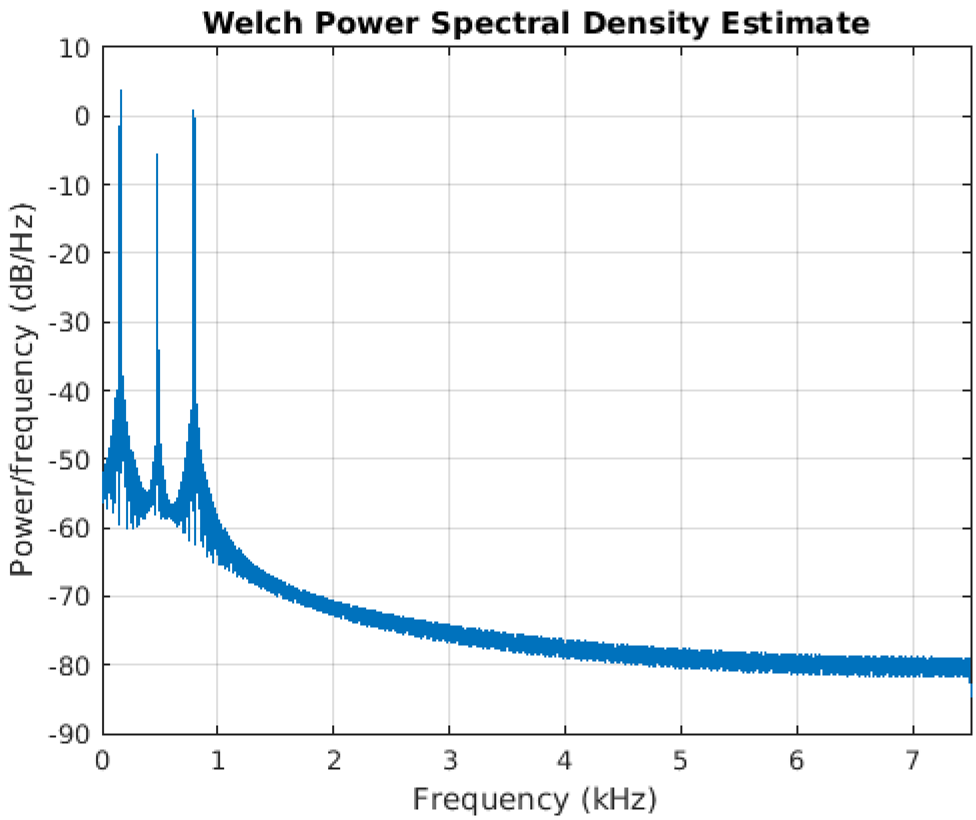
Calculando a potência média do sinal:

```
P = (norm(xt)^2)/length(xt);
fprintf('Potência média do sinal: %f\n', P);
```

Potência média do sinal: 28.013511

Calculando e plotando a densidade espectral de potência do sinal:

```
figure(3);
pwelch(xt,[],[],[],fs);
```



Published with MATLAB® R2015a

Table of Contents


```
%%Exercício 2:
clc;
clear all;
close all;
```

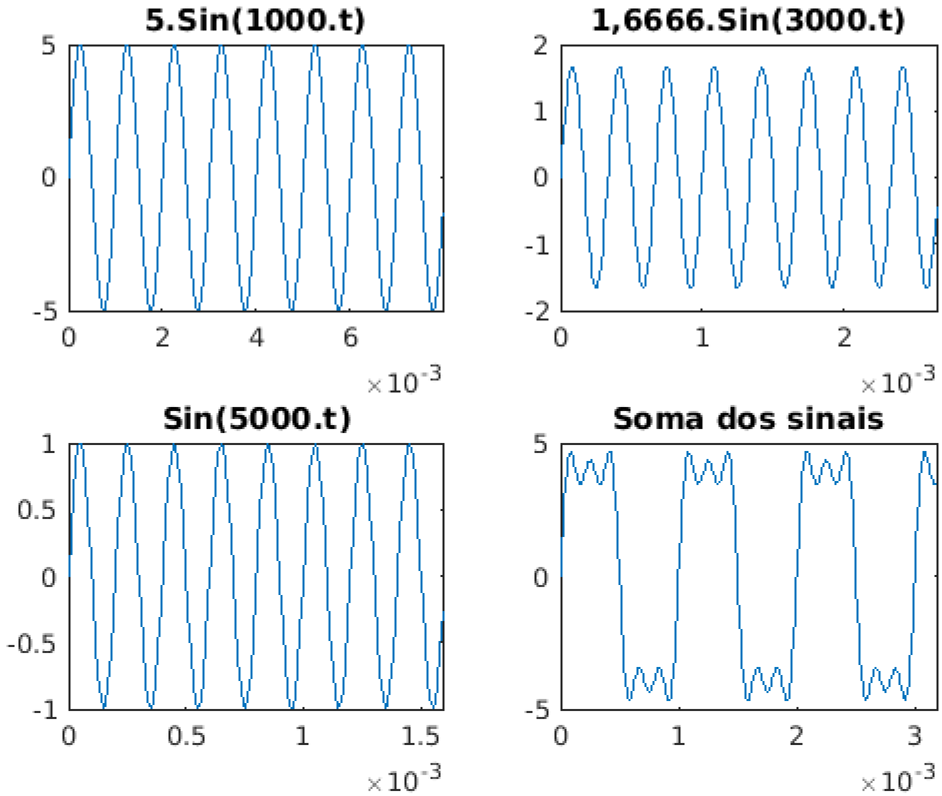
Criando os dados

```
f1 = 2*pi*1000;
f2 = 2*pi*3000;
f3 = 2*pi*5000;
fs = 2*pi*15000;

t = 0:1/fs:1;
```

Criando os sinais e plotando:

```
x1t = 5*sin(f1*t);
x2t = (5/3)*sin(f2*t);
x3t = sin(f3*t);

st = x1t + x2t + x3t;

figure(1);
subplot(2,2,1);
plot(t, x1t);
title('5.Sin(1000.t)');
xlim([0 50/f1]);

subplot(2,2,2);
plot(t,x2t);
title('1,6666.Sin(3000.t)');
xlim([0 50/f2]);

subplot(2,2,3);
plot(t,x3t);
title('Sin(5000.t)');
xlim([0 50/f3]);
```

```
subplot(2,2,4);
plot(t,st);
title('Soma dos sinais');
xlim([0 100/f3]);
```

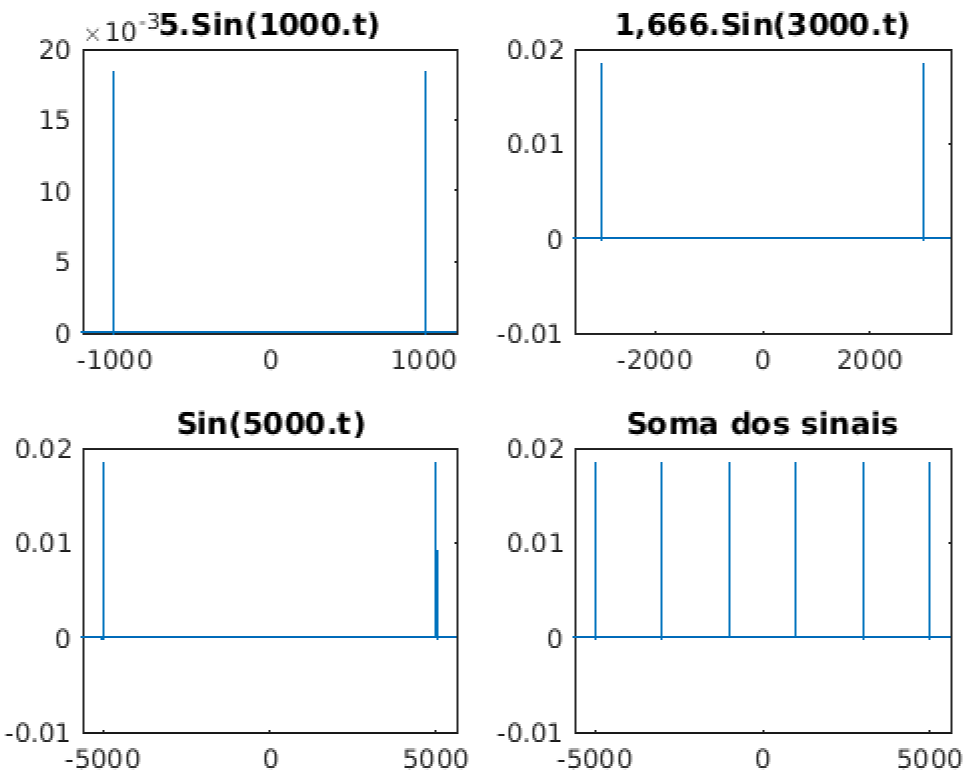


Gerando os sinais na frequência e plotando:

```
f = -fs/2:fs/2;

X1wz = fft(x1t);
X1w = fftshift(X1wz)/length(X1wz);

figure(2);
subplot(2,2,1);
plot(f, X1w);
title('5.Sin(1000.t)');
xlim([-1200 1200]);

X2wz = fft(x2t);
X2w = fftshift(X2wz)/length(X2wz);
subplot(2,2,2);
plot(f, X2w);
title('1,666.Sin(3000.t)');
xlim([-3500 3500]);

X3wz = fft(x3t);
```

```
X3w = fftshift(X3wz)/length(X3wz);
subplot(2,2,3);
plot(f,X3w);
title('Sin(5000.t)');
xlim([-5600 5600]);

Swz = fft(st);
Sw = fftshift(Swz)/length(Swz);
subplot(2,2,4);
plot(f, Sw);
title('Soma dos sinais');
xlim([-5600 5600]);
```

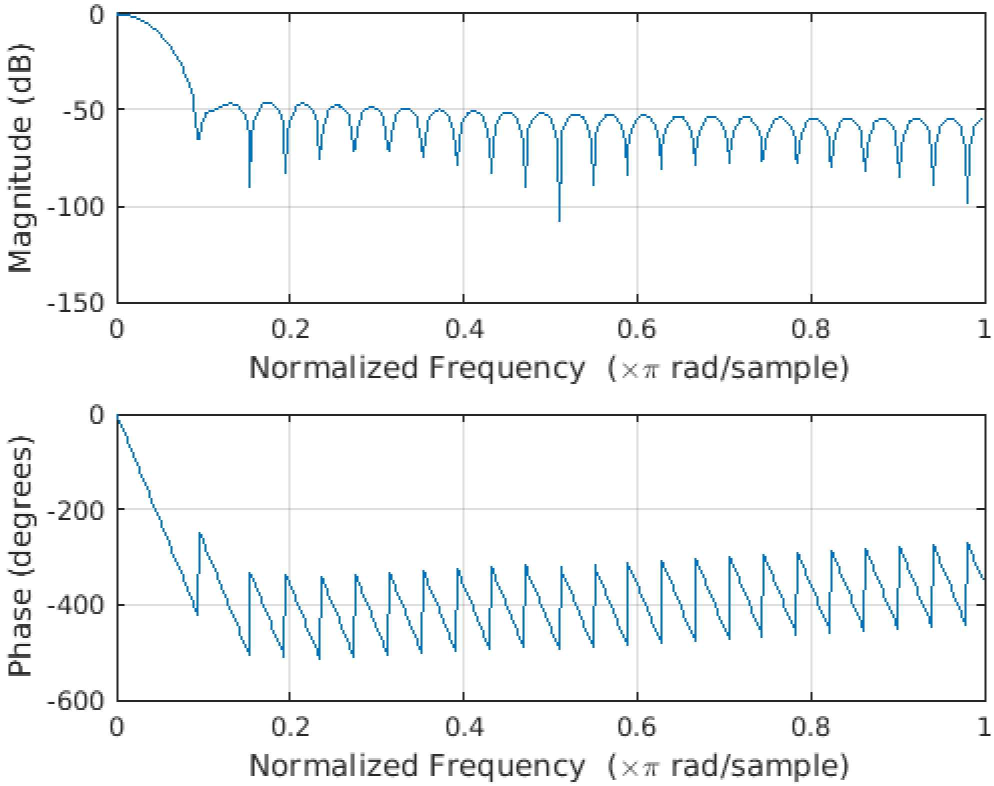
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored

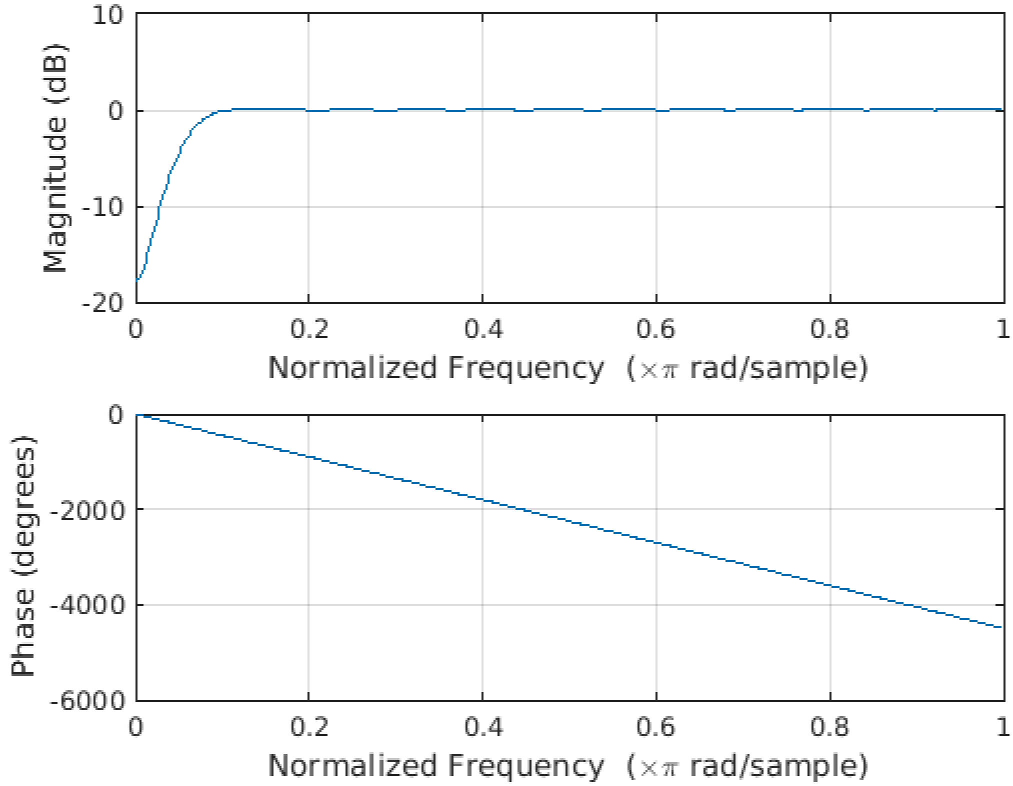


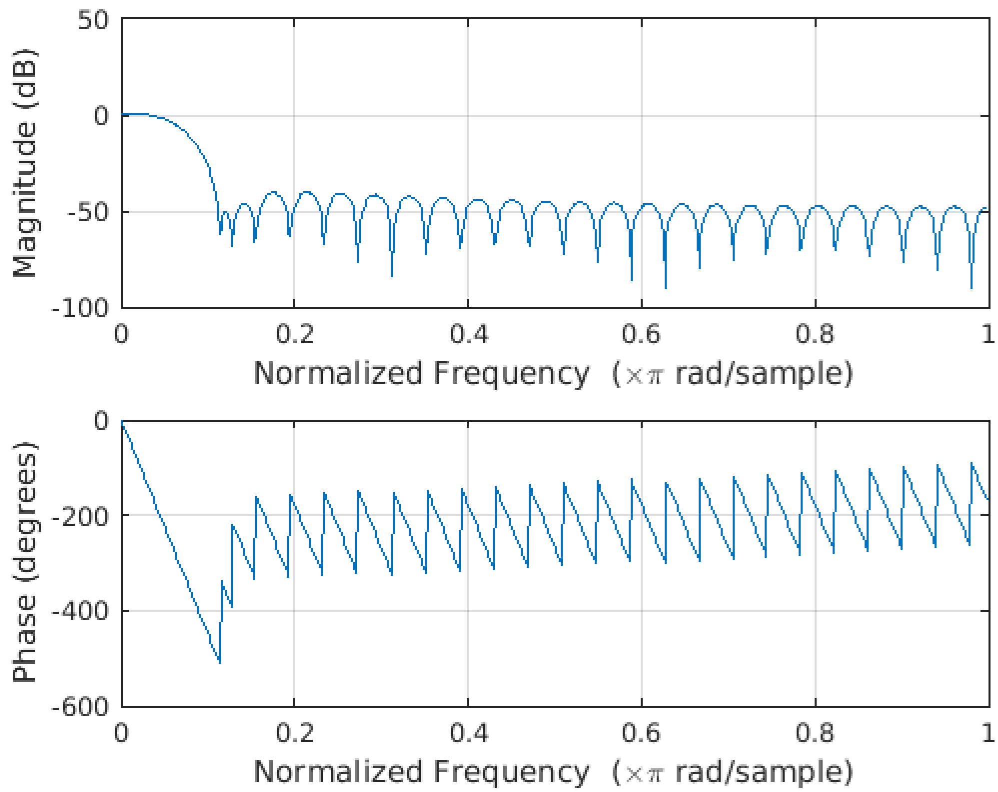
Gerando os filtros:

```
figure(3);
lowfil = fir1(50, (2000)/fs);
freqz(lowfil);

figure(4);
highfil = fir1(50, 4000/fs, 'high');
```

```
freqz(highfil);

figure(5);
middlefil = fir1(50, [2000 4000]/fs);
freqz(middlefil);
```

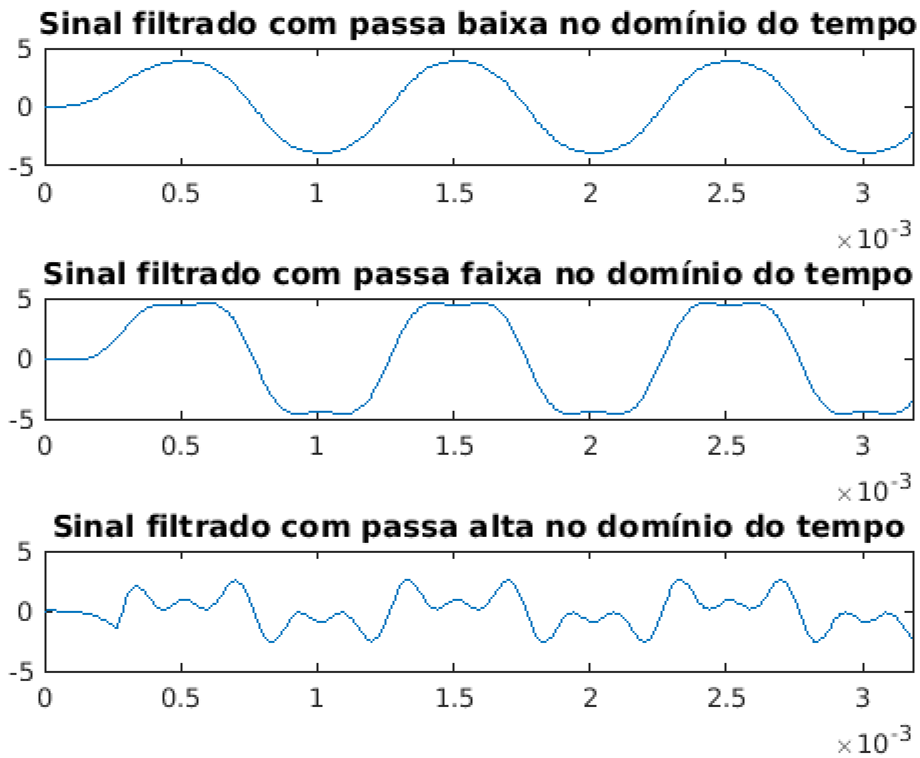



Magnitude (dB)
10
0
-10
-20
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-2000
-4000
-6000
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)


Magnitude (dB)
50
0
-50
-100
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-200
-400
-600
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)

Passando o sinal pelos filtros e plotando:

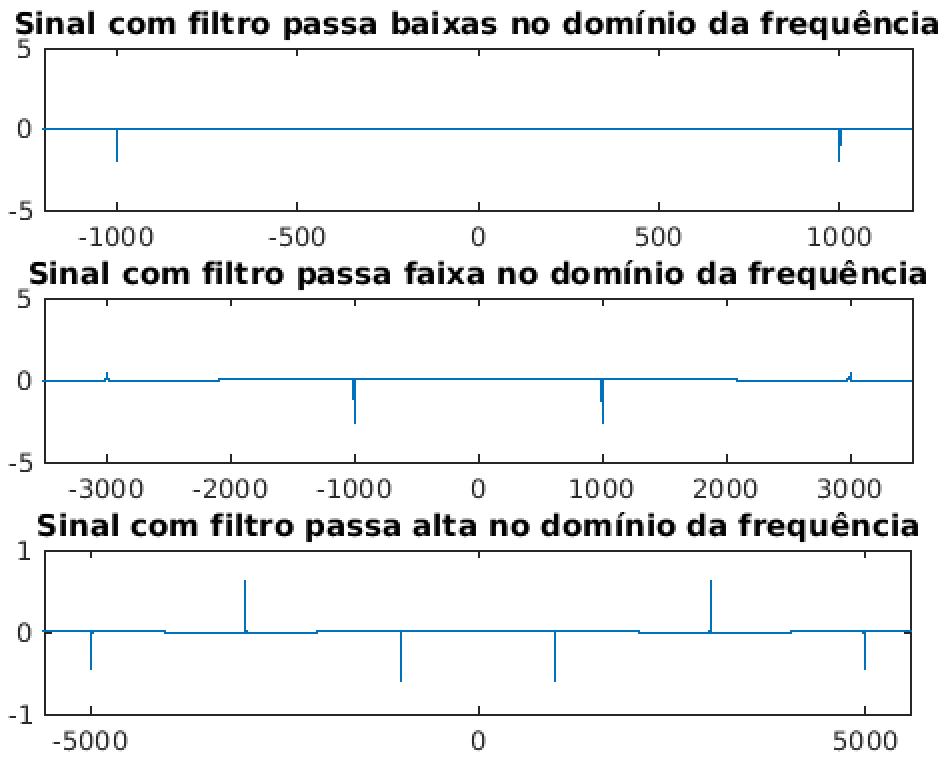
```
figure(6);
subplot(3,1,1);

slowfil = filter(lowfil, 1, st);
plot(t, slowfil);
xlim([0 100/f3]);
title('Sinal filtrado com passa baixa no domínio do tempo');

smiddlefil = filter(middlefil, 1, st);
subplot(3,1,2);
plot(t,smiddlefil);
xlim([0 100/f3]);
title('Sinal filtrado com passa faixa no domínio do tempo');

shighfil = filter(highfil,1,st);
subplot(3,1,3);
plot(t,shighfil);
xlim([0 100/f3]);
title('Sinal filtrado com passa alta no domínio do tempo');
```



Gerando e plotando os sinais filtrados no domínio da frequência:

```
figure(7);

Slowfilz = fft(slowfil);
Slowfilw = fftshift(Slowfilz)/length(Slowfilz);
subplot(3,1,1);
plot(f, Slowfilw);
title('Sinal com filtro passa baixas no domínio da frequência');
xlim([-1200 1200])

Smiddlefilz = fft(smiddlefil);
Smiddlefilw = fftshift(Smiddlefilz)/length(Smiddlefilz);
subplot(3,1,2);
plot(f, Smiddlefilw);
title('Sinal com filtro passa faixa no domínio da frequência');
xlim([-3500 3500]);

Shighfilz = fft(shighfil);
Shighfilw = fftshift(Shighfilz)/length(Shighfilz);
subplot(3,1,3);
plot(f,Shighfilw);
title('Sinal com filtro passa alta no domínio da frequência');
xlim([-5600 5600]);
```

```
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
```



Published with MATLAB® R2015a

Table of Contents


```
clc;
clear all;
close all;
```

Exercício2

Gerando a frequência de amostragem e tempo de amostragem:

```
fs = 10000;
t = 0:1/fs:1;
```

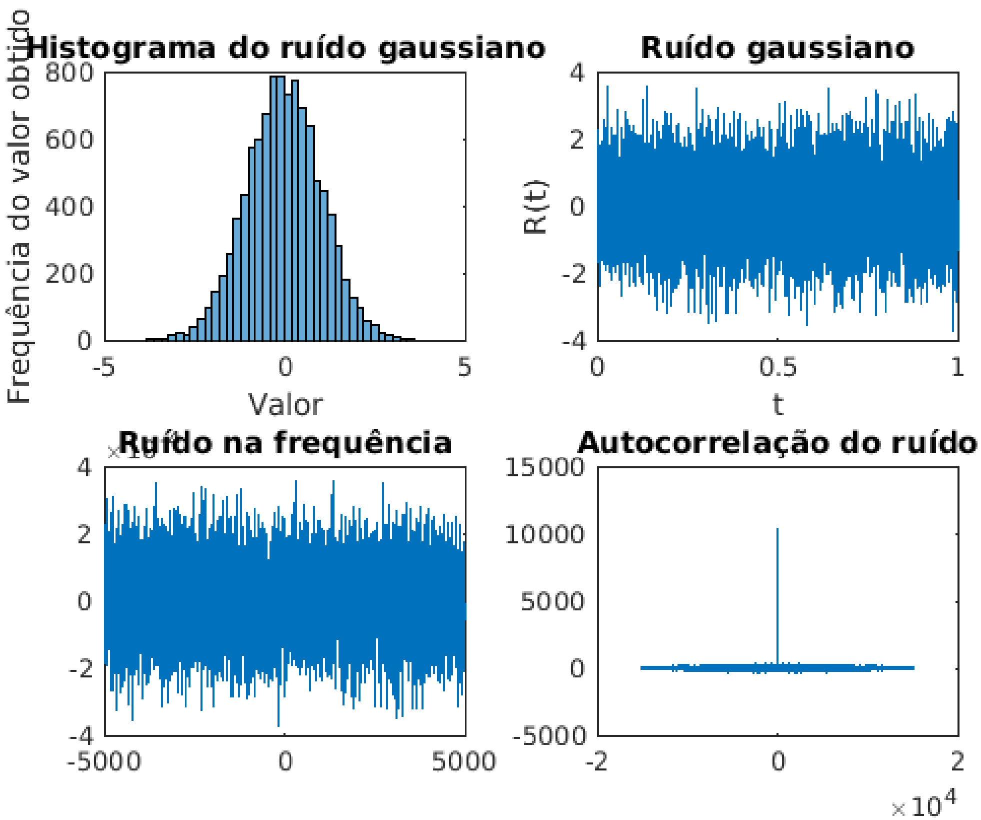
Gerando o ruído gaussiano:

```
rt = randn(1,length(t));
```

Tratando o ruído: Plotando histograma, domínio no tempo, frequência e autocorrelação:

```
figure(1);
subplot(2,2,1);
histogram(rt);
title('Histograma do ruído gaussiano');
xlabel('Valor');
ylabel('Frequência do valor obtido');

%Ruído no tempo:
subplot(2,2,2);
plot(t, rt);
title('Ruído gaussiano')
xlabel('t');
ylabel('R(t)');

% Ruído na frequência:
Yrt = fft(rt);
Yw = fftshift(rt)/length(rt);
```

```
f = [-fs/2:fs/2];

subplot(2,2,3);
plot(f, Yw);
title('Ruído na frequência');

% Autocorreçação

tx = linspace(-15000,15000, 20001);
Rx = xcorr(rt);
subplot(2,2,4);
plot(tx, Rx);
title('Autocorrelação do ruído');
```

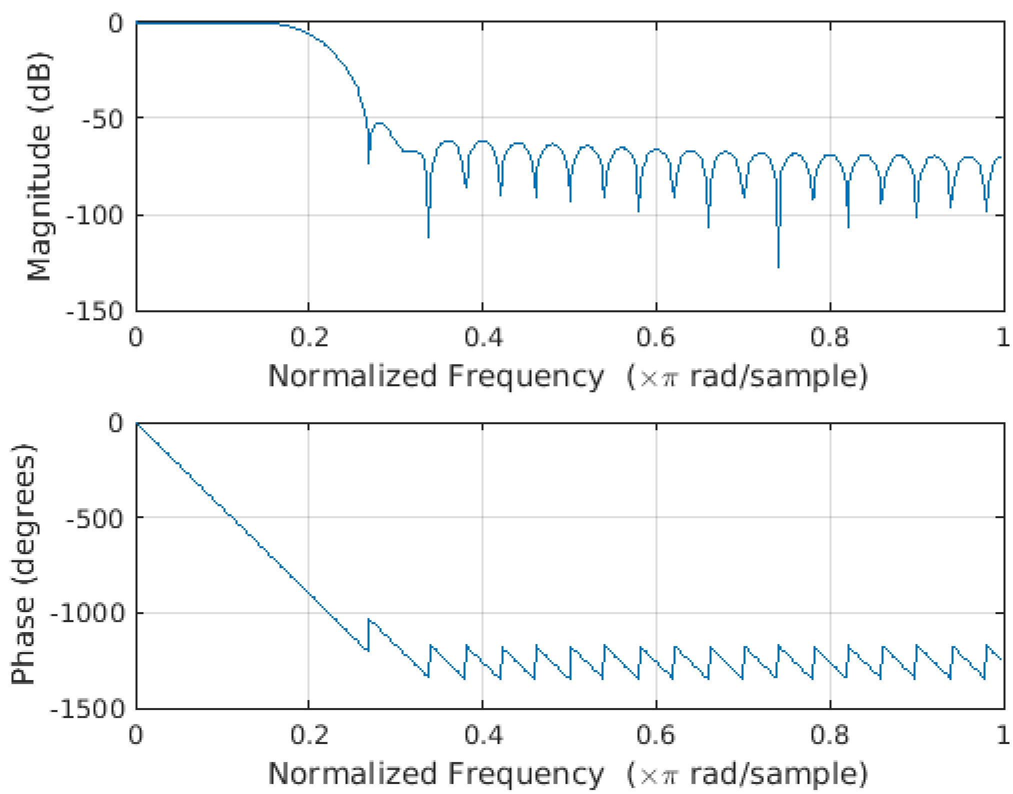


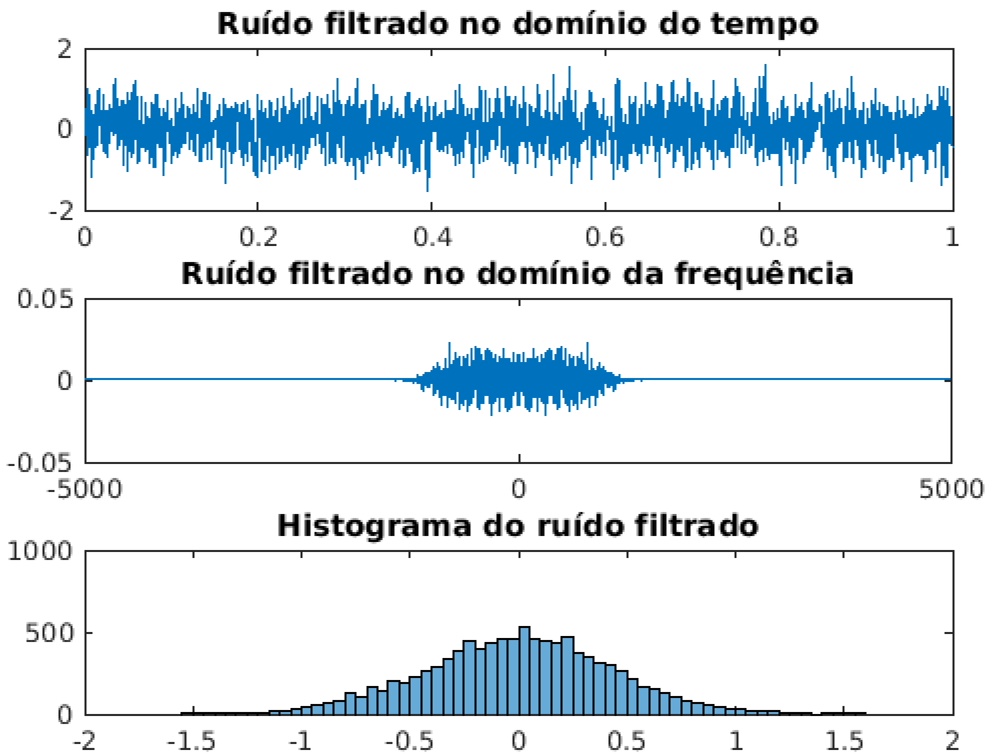
Filtrando o ruído e o plotando:

```
filtro=fir1(50,(1000*2)/fs);
figure(2);
freqz(filtro);
rfil = filter(filtro, 1, rt);

figure(3);
subplot(3,1,1);
plot(t, rfil);
title('Ruído filtrado no domínio do tempo');
```

```
Rfilws = fft(rfil);
Rfilw = fftshift(Rfilws)/length(Rfilws);

subplot(3,1,2);
plot(f, Rfilw);
title('Ruído filtrado no domínio da frequência');

subplot(3,1,3);
histogram(rfil);
title('Histograma do ruído filtrado');
```

Warning: Imaginary parts of complex X and/or Y arguments ignored





Published with MATLAB® R2015a