



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Sistemas de Comunicação I

Relatório 01 - Revisão de Sinais de Espectro

Arthur Cadore Matuella Barcella

04 de Março de 2024

Sumário

| | | |
|----------|--|-----------|
| 1 | Descrição de desenvolvimento | 3 |
| 2 | Conceitos teóricos utilizados | 3 |
| 3 | Apresentação e comentários dos gráficos | 3 |
| 3.1 | Item 1: | 3 |
| 3.2 | item 2: | 5 |
| 3.3 | Item 3: | 8 |
| 4 | Conclusão | 12 |
| 5 | Scripts Utilizados: | 12 |
| 5.1 | Gerar um sinal $s(t)$ composto pela somatória de 3 senos com amplitudes de 6V, 2V e 4V e frequências de 1, 3 e 5 kHz, respectivamente. | 12 |
| 5.2 | Plotar em uma figura os três cossenos e o sinal 's' no domínio do tempo e da frequência. | 13 |
| 5.3 | Utilizando a função 'norm', determine a potência média do sinal 's'. | 14 |
| 5.4 | Utilizando a função 'pwelch', plote a Densidade Espectral de Potência do sinal 's'. | 14 |
| 5.5 | Gerar um sinal $s(t)$ composto pela somatória de 3 senos com amplitudes de 5V, 5/3V e 1V e frequências de 1, 3 e 5 kHz, respectivamente. | 15 |
| 5.6 | Plotar em uma figura os três cossenos e o sinal 's' no domínio do tempo e da frequência. | 16 |
| 5.7 | Gerar 3 filtros ideais e plotar a resposta em frequência para os filtros: | 17 |
| 5.8 | Plotar o sinal 's' através dos filtros e plotar as saídas no domínio do tempo e no domínio da frequência. | 18 |
| 5.9 | 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. | 18 |
| 5.10 | Plotar o histograma do ruído para observar a distribuição Gaussiana. Utilizar a função 'histogram'. | 19 |
| 5.11 | Plotar o ruído no domínio do tempo e da frequência | 19 |
| 5.12 | Utilizando a função 'xcorr', plote a função de autocorrelação do ruído. | 19 |
| 5.13 | 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'. | 20 |
| 5.14 | Plote, no domínio do tempo e da frequência, a saída do filtro e o histograma do sinal filtrado | 20 |

1 Descrição de desenvolvimento

O objetivo deste relatório é apresentar os conceitos básicos de tratamento de sinais e modulação em amplitude, de maneira que possamos entender como a informação carregada por um sinal base pode ser modulada para altas frequências e depois demodulada para termos a informação novamente.

2 Conceitos teóricos utilizados

Os principais conceitos teóricos utilizados neste relatório são os seguintes:

- Modulação AM: Modulação por amplitude, onde a amplitude da modulante altera a portadora para que a portadora carregue a informação pelo espaço.
- sinal modulante: O sinal modulante é o que contém de fato a informação que o usuário/dispositivo deseja transmitir, porém inviável para a transmissão direta devido às suas características, tipicamente o comprimento de onda.
- sinal portador: A portadora é o que permite a transmissão do sinal, suas características são ideais para o meio onde a onda irá se propagar, a portadora é influenciada pela modulante de maneira em que o receptor consiga extrair a informação transmitida.
- Filtros de sinal: Os filtros de sinal são responsáveis por isolar as frequências que desejamos absorver das frequências que devem ser eliminadas. Em um filtro passa-baixas por exemplo, as frequências mais altas são todas eliminadas, o que é vital para transmissão em cenários de interferência de canais/faixas de frequência vizinhos.
- Demodulação AM: O processo de modulação ocorre ao multiplicar o sinal modulante pelo sinal da portadora, existe também o processo contrário, de demodulação, onde novamente multiplicamos o sinal recebido pela portadora a fim de receber a modulante pura. O processo de demodulação é vital para a recuperação da mensagem no destinatário.

3 Apresentação e comentários dos gráficos

3.1 Item 1:

Gerar um sinal $s(t)$ composto pela somatória de 3 senos com amplitudes de 6V, 2V e 4V e frequências de 1, 3 e 5 kHz, respectivamente e plotar em uma figura os três cossenos e o sinal 's' no domínio do tempo e da frequência.

Podemos notar abaixo os sinais no domínio do tempo gerados, inicialmente, temos 3 componentes cossenos (sub-figura 1, 2 e 3) que posteriormente são combinadas em um único sinal $s(t)$, visto no canto inferior direito da imagem abaixo.

FONTE: Elaborado Pelo Autor

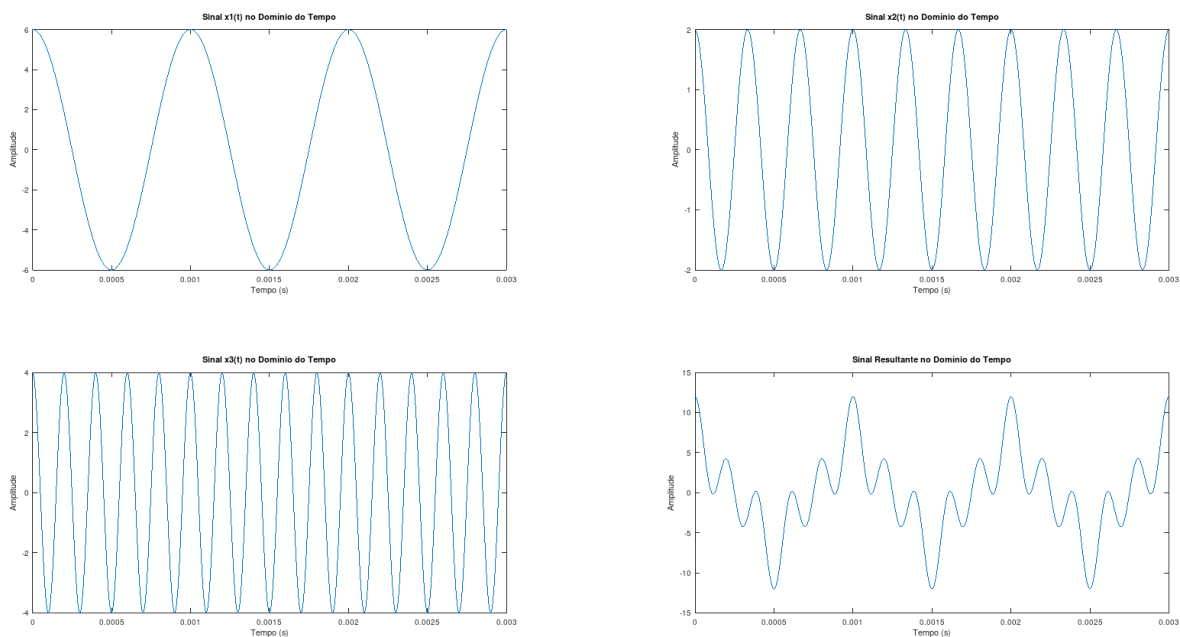


Figura 1: Plot do sinal de $x(t)$ composto por soma de cossenos

Para cada componente no domínio do tempo, foi realizado uma FFT para verificar seu estado no domínio da frequência, abaixo podemos validar os resultados.

FONTE: Elaborado Pelo Autor

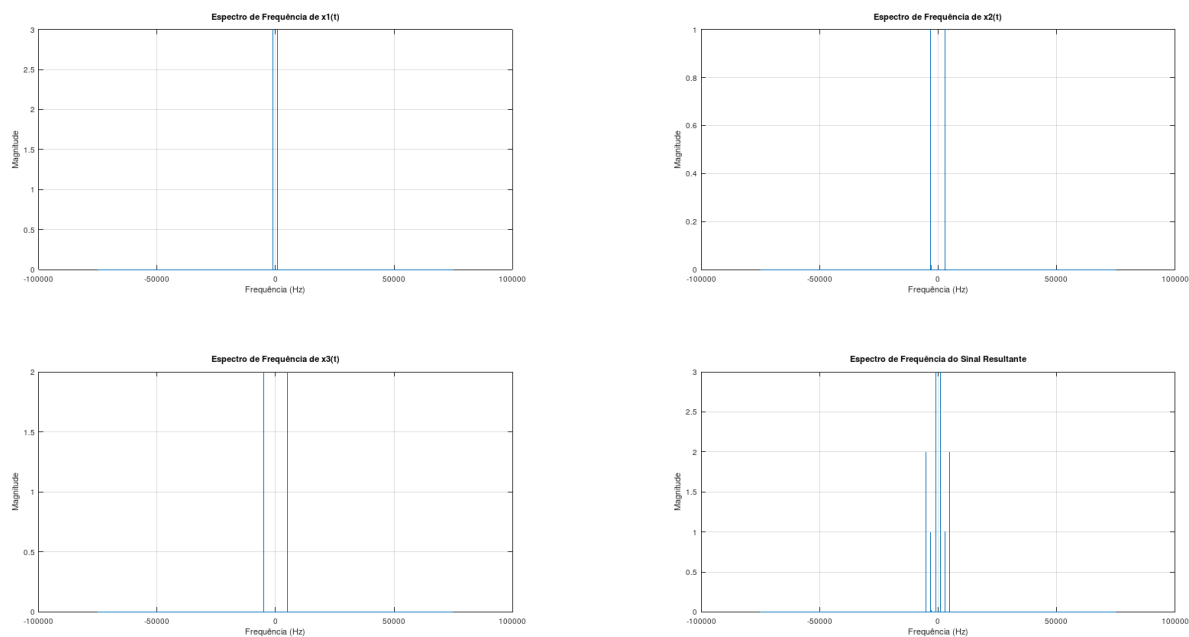


Figura 2: Plot do sinal de $x(t)$ composto por soma de cossenos (domínio da frequência)

Podemos notar que, ao aumentar a frequência do sinal (visível no domínio do tempo), o impulso apresentado no domínio da frequência se distancia do eixo, porém como as frequências entre os três sinais são diferentes, ao combina-los, não notamos um aumento de amplitude em uma das componentes, mas sim a coexistência das três componentes no mesmo sinal resultante.

Em seguida, utilizando a função "norm", podemos determinar a potência média do sinal. Após aplicar o método solicitado, obtive o valor de 28W, também utilizei outros dois métodos para calculá-la,

um utilizando somatório da função "x-t" e tirando sua média posteriormente, e outro método utilizando o somatório da potência contínua e potência alternada calculadas separadamente, conforme é possível notar no trecho de código abaixo:

```
1 % Calculando a potência média total do sinal (metodo 1):
2 potencia_media = norm(x_t).^2 / length(x_t)
3
4 % Calculando a potência AC e DC do sinal;
5 Pot_AC = var(x_t)
6 Pot_DC = mean(x_t)^2
7
8 % Calculando a potência média total do sinal (metodo 2):
9 Pot_Media_Total = Pot_AC + Pot_DC
10
11 % Calculando a potência média total do sinal (metodo 3):
12 Pot_Media_Total = 1/length(x_t)*sum(x_t.^2)
```

Para calcular a densidade espectral, utilizei a função solicitada pelo questionário, podemos validar o resultado na imagem abaixo:

FONTE: Elaborado Pelo Autor

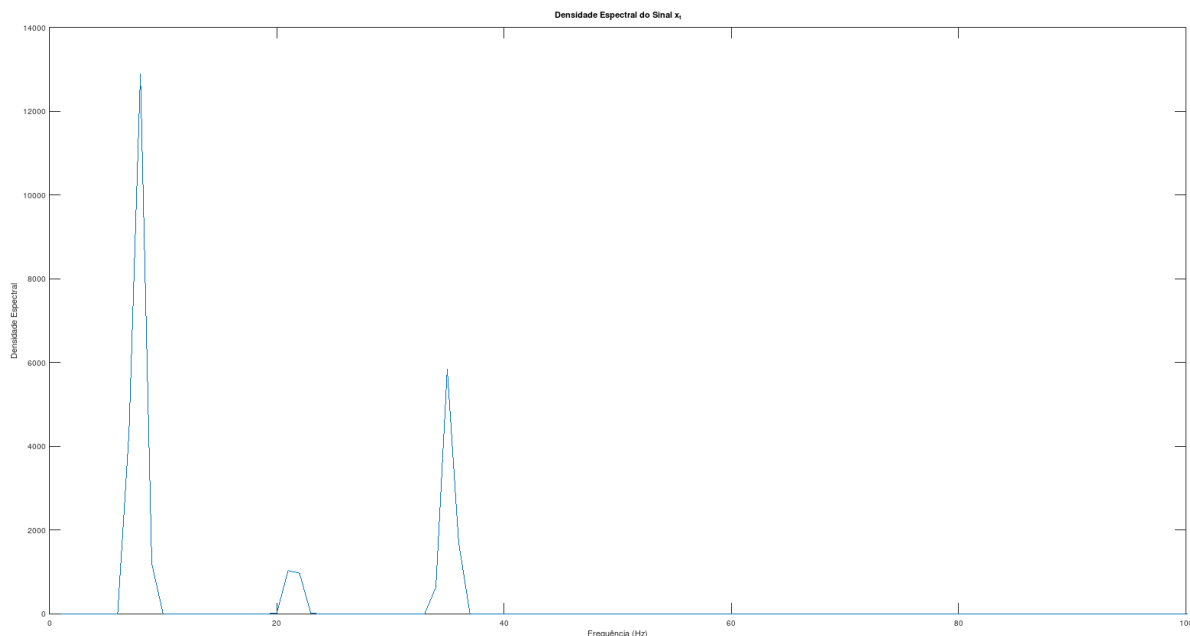


Figura 3: Densidade espectral do sinal de St

O gráfico acima mostra onde a maioria da potência está localizada no espectro, podemos validar que uma grande parte do sinal encontra-se no início do espectro eletromagnético, logo em seguida, temos a sua segunda harmônica e na sequência a terceira harmônica.

Pode-se notar que neste sinal as harmônicas ímpares detêm a maior parte da potência do sinal, o que ocorre em outras formas de onda, como a onda quadrada por exemplo.

3.2 item 2:

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

Novamente como na questão anterior, iniciamos o desenvolvimento partindo dos sinais que irão compor o $s(t)$, portanto, os três cossenos com diferentes frequências e amplitudes são gerados e

plotados individualmente abaixo, seguindo do sinal resultante da soma desses componentes.

FONTE: Elaborado Pelo Autor

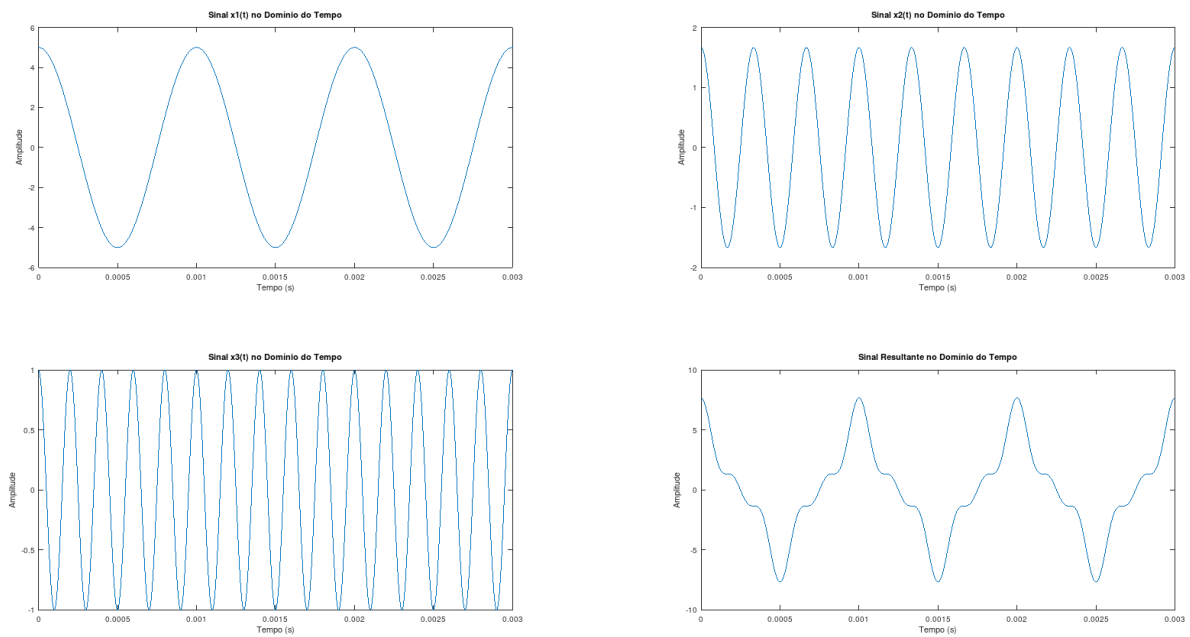


Figura 4: Sinal de St e suas componentes no dominio do tempo

Em seguida, a FFT de cada sinal é calculada, obtendo os gráficos apresentados abaixo, note que novamente, para cada sinal, a um impulso correspondente a sua frequência de propagação. Note que ao juntar os três sinais no sinal resultante, temos que sua resposta em frequência na verdade é a soma de todos os impulsos individuais.

FONTE: Elaborado Pelo Autor

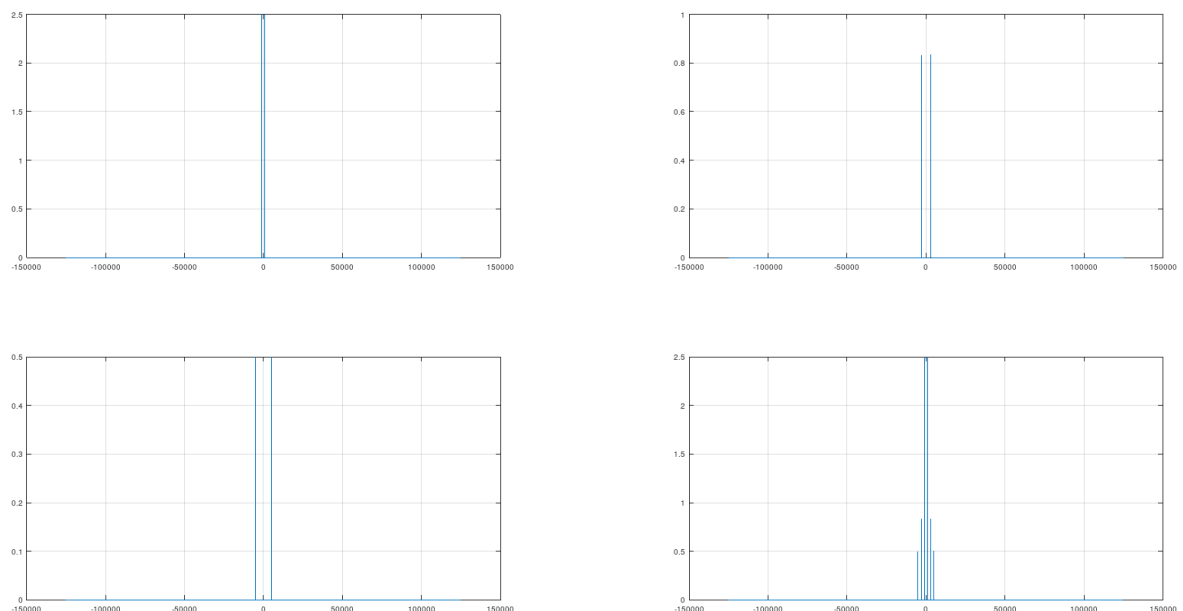


Figura 5: Sinal de St e suas componentes no dominio da frequencia

Em seguida, conforme solicitado foram gerados três diferentes filtros ideais através de vetores de zeros e uns, o primeiro filtro correspondente a PB (Passa Baixa), o segundo correspondente a PA (Passa Alta) e o terceiro correspondente a PF (Passa Faixa).

Para verificar se de fato os vetores configurados estão de acordo com a necessidade, o plot de sua resposta em frequência (eixo da frequência / amplitude) foi plotado, abaixo está o resultado para os três diferentes filtros.

FONTE: Elaborado Pelo Autor

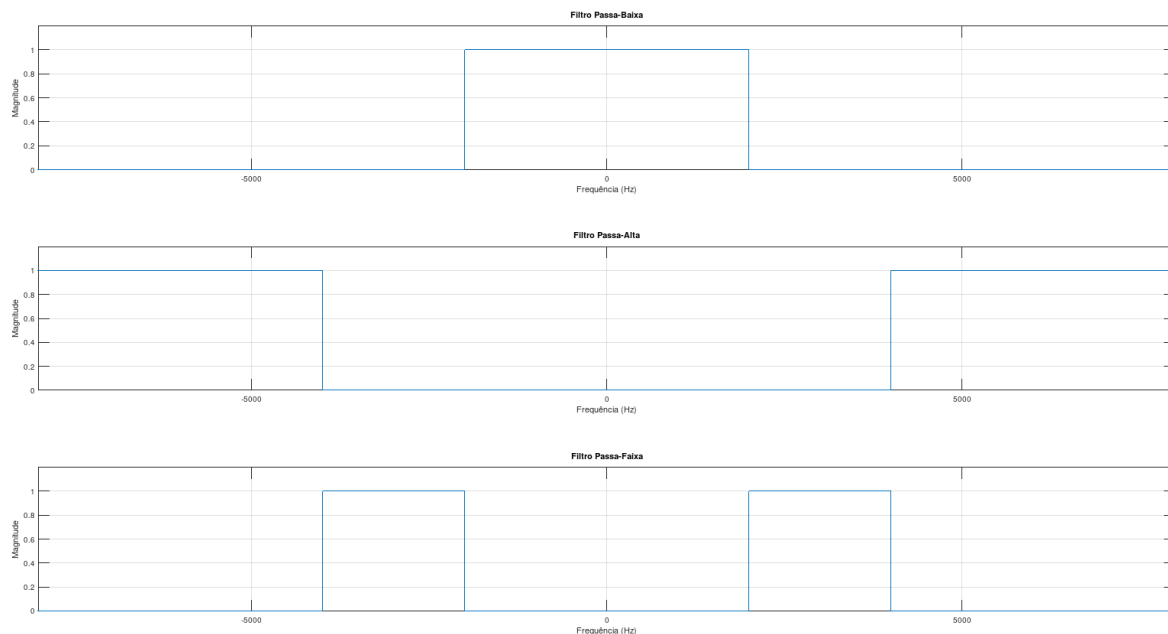


Figura 6: Resposta na frequência dos filtros PF, PA e PB

Note que o primeiro filtro da imagem cobre a parte inferior do eixo da frequência (onde se localiza o eixo), este filtro é o passa baixas.

Na sequência, temos um filtro que está eliminando toda a faixa de frequências baixas e permitindo a passagem apenas das maiores frequências, neste ponto, temos o filtro passa alta.

Por último, temos um filtro que elimina a origem e também as frequências mais altas, mas mantém uma faixa entre elas por onde o sinal pode ser absorvido, esse é o filtro Passa Faixa.

Por fim, simulando uma transmissão digital, ao receber o sinal precisamos filtra-lo para garantir que nenhuma (idealmente) componente de alta ou baixa frequência seja incorporada ao sinal que vamos interpretar.

Desta forma, após gerar os filtros, o mesmo sinal $s(t)$ foi inserido nos três diferentes filtros para verificarmos sua resposta em frequência e também no domínio do tempo, cujo o resultado está explicito abaixo:

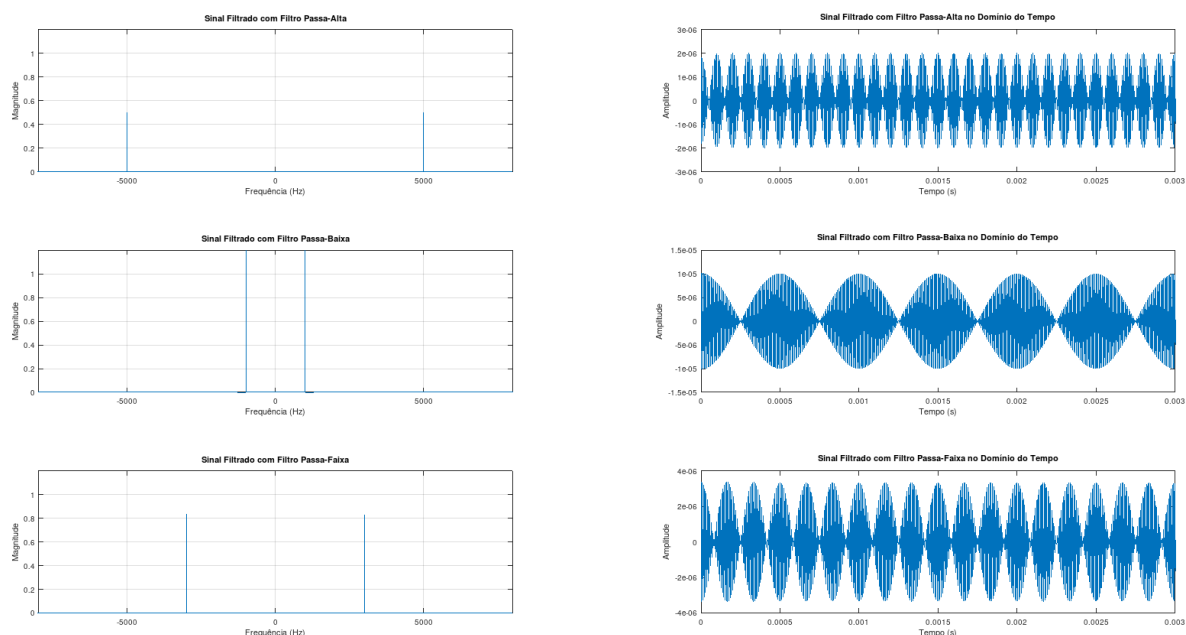


Figura 7: Sinal filtrado - Dominio do tempo e frequência

No primeiro caso, temos o filtro passa alta, note que ao multiplicar o sinal de $s(t)$ por esse filtro, apenas a componente de 5000Hz foi mantida, desta forma, tornando este o sinal com a maior frequência após a filtragem, note que há um reflexo desta característica no domínio do tempo, pois o primeiro sinal é também o que possui o menor período.

No segundo caso, temos o filtro passa baixa, onde após aplicar a filtragem apenas a componente 1000Hz foi mantida, desta forma tornando este o sinal resultante com o maior período entre os três.

Por fim, temos o sinal de filtrado pelo passa-faixa, note que este ficou entre o PB e o PF, devido a única componente ter permanecido foi a de 3000Hz. A vantagem deste tipo de filtro é a versatilidade em capturar apenas o que o usuário/receptor deseja capturar.

Apenas como exemplo, o filtro de passa-faixa poderia ser ajustado para trabalhar por exemplo na faixa de 2 á 10KHz, e neste caso deixaria passar tanto o sinal de 3KHz quanto o de 5KHz.

3.3 Item 3:

Para iniciar a terceira tarefa, inicialmente gerei um ruído branco com 10 mil amostras (de 0 á 1 segundo). O ruído foi gerado através da função 'randn', conforme apresentado abaixo:

FONTE: Elaborado Pelo Autor

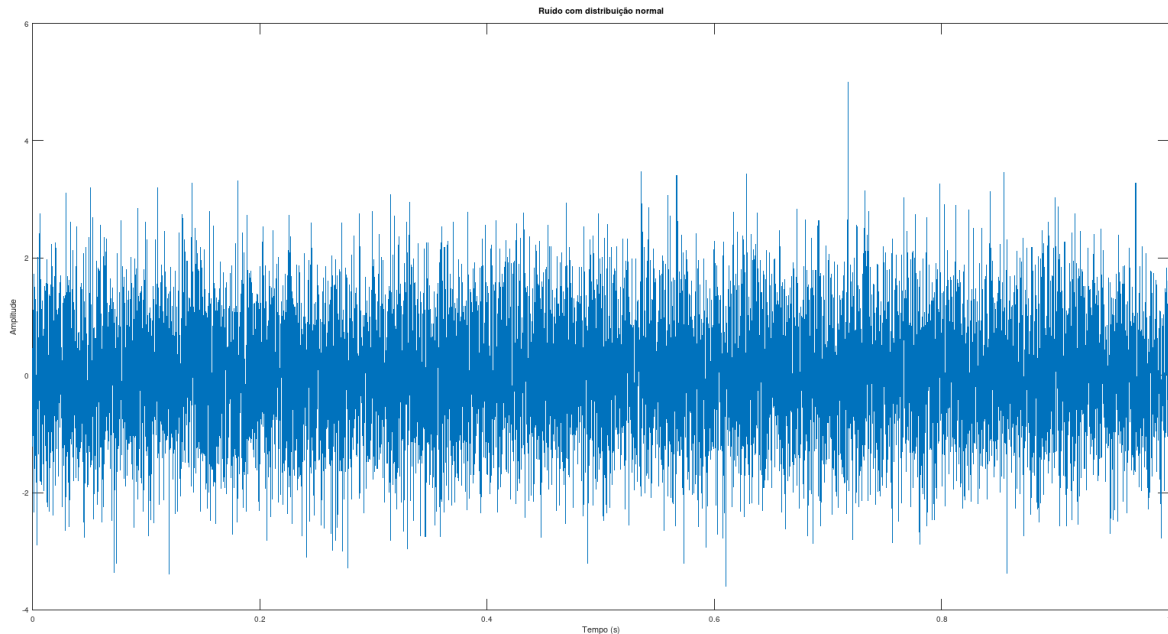


Figura 8: Ruído com distribuição normal

Ao utilizar a função `randn` garantimos que os valores apresentados serão os mais próximos de um sinal aleatório possível criando este aspecto de ruído ao sinal recém criado.

Para verificar se de fato o ruído encontra-se equilibrado, podemos calcular seu histograma (conforme apresentado abaixo) e validar se de fato a curva apresenta assemelha-se a de uma gaussiana (o que significa que a distribuição de probabilidades está o mais equilibrada possível).

FONTE: Elaborado Pelo Autor

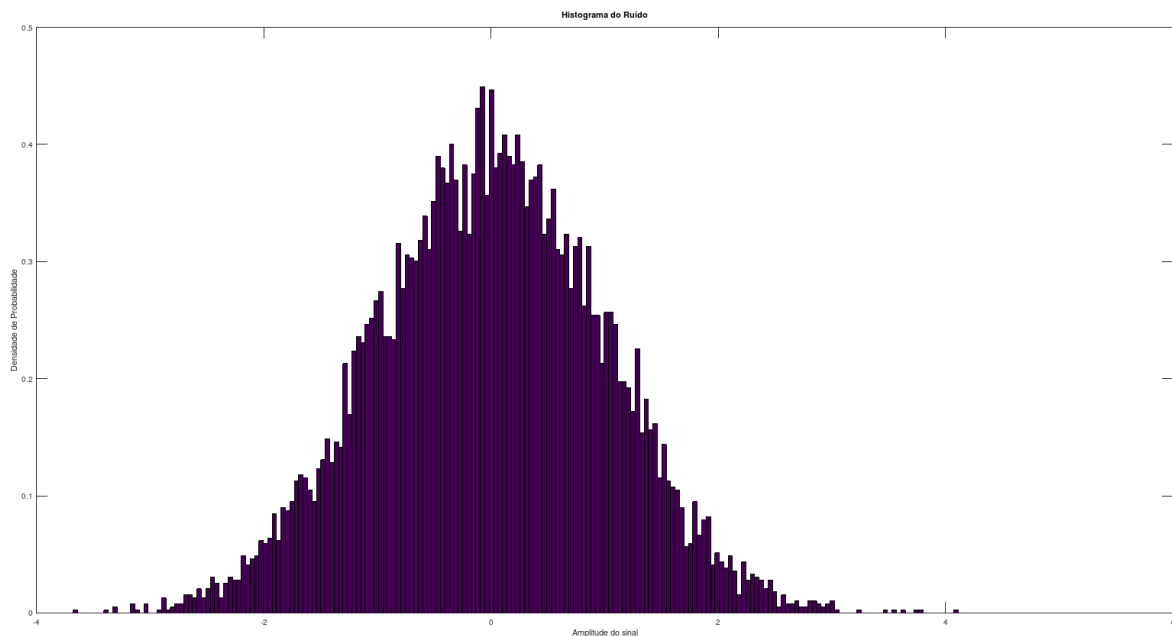


Figura 9: Histograma do ruído - Distribuição Gaussiana

Em seguida, após garantir que o ruído de fato está equilibrado, ou seja, não está alocando mais potência para uma determinada frequência / instante no tempo, podemos verificar seu comportamento no domínio da frequência.

Novamente, como o ruído não possui nenhuma componente periódica com potência mais alta que as demais (na realidade não há periodicidade neste sinal), nota-se que no domínio da frequência o formato do sinal se assemelha muito ao sinal no domínio do tempo, pois não há uma faixa de frequência com amplitude maior ou menor.

FONTE: Elaborado Pelo Autor

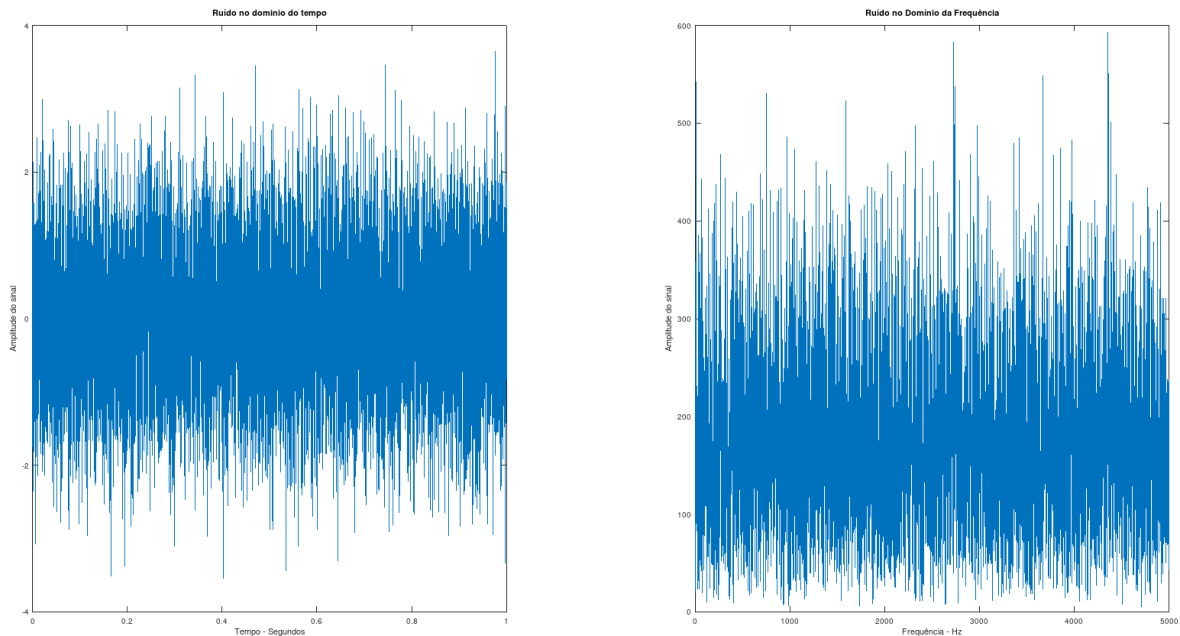


Figura 10: Plot do ruído no tempo e frequência

Para comprovar que não há uma componente periódica no sinal, podemos aplicar a função de autocorrelação no sinal para verificar se há ou não periodicidade em alguma faixa do sinal, note que como demonstrado abaixo, existe apenas um pico sobre o eixo, apontando que não há periodicidade sobre componentes alternadas do sinal.

FONTE: Elaborado Pelo Autor

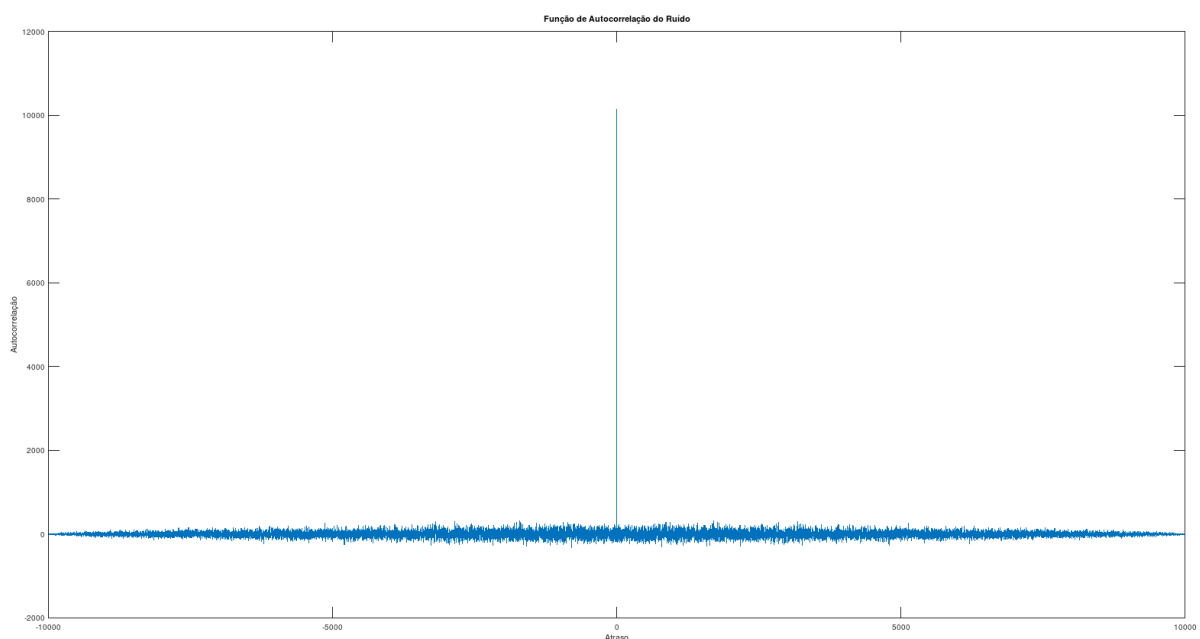


Figura 11: Função autocorrelação do ruído

Agora, vamos aplicar um filtro passa baixa no sinal aleatório para verificar sua resposta no domínio do tempo e também da frequência, para isso, um filtro PB de 1000Hz foi montado, com a seguinte resposta em frequência (ordem 50).

FONTE: Elaborado Pelo Autor

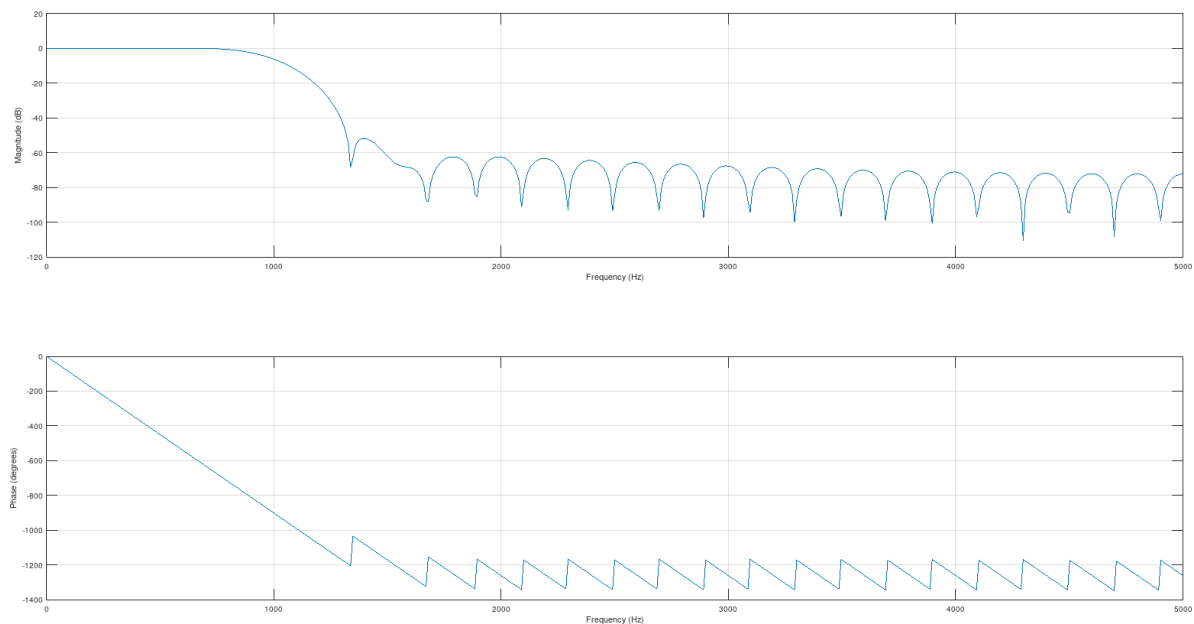


Figura 12: Resposta em frequência do filtro

O filtro foi multiplicado com o sinal original, e após sua multiplicação, podemos notar que o sinal ainda continua bastante aleatório, mas agora, sem uma boa parte de seu preenchimento (em azul), isso ocorre pois as frequências mais altas, correspondentes as maiores variações e portanto ao preenchimento do gráfico foram cortadas, desta forma, o gráfico aparece mais esbranquiçado.

FONTE: Elaborado Pelo Autor

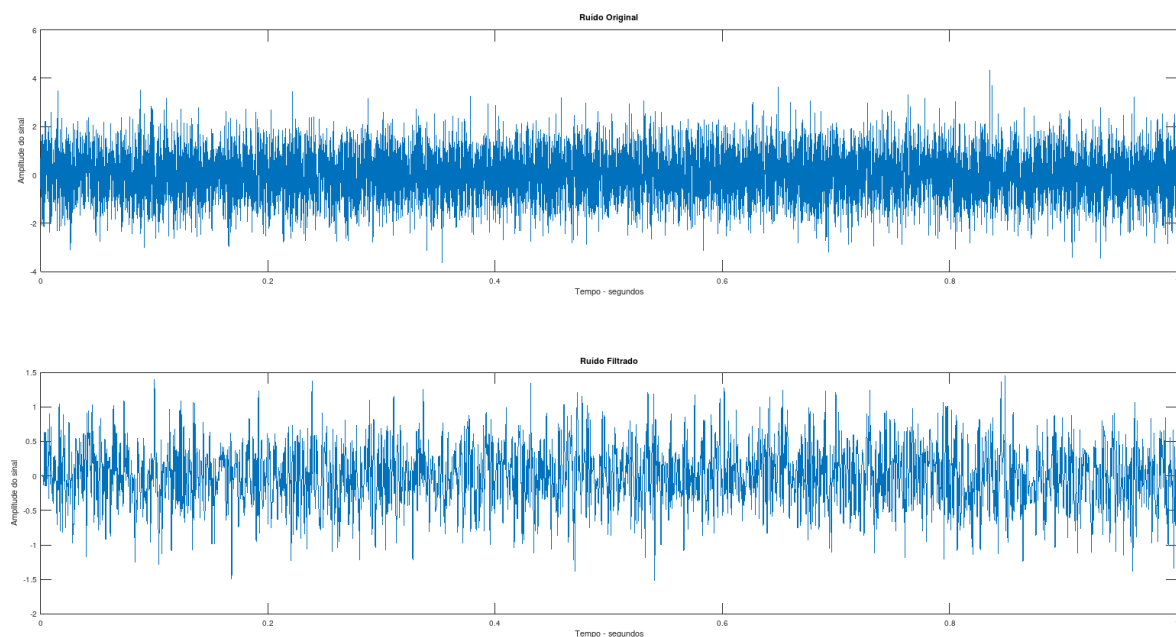


Figura 13: Ruído filtrado em $F < 1000\text{Hz}$

Por fim, para verificar se o filtro foi aplicado corretamente, o sinal filtrado foi plotado no domínio

do tempo e ao lado no domínio da frequência (imagem abaixo). Note que quase que imediatamente após 1000Hz toda a amplitude do sinal cai para zero.

FONTE: Elaborado Pelo Autor

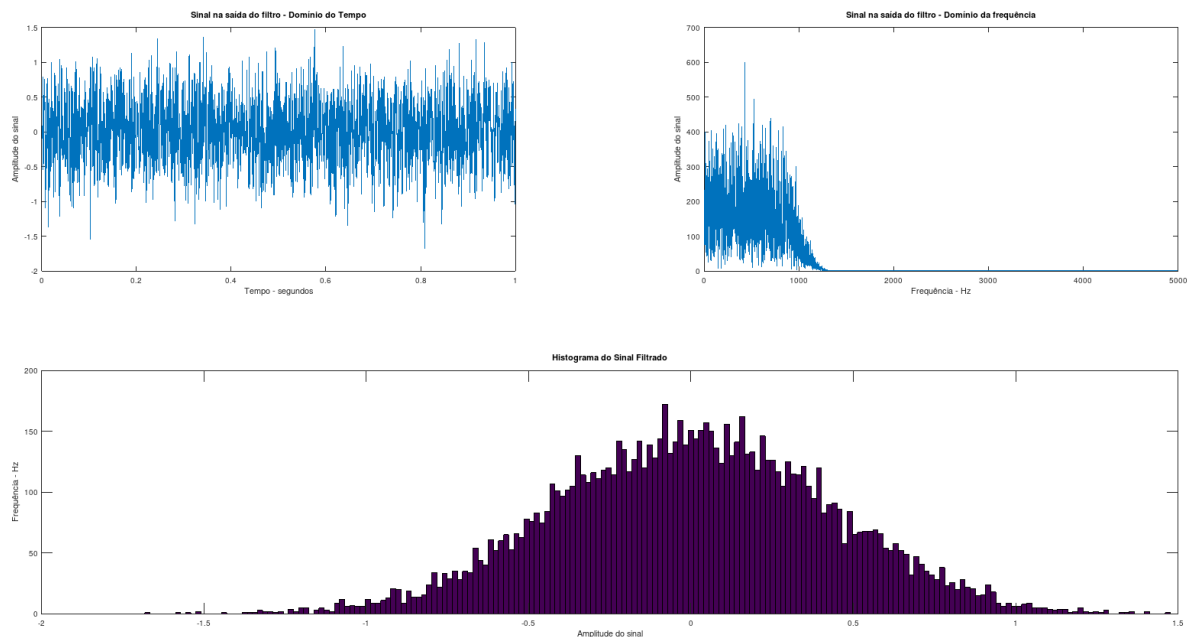


Figura 14: Ruído filtrado em $F < 1000\text{Hz}$

Para verificar se de fato o sinal ainda continua equilibrado após a filtragem, foi novamente plotada a gaussiana, agora com o sinal filtrado em 1000Hz, note que mesmo após o filtro a distribuição ainda continua apontando equilíbrio, isto ocorre pois o sinal mesmo abaixo de 1000Hz ainda estava equilibrado então após a filtragem, a distribuição normal permanece.

4 Conclusão

Com base nos conceitos discutidos e nos resultados obtidos com a modulação AM-DSB-SC e AM-DSB-TC, podemos concluir que o processamento de sinais em sistemas de modulação por amplitude (AM) é fundamental para a transmissão de informações em comunicações analógicas.

A modulação AM-DSB-SC, ao modular a amplitude do sinal portador com a informação, permite a transmissão eficiente de dados. Já a AM-DSB-TC, ao suprimir a portadora, reduz a largura de banda necessária para transmitir o sinal modulado, tornando-a uma técnica mais eficiente em termos de espectro.

Após a transmissão, os processos de recepção, demodulação e interpretação são cruciais para recuperar a informação original com precisão. A compreensão desses processos é essencial para o correto funcionamento dos sistemas de comunicação que utilizam modulação AM.

5 Scripts Utilizados:

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

```
1 clc; clear all; close all
2
3 % amplitude de cada componente do sinal;
4 A1 = 6;
```

```

5 A2 = 2;
6 A3 = 4;
7
8 % definindo frequências das componentes do sinal;
9 f1 = 1000;
10 f2 = 3000;
11 f3 = 5000;
12
13 % definindo periodo e frequência de amostragem;
14 fs = 50*f2;
15 Ts = 1/fs;
16 T = 1/f1;
17
18 % definindo priodos do sinal;
19 t_inicial = 0;
20 t_final = 2;
21
22 % vetor "t" correspondente ao periodo de análise (dominio da frequência);
23 t = [t_inicial:Ts:t_final];
24
25 % calculando o passo no dominio da frequência;
26 passo_f = 1/t_final;
27
28 % vetor "f" correspondente ao periodo de análise (dominio da frequência);
29 f = [-fs/2:passo_f:fs/2];
30
31
32 % definição das funções de cada componente do sinal;
33 x1_t = A1*cos(2*pi*f1*t);
34 x2_t = A2*cos(2*pi*f2*t);
35 x3_t = A3*cos(2*pi*f3*t);
36
37 % adicionando uma componente DC ao sinal;
38 % x4_t = 2;
39
40 % somatório das componentes para compor o sinal completo
41 x_t = x1_t + x2_t + x3_t;
42
43 % realizando a FFT (fast Fourier Transform) do sinal para o domínio da frequência;
44 X_f = fft(x_t)/length(x_t);
45 X_f = fftshift(X_f);
46
47 X1_f = fft(x1_t)/length(x1_t);
48 X1_f = fftshift(X1_f);
49
50 X2_f = fft(x2_t)/length(x2_t);
51 X2_f = fftshift(X2_f);
52
53 X3_f = fft(x3_t)/length(x3_t);
54 X3_f = fftshift(X3_f);

```

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

```

1 % Plotagem do sinal x1_t no domínio do tempo;
2 figure(1)
3 subplot(221)
4 plot(t,x1_t)
5 xlim([0 3*T])
6

```

```

7 % Plotagem do sinal x2_t no domínio do tempo;
8 subplot(222)
9 plot(t,x2_t)
10 xlim([0 3*T])
11
12 % Plotagem do sinal x3_t no domínio do tempo;
13 subplot(223)
14 plot(t,x3_t)
15 xlim([0 3*T])
16
17 % Plotagem do sinal resultante no domínio do tempo;
18 subplot(224)
19 plot(t,x_t)
20 xlim([0 3*T])
21
22 % Plotagem do sinal x1_t no domínio da frequência;
23 figure(2)
24 subplot(221)
25 plot(f,abs(X1_f))
26 grid on
27
28 % Plotagem do sinal x1_t no domínio da frequência;
29 subplot(222)
30 plot(f,abs(X2_f))
31 grid on
32
33 % Plotagem do sinal x1_t no domínio da frequência;
34 subplot(223)
35 plot(f,abs(X3_f))
36 grid on
37
38 % Plotagem do sinal x1_t no domínio da frequência;
39 subplot(224)
40 plot(f,abs(X_f))
41 grid on

```

5.3 Utilizando a função 'norm', determine a potência média do sinal 's'.

```

1 % Calculando a potência média total do sinal (metodo 1):
2 potencia_media = norm(x_t).^2 / length(x_t)
3
4 % Calculando a potência AC e DC do sinal;
5 Pot_AC = var(x_t)
6 Pot_DC = mean(x_t)^2
7
8 % Calculando a potência média total do sinal (metodo 2):
9 Pot_Media_Total = Pot_AC + Pot_DC
10
11 % Calculando a potência média total do sinal (metodo 3):
12 Pot_Media_Total = 1/length(x_t)*sum(x_t.^2)

```

5.4 Utilizando a função 'pwelch', plote a Densidade Espectral de Potência do sinal 's'.

```

1 % realizando plot de densidade espectral utilizando "Pwelch":
2 figure(4)
3 plot(pwelch(x_t));
4 xlim([0 5000])

```

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

```
1  clc; clear all; close all
2
3  % amplitude de cada componente do sinal;
4  A1 = 5;
5  A2 = 5/3;
6  A3 = 1;
7
8  % definindo frequências das componentes do sinal;
9  f1 = 1000;
10 f2 = 3000;
11 f3 = 5000;
12
13 % definindo periodo e frequência de amostragem;
14 fs = 50*f3;
15 Ts = 1/fs;
16 T = 1/f1;
17
18 % definindo priodos do sinal;
19 t_inicial = 0;
20 t_final = 2;
21
22 % vetor "t" correspondente ao periodo de análise (dominio da frequência);
23 t = [t_inicial:Ts:t_final];
24
25 % calculando o passo no dominio da frequência;
26 passo_f = 1/t_final;
27
28 % vetor "f" correspondente ao periodo de análise (dominio da frequência);
29 f = [-fs/2:passo_f:fs/2];
30
31
32 % definição das funções de cada componente do sinal;
33 x1_t = A1*cos(2*pi*f1*t);
34 x2_t = A2*cos(2*pi*f2*t);
35 x3_t = A3*cos(2*pi*f3*t);
36
37 % adicionando uma componente DC ao sinal;
38 % x4_t = 2;
39
40 % somatório das componentes para compor o sinal completo
41 x_t = x1_t + x2_t + x3_t;
42
43 % realizando a FFT (fast Fourier Transform) do sinal para o domínio da frequência;
44 X_f = fft(x_t)/length(x_t);
45 X_f = fftshift(X_f);
46
47 X1_f = fft(x1_t)/length(x1_t);
48 X1_f = fftshift(X1_f);
49
50 X2_f = fft(x2_t)/length(x2_t);
51 X2_f = fftshift(X2_f);
52
53 X3_f = fft(x3_t)/length(x3_t);
54 X3_f = fftshift(X3_f);
```

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

```
1 % Plotagem do sinal x1_t no domínio do tempo;
2 figure(1)
3 subplot(221)
4 plot(t,x1_t)
5 xlim([0 3*T])
6
7 % Plotagem do sinal x2_t no domínio do tempo;
8 subplot(222)
9 plot(t,x2_t)
10 xlim([0 3*T])
11
12 % Plotagem do sinal x3_t no domínio do tempo;
13 subplot(223)
14 plot(t,x3_t)
15 xlim([0 3*T])
16
17 % Plotagem do sinal resultante no domínio do tempo;
18 subplot(224)
19 plot(t,x_t)
20 xlim([0 3*T])
21
22 % Plotagem do sinal x1_t no domínio da frequência;
23 figure(2)
24 subplot(221)
25 plot(f,abs(X1_f))
26 grid on
27
28 % Plotagem do sinal x1_t no domínio da frequência;
29 subplot(222)
30 plot(f,abs(X2_f))
31 grid on
32
33 % Plotagem do sinal x1_t no domínio da frequência;
34 subplot(223)
35 plot(f,abs(X3_f))
36 grid on
37
38 % Plotagem do sinal x1_t no domínio da frequência;
39 subplot(224)
40 plot(f,abs(X_f))
41 grid on
```

```
1 clc; clear all; close all
2
3 % amplitude de cada componente do sinal;
4 A1 = 5;
5 A2 = 5/3;
6 A3 = 1;
7
8 % definindo frequências das componentes do sinal;
9 f1 = 1000;
10 f2 = 3000;
11 f3 = 5000;
12
13 % definindo periodo e frequência de amostragem;
14 fs = 50*f3;
15 Ts = 1/fs;
16 T = 1/f1;
```



```

17
18 % definindo priodos do sinal;
19 t_inicial = 0;
20 t_final = 2;
21
22 % vetor "t" correspondente ao periodo de análise (dominio da frequência);
23 t = [t_inicial:Ts:t_final];
24
25 % calculando o passo no dominio da frequência;
26 passo_f = 1/t_final;
27
28 % vetor "f" correspondente ao periodo de análise (dominio da frequência);
29 f = [-fs/2:passo_f:fs/2];
30
31
32 % definição das funções de cada componente do sinal;
33 x1_t = A1*cos(2*pi*f1*t);
34 x2_t = A2*cos(2*pi*f2*t);
35 x3_t = A3*cos(2*pi*f3*t);
36
37 % adicionando uma componente DC ao sinal;
38 % x4_t = 2;
39
40 % somatório das componentes para compor o sinal completo
41 x_t = x1_t + x2_t + x3_t;
42
43 % realizando a FFT (fast Fourier Transform) do sinal para o domínio da frequência;
44 X_f = fft(x_t)/length(x_t);
45 X_f = fftshift(X_f);
46
47 X1_f = fft(x1_t)/length(x1_t);
48 X1_f = fftshift(X1_f);
49
50 X2_f = fft(x2_t)/length(x2_t);
51 X2_f = fftshift(X2_f);
52
53 X3_f = fft(x3_t)/length(x3_t);
54 X3_f = fftshift(X3_f);

```

5.7 Gerar 3 filtros ideais e plotar a resposta em frequencia para os filtros:

```

1
2 filtro_PB = [zeros(1,246e3) ones(1,8001) zeros(1,246e3)];
3 filtro_PA = [ones(1,242e3) zeros(1,16001) ones(1,242e3)];
4 filtro_PF = [zeros(1,242e3) ones(1,4000) zeros(1,8001) ones(1,4000) zeros(1,242e3)];
5
6 % Plotagem do sinal PF no domínio da frequência;
7 figure(6)
8 subplot(311)
9 plot(f,filtro_PB)
10 grid on
11 xlim([-8000 8000])
12
13 subplot(312)
14 plot(f,filtro_PA)
15 grid on
16 xlim([-8000 8000])
17
18 subplot(313)
19 plot(f,filtro_PF)
20 grid on

```

```
21 xlim([-8000 8000])
```

5.8 Plotar o sinal 's' t através dos filtros e plotar as saídas no domínio do tempo e no domínio da frequência.

```
1
2 %Sinais filtrados:
3 X_filtradoA = X_f .* filtro_PA;
4 X_filtradoB = X_f .* filtro_PB;
5 X_filtradoF = X_f .* filtro_PF;
6
7 % Plotagem do sinal PF no domínio da frequência;
8 figure(7)
9 subplot(321)
10 plot(f,X_filtradoA)
11 grid on
12 ylim([0 1.2])
13 xlim([-8000 8000])
14
15 subplot(323)
16 plot(f,X_filtradoB)
17 grid on
18 ylim([0 1.2])
19 xlim([-8000 8000])
20
21 subplot(325)
22 plot(f,X_filtradoF)
23 grid on
24 ylim([0 1.2])
25 xlim([-8000 8000])
26
27 x_filtradoA = ifft(X_filtradoA);
28 x_filtradoB = ifft(X_filtradoB);
29 x_filtradoF = ifft(X_filtradoF);
30
31 subplot(322)
32 plot(t,x_filtradoA)
33 xlim([0 3*T])
34
35 subplot(324)
36 plot(t,x_filtradoB)
37 xlim([0 3*T])
38
39 subplot(326)
40 plot(t,x_filtradoF)
41 xlim([0 3*T])
```

5.9 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.

```
1 close all; clc; clear all;
2
3 % Frequência de amostragem (10k amostras por segundo)
4 Fs = 10000;
5
6 % Vetor de tempo de 0 a 1 segundo com passo de 1/Fs
```

```

7 t = 0:1/Fs:1;
8
9 % Gerar o ruído com distribuição normal
10 ruído = randn(size(t));
11
12 % Plotar o ruído
13 plot(t, ruído);
14 xlabel('Tempo - Segundos');
15 ylabel('Amplitude do sinal');
16 title('Ruído com distribuição normal');

```

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

```

1 % Não foi possível utilizar a função histogram diretamente pois durante a realização deste
  trabalho utilizei o octave, e a função histogram não estava disponível, utilizei o seguinte c
  ódigo para resolver a questão:
2
3 % Definir o número de bins para o histograma
4 num_bins = 200;
5
6 % Plotar o histograma do ruído
7 [counts, edges] = hist(ruído, num_bins);
8 bin_width = edges(2) - edges(1);
9 bar(edges, counts / (sum(counts) * bin_width), 'hist');
10 xlabel('Amplitude do sinal');
11 ylabel('Densidade de Probabilidade');
12 title('Histograma do Ruído');

```

5.11 Plotar o ruído no domínio do tempo e da frequência

```

1 % Plotar o ruído
2 figure(1)
3 subplot(1, 2, 1);
4 plot(t, ruído);
5 xlabel('Tempo - Segundos');
6 ylabel('Amplitude do sinal');
7 title('Ruído no domínio do tempo');
8
9 % Calcular a FFT do ruído
10 ruído_F = fft(ruído);
11
12 % Calcular o vetor de frequências correspondente à FFT
13 f = Fs*(0:(length(t)/2))/length(t);
14
15 % Plotar a magnitude da FFT (apenas metade, pois é simétrica)
16 subplot(1, 2, 2);
17 plot(f, 2*abs(ruído_F(1:length(t)/2+1)));
18 xlabel('Frequência - Hz');
19 ylabel('Amplitude do sinal');
20 title('Ruído no Domínio da Frequência');

```

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

```

1 % Calcular a função de autocorrelação do ruído
2 [correlation, lag] = xcorr(ruido);
3
4 figure(2)
5 % Plotar a função de autocorrelação
6 plot(lag, correlation);
7 xlabel('Atraso');
8 ylabel('Autocorrelação');
9 title('Função de Autocorrelação do Ruído');

```

5.13 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'.

```

1 % Definir a frequência de corte do filtro
2 Fc = 1000; % Frequência de corte em Hz
3
4 % Definir a ordem do filtro
5 ordem = 50;
6
7 % Projetar o filtro FIR passa-baixa
8 filtro = fir1(ordem, (Fc*2)/Fs);
9
10 % Aplicar o filtro ao ruído
11 sinal_filtrado = filter(filtro, 1, ruido);
12
13 % Plotar a resposta em frequência do filtro
14 figure(3);
15 freqz(filtro, 1, 512, Fs);
16
17 % Plotar o sinal original e o sinal filtrado
18 figure(4);
19 subplot(2,1,1);
20 plot(t, ruido);
21 title('Ruído Original');
22 xlabel('Tempo - segundos');
23 ylabel('Amplitude do sinal');
24
25 subplot(2,1,2);
26 plot(t, sinal_filtrado);
27 title('Ruído Filtrado');
28 xlabel('Tempo - segundos');
29 ylabel('Amplitude do sinal');

```

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

```

1 % Plotar a saída do filtro no domínio do tempo
2 subplot(2, 2, 1);
3 plot(t, sinal_filtrado);
4 title('Sinal na saída do filtro - Domínio do Tempo');
5 xlabel('Tempo - segundos');
6 ylabel('Amplitude do sinal');
7
8 % Calcular a FFT da saída do filtro
9 L = length(sinal_filtrado);
10 Y_filtrado = fft(sinal_filtrado);

```

```

11
12 % Calcular o vetor de frequências correspondente à FFT
13 f = Fs*(0:(L/2))/L;
14
15 % Plotar a magnitude da FFT (apenas metade, pois é simétrica)
16 subplot(2, 2, 2);
17 plot(f, 2*abs(Y_filtrado(1:L/2+1)));
18 title('Sinal na saída do filtro - Domínio da frequência');
19 xlabel('Frequência - Hz');
20 ylabel('Amplitude do sinal');
21
22 % Plotar o histograma do sinal filtrado
23 subplot(2, 2, [3, 4]);
24 hist(sinal_filtrado, num_bins);
25 title('Histograma do Sinal Filtrado');
26 xlabel('Amplitude do sinal');
27 ylabel('Frequência - Hz');

```