



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Projeto de Filtros Por Janelamento

Processamento de Sinais Digitais

Curso: Engenharia de Telecomunicações

Disciplina: PSD129005 - Processamento de Sinais Digitais

Professora: Elen Macedo Lobato

Aluna

Luiza Kuze Gomes

2 de julho de 2025

Sumário

| | | |
|----------|---------------------|-----------|
| 1 | Introdução | 2 |
| 2 | Questões | 2 |
| 2.1 | Questão 1 | 2 |
| 2.2 | Questão 2 | 4 |
| 2.3 | Questão 3 | 9 |
| 2.4 | Questão 4 | 14 |
| 3 | Conclusão | 17 |

1 Introdução

No processamento de sinais, o projeto de filtros digitais é crucial para a gestão de faixas de frequência, permitindo sua seleção, atenuação ou eliminação. A modelagem de filtros FIR (Finite Impulse Response) com o uso de janelas é uma abordagem central, e entender como cada tipo de janela influencia o desempenho do filtro é fundamental. Este relatório aborda a resolução dos exercícios propostos em aula, com simulações em MATLAB e suas análises.

2 Questões

2.1 Questão 1

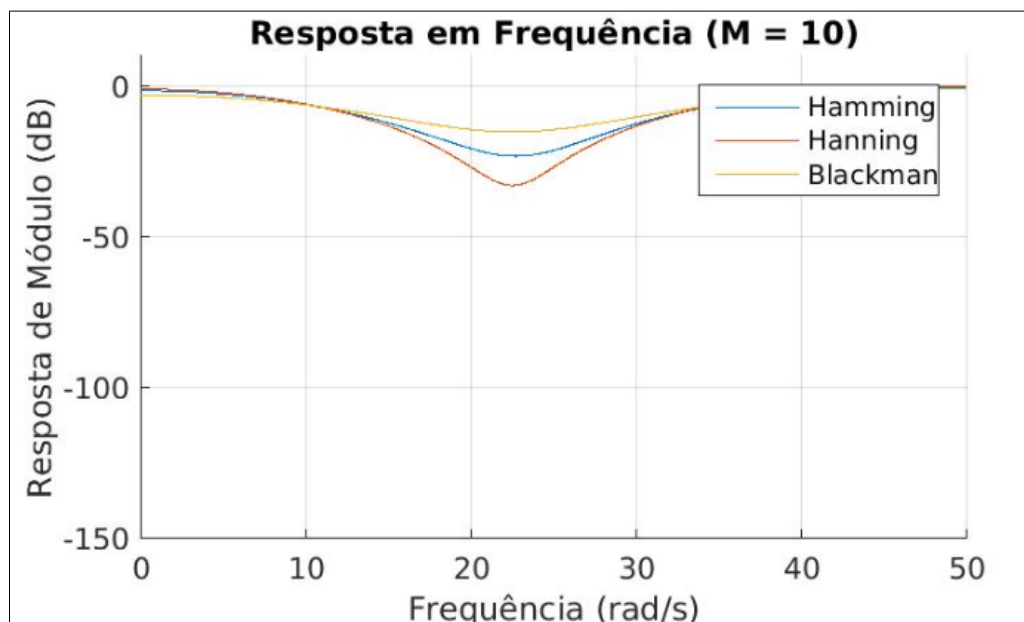
Projetar um filtro passa-faixa utilizando as janelas de Hamming, Hanning e Blackman.

- $M = 10, 100, 1000$
- $\Omega_{c1} = 10 \text{ rad/s}$
- $\Omega_{c2} = 35 \text{ rad/s}$
- $\Omega_s = 100 \text{ rad/s}$

Resolução

Para $M = 10$, a resposta em frequência mostra transições pouco definidas entre as bandas de passagem e de rejeição. A atenuação fora da banda desejada é limitada, e a resposta ainda se afasta do ideal, com variações no ganho dentro da banda de passagem.

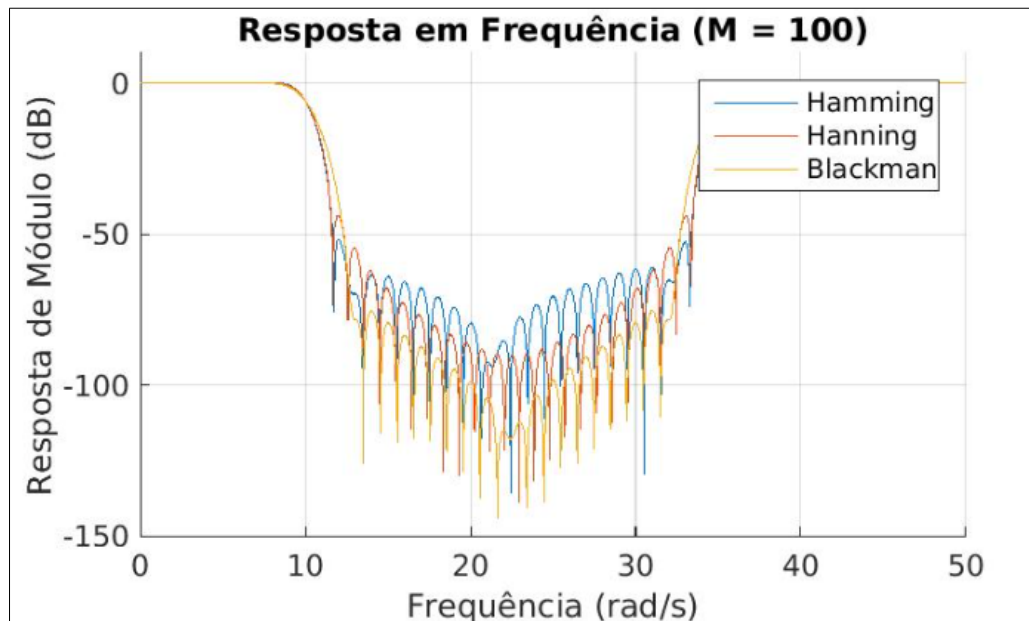
Figura 1: Resposta em Frequência com $M = 10$ para janelas Hamming, Hanning e Blackman



Fonte: Própria Autora

Com o aumento da ordem para $M = 100$, a separação entre as faixas fica mais evidente. A banda de passagem exibe menor variação no ganho e a atenuação na banda de rejeição melhora, indicando um desempenho mais próximo de um filtro ideal.

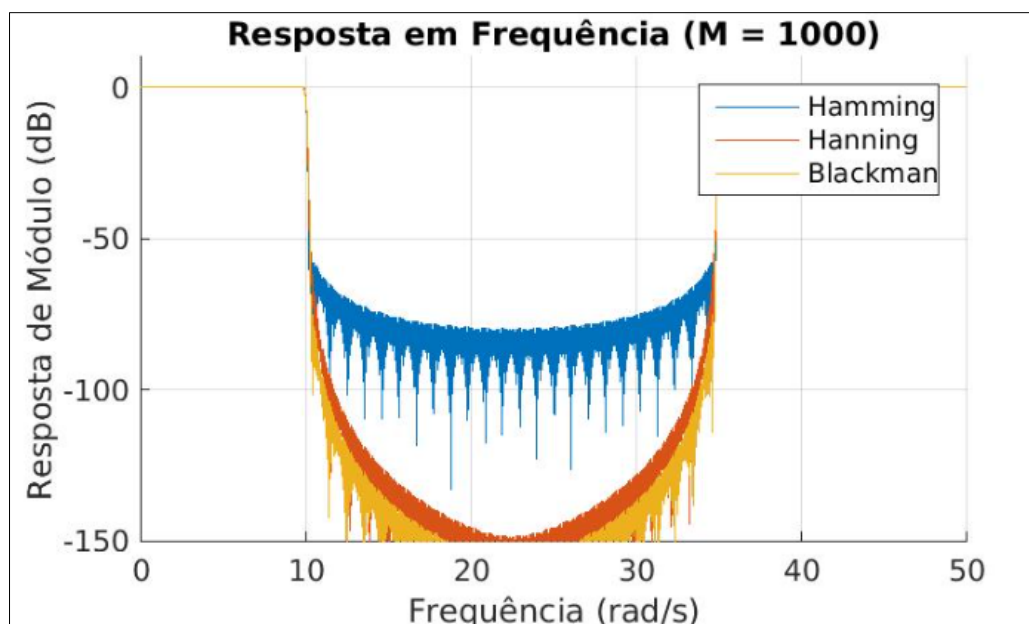
Figura 2: Resposta em Frequência com $M = 100$ para janelas Hamming, Hanning e Blackman



Fonte: Própria Autora

Para $M = 1000$, a resposta em frequência apresenta uma definição ainda maior. A banda de rejeição fica mais pronunciada, com uma atenuação superior, e a banda de passagem se torna mais plana. O gráfico, nesse ponto, aproxima-se significativamente da resposta ideal esperada para um filtro passa-faixa.

Figura 3: Resposta em Frequência com $M = 1000$ para janelas Hamming, Hanning e Blackman



Fonte: Própria Autora

Conforme ilustrado nas Figuras 1, 2 e 3, o aumento da ordem do filtro resulta em maior seletividade e em uma aproximação mais fiel da resposta ideal. As transições tornam-se mais abruptas e a rejeição fora da banda de passagem mais eficaz.

Simulação no MATLAB

```

1  Omega_c1 = 10;
2  Omega_c2 = 35;
3  Omega_s = 100;
4
5  wc1 = Omega_c1 * 2 * pi / Omega_s;
6  wc2 = Omega_c2 * 2 * pi / Omega_s;
7
8  M_values = [10, 100, 1000];
9
10 for i = 1:length(M_values)
11     M = M_values(i);
12     n = 1:M/2;
13
14     % Filtro ideal
15     h0 = 1 - (wc2 - wc1) / pi;
16     haux = (sin(wc1.*n) - sin(wc2.*n)) ./ (pi.*n);
17     h_ideal = [fliplr(haux), h0, haux];
18
19     % Lista de janelas e funções associadas
20     windows = {
21         'Hamming', @hamming;
22         'Hanning', @hanning;
23         'Blackman', @blackman
24     };
25
26     % Preparar figura
27     figure; hold on; grid on;
28     title(['Resposta em Frequência (M = ', num2str(M), ')']);
29     ylabel('Resposta de Módulo (dB)');
30     xlabel('Frequência (rad/s)');
31     axis([0 Omega_s/2 -150 10]);
32
33     % Plot de cada janela
34     for j = 1:size(windows, 1)
35         win_name = windows{j, 1};
36         win_func = windows{j, 2};
37         w = win_func(M + 1)';
38         h = h_ideal .* w;
39         [H, w_freq] = freqz(h, 1, 2048, Omega_s);
40         plot(w_freq, 20*log10(abs(H)), 'DisplayName', win_name);
41     end
42
43     legend show;
44 end

```

2.2 Questão 2

Projetar filtros digitais utilizando a janela de Kaiser.

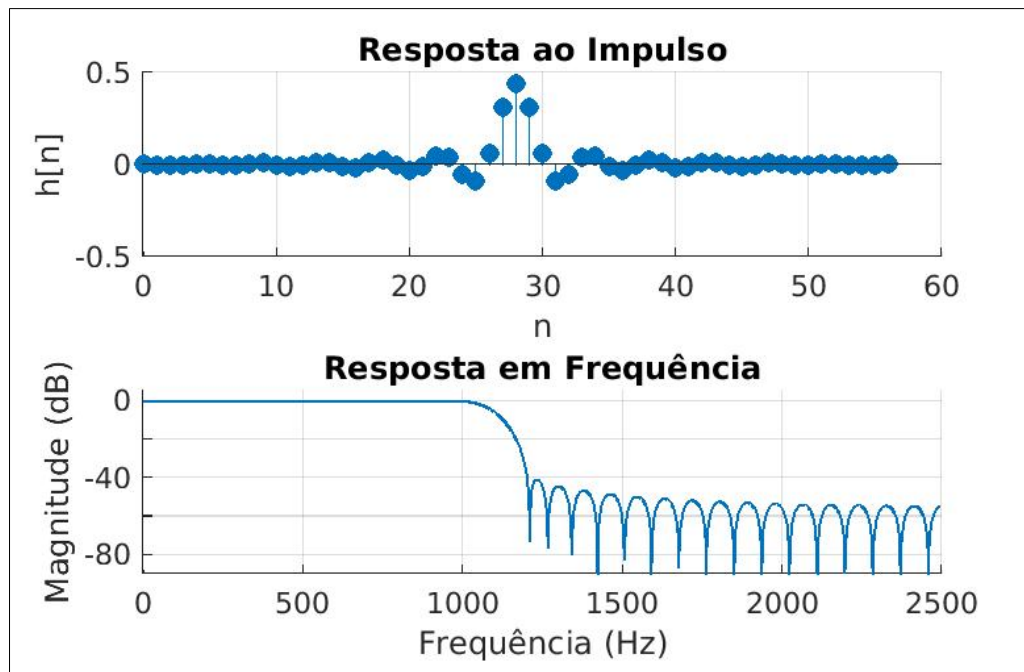
Primeiro Filtro

- $A_p = 1.0$ dB
- $A_r = 40$ dB
- $\Omega_p = 1000$ rad/s
- $\Omega_r = 1200$ rad/s
- $\Omega_s = 5000$ rad/s

Resolução

Na Figura 4, observa-se que o filtro passa-baixas exibe uma atenuação gradual a partir da frequência de corte $\Omega_p = 1000$ rad/s, alcançando a rejeição especificada em $\Omega_r = 1200$ rad/s. A banda de passagem mostra-se relativamente plana, com ripple dentro do limite de 1 dB, e a banda de rejeição demonstra um desempenho coerente com os requisitos de 40 dB de atenuação.

Figura 4: Resposta do Filtro Passa-Baixas projetado com janela de Kaiser



Fonte: Própria Autora

Simulação no MATLAB

```
1 %% Especificações do filtro
2 Ap = 1;           % Ripple banda passante (dB)
3 Ar = 40;          % Atenuação mínima (dB)
4 Omega_p = 1000;   % Frequência passante (Hz)
5 Omega_r = 1200;   % Frequência de rejeição (Hz)
6 Omega_s = 5000;   % Frequência de amostragem (Hz)
7
8 F = [Omega_p Omega_r]; % Frequências de transição
9 A = [1 0];          % Magnitudes desejadas
10
11 % Conversão dB (linear)
12 delta_p = (10^(0.05 * Ap) - 1)/(10^(0.05 * Ap) + 1);
13 delta_r = 10^(-0.05 * Ar);
14 ripples = [delta_p delta_r];
15
16 % Projeto com janela de Kaiser
17 [M, Wn, beta, tipo] = kaiserord(F, A, ripples, Omega_s);
18 janela = kaiser(M+1, beta);
19 h = fir1(M, Wn, tipo, janela, 'noscale');
20
21 % Resposta em frequência
22 [H, w] = freqz(h, 1, 2048, Omega_s);
23 H_dB = 20*log10(abs(H));
24 H_dB(H_dB < -100) = -100; % evita -Inf
```

```

25
26 %% Plots
27 figure('Name','Filtro FIR com Kaiser','NumberTitle','off');
28
29 % Resposta ao impulso
30 subplot(2,1,1);
31 stem(0:M, h, 'filled');
32 title('Resposta ao Impulso');
33 xlabel('n'); ylabel('h[n]');
34 grid on;
35 box off;
36
37 % Resposta em frequência
38 subplot(2,1,2);
39 plot(w, H_dB, 'LineWidth', 1.2);
40 xlabel('Frequência (Hz)');
41 ylabel('Magnitudo (dB)');
42 title('Resposta em Frequência');
43 axis([0 Omega_s/2 -90 5]);
44 grid on;
45 box off;

```

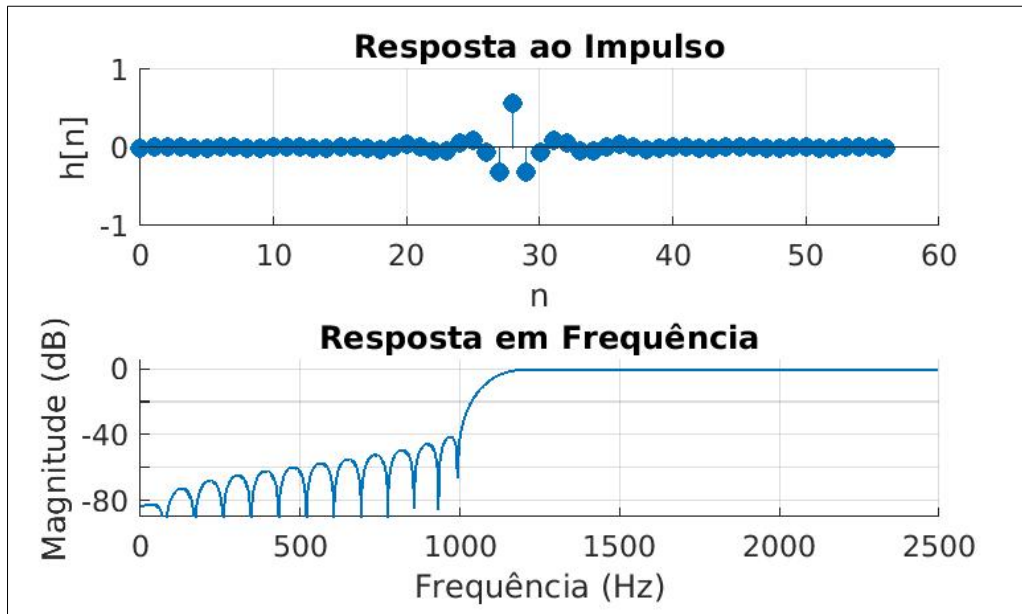
Segundo Filtro

- $A_p = 1.0$ dB
- $A_r = 40$ dB
- $\Omega_r = 1000$ rad/s
- $\Omega_p = 1200$ rad/s
- $\Omega_s = 5000$ rad/s

Resolução

A Figura 5 mostra o comportamento do filtro passa-altas. A banda de rejeição atinge rapidamente a atenuação de 40 dB exigida até $\Omega_r = 1000$ rad/s, enquanto a banda de passagem a partir de $\Omega_p = 1200$ rad/s é bem definida e estável. A transição entre as bandas é curta, evidenciando boa eficiência espectral do projeto com a janela de Kaiser.

Figura 5: Resposta do Filtro Passa-Altas projetado com janela de Kaiser



Fonte: Própria Autora

Simulação no MATLAB

```

1 %% Especificações do filtro
2 Ap = 1;           % Ripple na banda passante (dB)
3 Ar = 40;          % Atenuação mínima (dB)
4 Omega_p = 1200;   % Frequência passante (Hz)
5 Omega_r = 1000;   % Frequência de rejeição (Hz)
6 Omega_s = 5000;   % Frequência de amostragem (Hz)
7
8 F = [Omega_r Omega_p]; % Frequência de transição (em ordem crescente!)
9 A = [0 1];          % Resposta desejada: rejeita antes, passa depois
10
11 % Conversão dB (linear)
12 delta_p = (10^(0.05 * Ap) - 1) / (10^(0.05 * Ap) + 1);
13 delta_r = 10^(-0.05 * Ar);
14 ripples = [delta_r delta_p]; % [rejeição, passagem]
15
16 % Projeto com janela de Kaiser
17 [M, Wn, beta, tipo] = kaiserord(F, A, ripples, Omega_s);
18 janela = kaiser(M+1, beta);
19 h = fir1(M, Wn, tipo, janela, 'noscale');
20
21 % Resposta em frequência
22 [H, w] = freqz(h, 1, 2048, Omega_s);
23 H_dB = 20*log10(abs(H));
24 H_dB(H_dB < -100) = -100; % evita -Inf no gráfico
25
26 %% Plotagem final estilo limpo
27 figure('Name','Filtro FIR com Kaiser','NumberTitle','off');
28
29 % Resposta ao impulso
30 subplot(2,1,1);
31 stem(0:M, h, 'filled');
32 xlabel('n'); ylabel('h[n]');
33 title('Resposta ao Impulso');
34 grid on;

```



```

35 box off;
36
37 % Resposta em frequência
38 subplot(2,1,2);
39 plot(w, H_dB, 'LineWidth', 1.2);
40 xlabel('Frequência (Hz)');
41 ylabel('Magnitude (dB)');
42 title('Resposta em Frequência');
43 axis([0 Omega_s/2 -90 5]);
44 grid on;
45 box off;

```

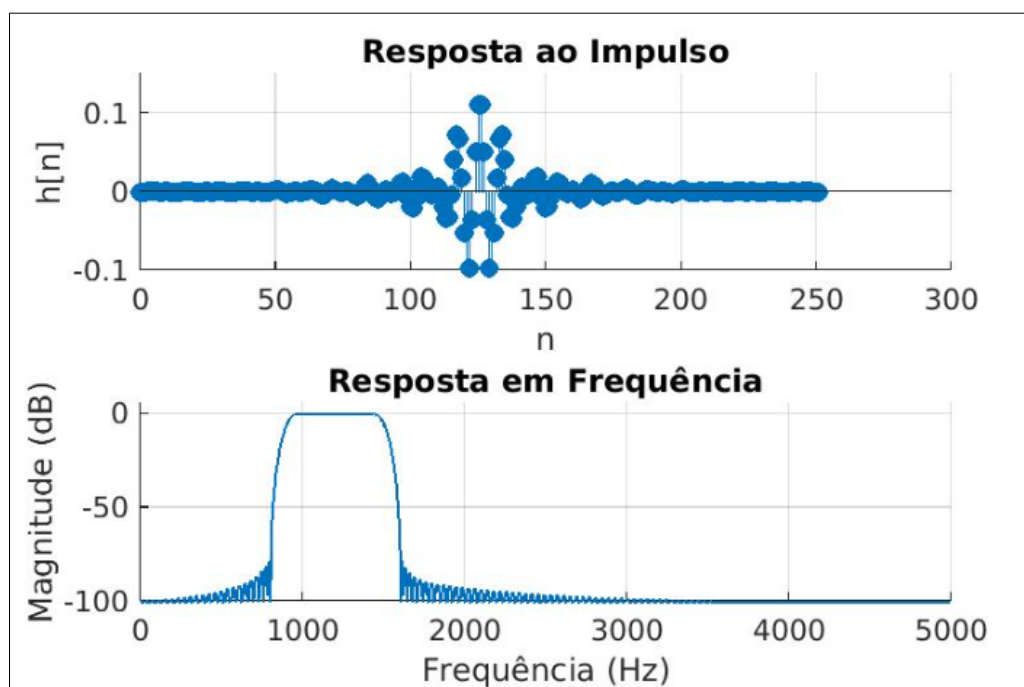
Terceiro Filtro

- $A_p = 1.0$ dB
- $A_r = 80$ dB
- $\Omega_{r1} = 800$ rad/s
- $\Omega_{p1} = 1000$ rad/s
- $\Omega_{p2} = 1400$ rad/s
- $\Omega_{r2} = 1600$ rad/s
- $\Omega_s = 10000$ rad/s

Resolução

Já na Figura 6, o filtro rejeita-faixa atende aos parâmetros do terceiro filtro, com $A_r = 80$ dB de atenuação. As bandas de passagem (Ω_{p1} e Ω_{p2}) preservam o conteúdo espectral, enquanto a faixa de rejeição entre $\Omega_{r1} = 800$ rad/s e $\Omega_{r2} = 1600$ rad/s é bem definida. O uso da janela de Kaiser foi crucial para controlar o ripple e garantir a elevada atenuação necessária.

Figura 6: Resposta do Filtro Rejeita-Faixa projetado com janela de Kaiser



Fonte: Própria Autora

Simulação no MATLAB

```
1 clear all; close all; clc;
2
3 %% Especificações do filtro
4 Ap = 1;           % Ripple na banda passante (dB)
5 Ar = 80;          % Atenuação mínima (dB)
6 Omega_p1 = 1000;  % Início da banda passante (Hz)
7 Omega_r1 = 800;   % Início da banda de rejeição (Hz)
8 Omega_r2 = 1600;  % Fim da banda de rejeição (Hz)
9 Omega_p2 = 1400;  % Fim da banda passante (Hz)
10 Omega_s = 10000; % Frequência de amostragem (Hz)
11
12 % Frequências normalizadas e amplitude desejada
13 F = [Omega_r1 Omega_p1 Omega_p2 Omega_r2];
14 A = [0 1 0];
15
16 % Conversão de dB (linear)
17 delta_p = (10^(0.05 * Ap) - 1)/(10^(0.05 * Ap) + 1);
18 delta_r = 10^(-0.05 * Ar);
19 ripples = [delta_r delta_p delta_r];
20
21 % Projeto do filtro com janela de Kaiser
22 [M, Wn, beta, tipo] = kaiserord(F, A, ripples, Omega_s);
23 janela = kaiser(M + 1, beta);
24 h = fir1(M, Wn, tipo, janela, 'noscale');
25
26 % Resposta em frequência
27 [H, w] = freqz(h, 1, 2048, Omega_s);
28 H_dB = 20*log10(abs(H));
29 H_dB(H_dB < -100) = -100; % evita -Inf visualmente
30
31 %% Plots
32 figure('Name','Filtro FIR Rejeita-Faixa (Kaiser)','NumberTitle','off');
33
34 % Resposta ao impulso
35 subplot(2,1,1);
36 stem(0:M, h, 'filled');
37 title('Resposta ao Impulso');
38 xlabel('n'); ylabel('h[n]');
39 ylim([-0.1 0.15]);
40 grid on;
41 box off;
42
43 % Resposta em frequência (em dB)
44 subplot(2,1,2);
45 plot(w, H_dB, 'LineWidth', 1.2);
46 xlabel('Frequência (Hz)');
47 ylabel('Magnitude (dB)');
48 title('Resposta em Frequência');
49 axis([0 Omega_s/2 -100 5]);
50 grid on;
51 box off;
```

2.3 Questão 3

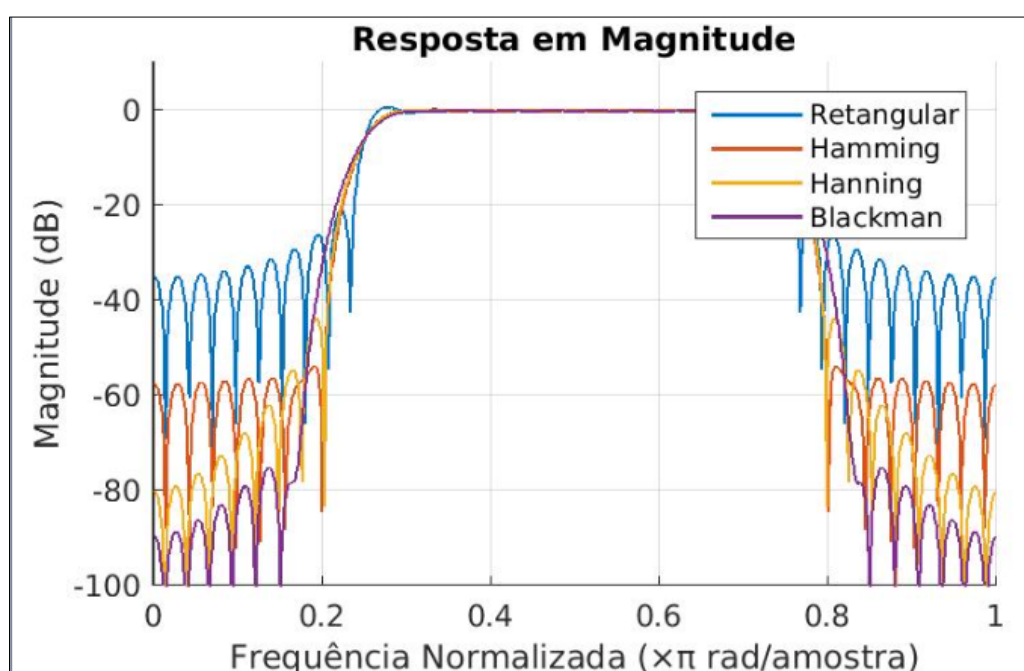
Criar um sinal composto por três senóides (50 Hz, 350 Hz e 900 Hz) e projetar filtros digitais utilizando as janelas Retangular, Hamming, Hanning e Blackman para eliminar as componentes de 50 Hz e 900 Hz, preservando a de 350 Hz.

Resolução

A análise do sinal de entrada e da resposta dos filtros projetados permite observar o comportamento de cada janela na filtragem das frequências indesejadas. A Figura 7 apresenta a resposta em magnitude dos filtros, enquanto a Figura 8 mostra a resposta em fase. Por fim, a Figura 9 compara o espectro do sinal original e o espectro após a aplicação de um dos filtros.

Com base na Figura 7, é possível perceber que todas as janelas foram eficazes em atenuar as componentes de 50 Hz e 900 Hz, mantendo a de 350 Hz, como desejado. No entanto, observam-se diferenças na atenuação fora da banda de passagem. A janela retangular apresenta lobos laterais mais elevados, o que indica menor supressão das componentes fora da banda de interesse. Já as janelas de Hamming, Hanning e Blackman promovem uma maior atenuação nesses lobos, sendo o efeito mais visível na janela de Blackman, devido à sua natureza suavizada.

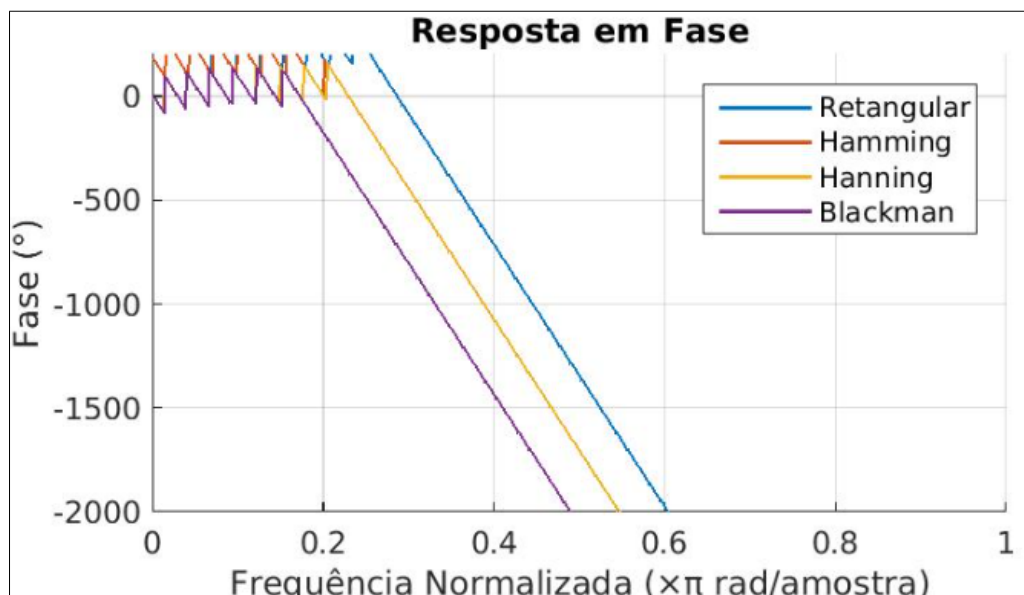
Figura 7: Resposta em Magnitude dos filtros com janelas Retangular, Hamming, Hanning e Blackman



Fonte: Própria Autora

A Figura 8 evidencia que a resposta em fase dos filtros é aproximadamente linear em todas as janelas, o que é uma característica desejável para preservar a forma do sinal na banda de passagem.

Figura 8: Resposta em Fase dos filtros com janelas Retangular, Hamming, Hanning e Blackman

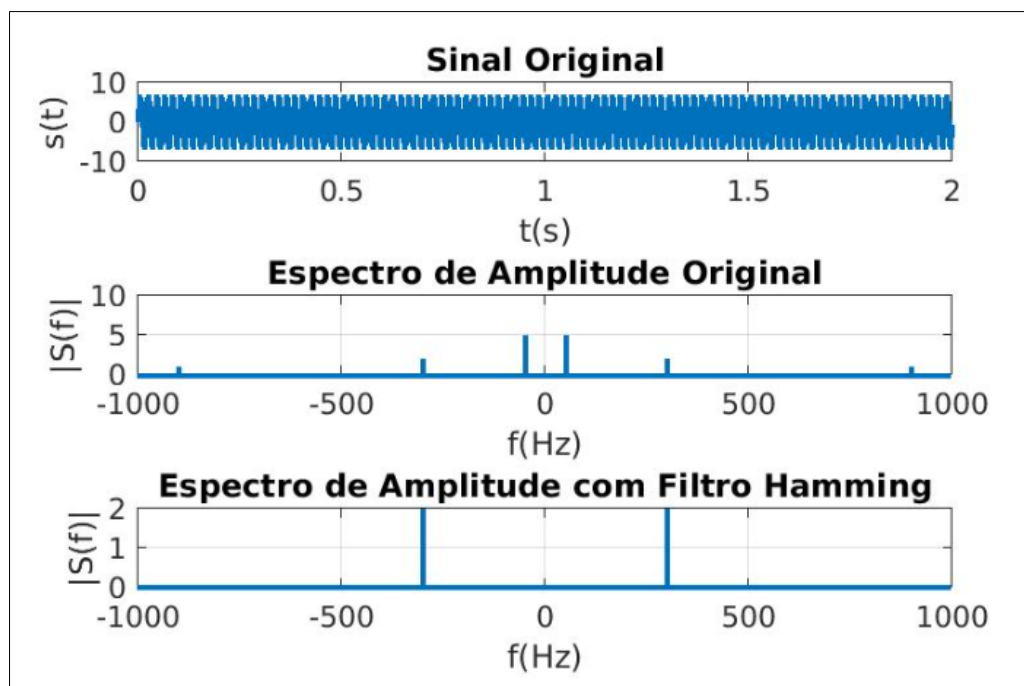


Fonte: Própria Autora

Simulação no MATLAB

Por fim, a Figura 9 demonstra claramente o efeito da filtragem no domínio da frequência: após a aplicação do filtro, as componentes indesejadas em 50 Hz e 900 Hz são eliminadas, e a frequência de 350 Hz é preservada com boa fidelidade, confirmando a eficácia da técnica de janelamento no projeto de filtros FIR.

Figura 9: Sinal original no tempo e espectros de amplitude antes e depois da filtragem



Fonte: Própria Autora

Simulação no MATLAB - Resposta em Magnitude e Fase

```

1 clear all; close all; clc;
2
3 %% Parâmetros
4 fs = 2000; % Frequência de amostragem (Hz)
5 M = 71; % Ordem do filtro FIR (número de coeficientes será M)
6 Omega_c1 = 250; % Frequência de corte inferior desejada (Hz)
7 Omega_c2 = 750; % Frequência de corte superior desejada (Hz)
8
9 % Conversão das frequências de corte para radianos/amostra (frequência angular normalizada)
10 wc1 = Omega_c1 * 2 * pi / fs; % Frequência de corte inferior normalizada (rad/amostra)
11 wc2 = Omega_c2 * 2 * pi / fs; % Frequência de corte superior normalizada (rad/amostra)
12
13 % Vetor de índices do filtro (centrado em zero), necessário para implementar h[n]
14 n = -(M-1)/2 : (M-1)/2;
15
16 %% Filtro ideal
17 h_n = zeros(1, M);
18 for i = 1:M
19     if n(i) == 0
20         h_n(i) = (wc2 - wc1)/pi;
21     else
22         h_n(i) = (sin(wc2*n(i)) - sin(wc1*n(i))) / (pi*n(i));
23     end
24 end
25
26 %% Janelas
27 w_ret = ones(1, M);
28 w_hamm = 0.54 - 0.46 * cos(2 * pi * (0:M-1) / (M - 1));
29 w_hann = 0.5 - 0.5 * cos(2 * pi * (0:M-1) / (M - 1));
30 w_black = 0.42 - 0.5 * cos(2 * pi * (0:M-1) / (M - 1)) + 0.08 * cos(4 * pi * (0:M-1) / (M - 1));
31
32 %% Aplicação
33 h_ret = h_n .* w_ret;
34 h_hamm = h_n .* w_hamm;
35 h_hann = h_n .* w_hann;
36 h_black = h_n .* w_black;
37
38 %% Resposta em frequência
39 [Hr, wn] = freqz(h_ret, 1, 1024);
40 [Hh, ~] = freqz(h_hamm, 1, 1024);
41 [Hn, ~] = freqz(h_hann, 1, 1024);
42 [Hb, ~] = freqz(h_black, 1, 1024);
43 f_norm = wn / pi;
44
45 %% Plot 1: Magnitude
46 figure('Name', 'Magnitude (Filtros FIR)', 'NumberTitle', 'off');
47 hold on; grid on;
48 plot(f_norm, 20*log10(abs(Hr)), 'LineWidth', 1.2, 'DisplayName', 'Retangular');
49 plot(f_norm, 20*log10(abs(Hh)), 'LineWidth', 1.2, 'DisplayName', 'Hamming');
50 plot(f_norm, 20*log10(abs(Hn)), 'LineWidth', 1.2, 'DisplayName', 'Hanning');
51 plot(f_norm, 20*log10(abs(Hb)), 'LineWidth', 1.2, 'DisplayName', 'Blackman');
52 xlabel('Frequência Normalizada ( rad/amostra)');
53 ylabel('Magnitude (dB)');
54 title('Resposta em Magnitude');
55 axis([0 1 -100 10]);
56 legend('show');
57
58 %% Plot 2: Fase
59 figure('Name', 'Fase (Filtros FIR)', 'NumberTitle', 'off');
60 hold on; grid on;
61 plot(f_norm, unwrap(angle(Hr)) * 180/pi, 'LineWidth', 1.2, 'DisplayName', 'Retangular');
62 plot(f_norm, unwrap(angle(Hh)) * 180/pi, 'LineWidth', 1.2, 'DisplayName', 'Hamming');

```

```

63 plot(f_norm, unwrap(angle(Hn)) * 180/pi, 'LineWidth', 1.2, 'DisplayName', 'Hanning');
64 plot(f_norm, unwrap(angle(Hb)) * 180/pi, 'LineWidth', 1.2, 'DisplayName', 'Blackman');
65 xlabel('Frequência Normalizada ( rad/amostra)');
66 ylabel('Fase (°)');
67 title('Resposta em Fase');
68 axis([0 1 -2000 200]);
69 legend('show');

```

Simulação no MATLAB - Sinal original e filtro Hanning

```

1 clear all; close all; clc;
2
3 %% Parâmetros do sinal
4 tmin = 0;           % Tempo inicial do sinal (segundos)
5 tmax = 2;           % Tempo final do sinal (segundos)
6
7 Fs = 2000;          % Frequência de amostragem (Hz)  número de amostras por segundo
8 Ts = 1/Fs;          % Período de amostragem (s)  intervalo entre duas amostras
9
10 t = tmin:Ts:tmax-Ts; % Vetor de tempo com espaçamento Ts, indo de tmin até tmax (exclusive)
11 L = length(t);       % Número total de amostras no sinal
12
13 %% Sinal de entrada s(t)
14 s = 5*sin(2*pi*50*t) + 2*sin(2*pi*300*t) + sin(2*pi*900*t);
15
16 %% FFT do sinal original
17 S = abs(fft(s));
18 S = 2*S/L;
19 S = fftshift(S);
20 freq = Fs * (-L/2:L/2-1) / L;
21
22 %% Filtro FIR com janela de Hamming
23 Omega_c1 = 250;
24 Omega_c2 = 750;
25 wc1 = Omega_c1 * 2 * pi / Fs;
26 wc2 = Omega_c2 * 2 * pi / Fs;
27 M = 71;
28 n = -(M-1)/2:(M-1)/2;
29
30 h_n = zeros(1, M);
31 for i = 1:M
32     if n(i) == 0
33         h_n(i) = (wc2 - wc1)/pi;
34     else
35         h_n(i) = (sin(wc2*n(i)) - sin(wc1*n(i))) / (pi*n(i));
36     end
37 end
38
39 w_hamm = 0.54 - 0.46 * cos(2 * pi * (0:M-1) / (M - 1));
40 h_hamm = h_n .* w_hamm;
41
42 s_filt = filter(h_hamm, 1, s);
43
44 S_filt = abs(fft(s_filt));
45 S_filt = 2*S_filt/L;
46 S_filt = fftshift(S_filt);
47
48 %% Plots
49 figure('Name','Sinal e Espectros','NumberTitle','off');
50
51 subplot(3,1,1);
52 plot(t, s, 'LineWidth', 2);

```

```

53 title('Sinal Original');
54 xlabel('t(s)');
55 ylabel('s(t)');
56 grid on;
57
58 subplot(3,1,2);
59 plot(freq, S, 'LineWidth', 2);
60 title('Espectro de Amplitude Original');
61 xlabel('f(Hz)');
62 ylabel('|S(f)|');
63 xlim([-1000 1000]);
64 grid on;
65
66 subplot(3,1,3);
67 plot(freq, S_filt, 'LineWidth', 2);
68 title('Espectro de Amplitude com Filtro Hamming');
69 xlabel('f(Hz)');
70 ylabel('|S(f)|');
71 grid on;

```

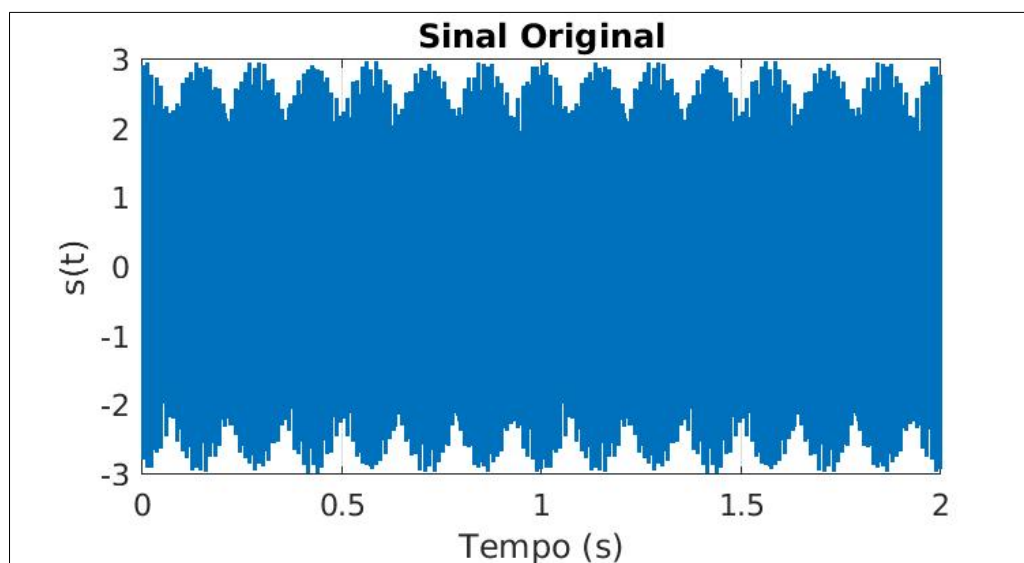
2.4 Questão 4

Criar um sinal com três senóides (770 Hz, 852 Hz e 941 Hz) e projetar três filtros passa-faixa, cada um centrado em uma dessas frequências. Cada filtro deve rejeitar as componentes vizinhas com pelo menos 60 dB de atenuação.

Resolução

A Figura 10 mostra o sinal composto pelas três senóides com frequências de 770 Hz, 852 Hz e 941 Hz. Observa-se um padrão de interferência típico de sinais senoidais com frequências próximas. Devido à superposição dessas componentes e à alta densidade de variações no tempo, a visualização do sinal apresenta um aspecto compactado, tornando difícil a identificação individual de cada frequência na forma de onda.

Figura 10: Sinal de entrada composto por senóides em 770 Hz, 852 Hz e 941 Hz

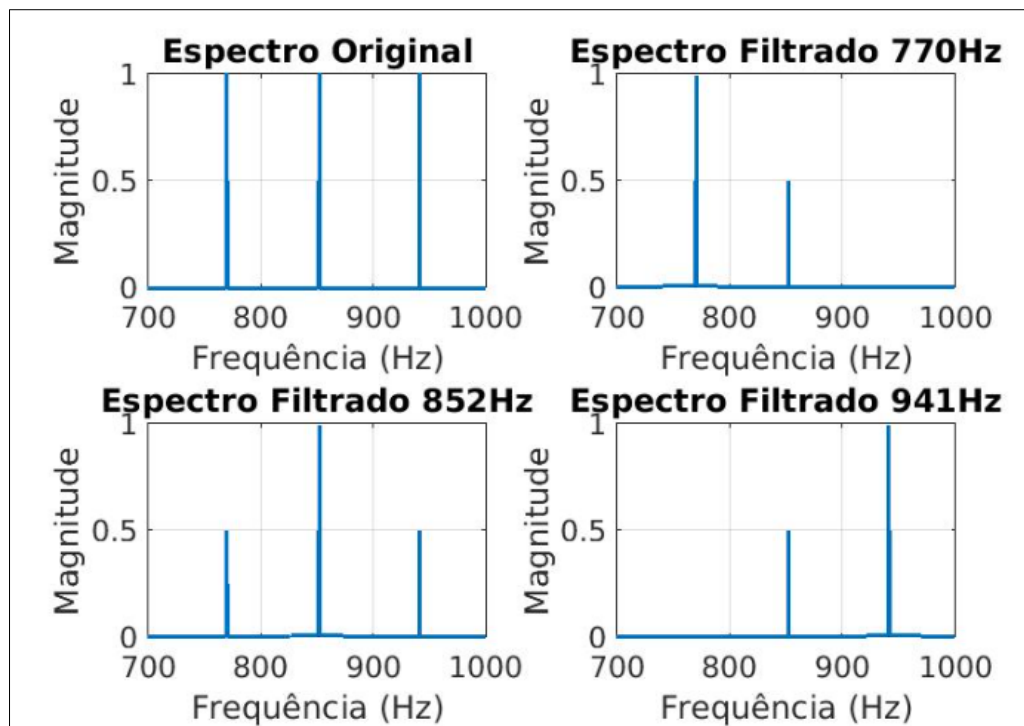


Fonte: Própria Autora

Na Figura 11, são apresentados os espectros de amplitude do sinal original e os espectros resultantes após a aplicação de cada um dos três filtros passa-faixa projetados com janelas apropriadas.

Os filtros foram centrados, respectivamente, em 770 Hz, 852 Hz e 941 Hz, rejeitando as componentes vizinhas com pelo menos 60 dB de atenuação, conforme solicitado.

Figura 11: Espectros de amplitude do sinal original e dos sinais filtrados para 770 Hz, 852 Hz e 941 Hz



Fonte: Própria Autora

A análise dos espectros mostra que cada filtro projetado isola eficientemente sua frequência central (770 Hz, 852 Hz ou 941 Hz), suprimindo as demais componentes. Além disso, as bandas de rejeição atingem níveis superiores a 60 dB, o que confirma a eficácia dos filtros. Esse resultado está em conformidade com o comportamento esperado dos filtros FIR projetados.

Simulação no MATLAB

```

1 clear all; close all; clc;
2
3 %% Parâmetros
4
5 M = 301;           % Ordem do filtro FIR (número de coeficientes = M)
6 Fs = 8000;         % Frequência de amostragem em Hz (amostras por segundo)
7 Ts = 1/Fs;         % Período de amostragem (segundos) intervalo entre amostras consecutivas
8 tmin = 0;          % Tempo inicial da simulação (em segundos)
9 tmax = 2;          % Tempo final da simulação (em segundos)
10 t = tmin:Ts:tmax-Ts; % Vetor de tempo discretizado de tmin até tmax (exclusivo), com passo Ts
11 L = length(t);     % Comprimento do vetor de tempo (número total de amostras do sinal)
12
13 %% Frequências dos filtros
14 % Filtro 1: 697852 Hz
15 wc1 = 697 * 2*pi / Fs;
16 wc2 = 852 * 2*pi / Fs;
17
18 % Filtro 2: 770941 Hz
19 wc3 = 770 * 2*pi / Fs;
20 wc4 = 941 * 2*pi / Fs;
21

```



```

22 % Filtro 3: 8521209 Hz
23 wc5 = 852 * 2*pi / Fs;
24 wc6 = 1209 * 2*pi / Fs;
25
26 %% Vetor n e janela Blackman
27 n = -(M-1)/2:(M-1)/2;
28 w_black = 0.42 + 0.5*cos(2*pi*n/M) + 0.08*cos(4*pi*n/M);
29
30 %% Respostas ao impulso ideais + correções em n=0
31 h1 = (sin(wc2*n) - sin(wc1*n)) ./ (pi*n); h1((M+1)/2) = (wc2 - wc1)/pi;
32 h2 = (sin(wc4*n) - sin(wc3*n)) ./ (pi*n); h2((M+1)/2) = (wc4 - wc3)/pi;
33 h3 = (sin(wc6*n) - sin(wc5*n)) ./ (pi*n); h3((M+1)/2) = (wc6 - wc5)/pi;
34
35 % Aplicar janela Blackman
36 h_black1 = h1 .* w_black;
37 h_black2 = h2 .* w_black;
38 h_black3 = h3 .* w_black;
39
40 %% Sinal original (soma das três frequências)
41 s_t = sin(2*pi*770*t) + sin(2*pi*852*t) + sin(2*pi*941*t);
42
43 %% FFT do sinal original
44 S_f = fftshift(fft(s_t));
45 S_f = abs(S_f) * 2 / L;
46 freq = Fs * (-L/2:L/2-1) / L;
47
48 %% Aplicação dos filtros
49 y1 = filter(h_black1, 1, s_t);
50 y2 = filter(h_black2, 1, s_t);
51 y3 = filter(h_black3, 1, s_t);
52
53 % FFTs dos sinais filtrados
54 Y1 = abs(fftshift(fft(y1))) * 2 / L;
55 Y2 = abs(fftshift(fft(y2))) * 2 / L;
56 Y3 = abs(fftshift(fft(y3))) * 2 / L;
57
58 %% Plot 1: Sinal original no tempo
59 figure('Name','Sinal Original no Tempo','NumberTitle','off');
60 plot(t, s_t, 'LineWidth', 1.5);
61 xlabel('Tempo (s)');
62 ylabel('s(t)');
63 title('Sinal Original');
64 grid on;
65
66 %% Plot 2: Espectros de amplitude
67 figure('Name','Espectros dos Sinais Filtrados','NumberTitle','off');
68
69 subplot(2,2,1);
70 plot(freq, S_f, 'LineWidth', 1.5);
71 title('Espectro Original');
72 xlabel('Frequência (Hz)');
73 ylabel('Magnitude');
74 xlim([700 1000]); ylim([0 1]);
75 grid on;
76
77 subplot(2,2,2);
78 plot(freq, Y1, 'LineWidth', 1.5);
79 title('Espectro Filtrado 770Hz');
80 xlabel('Frequência (Hz)');
81 ylabel('Magnitude');
82 xlim([700 1000]); ylim([0 1]);
83 grid on;
84

```

```

85 subplot(2,2,3);
86 plot(freq, Y2, 'LineWidth', 1.5);
87 title('Espectro Filtrado 852Hz');
88 xlabel('Frequência (Hz)');
89 ylabel('Magnitude');
90 xlim([700 1000]); ylim([0 1]);
91 grid on;
92
93 subplot(2,2,4);
94 plot(freq, Y3, 'LineWidth', 1.5);
95 title('Espectro Filtrado 941Hz');
96 xlabel('Frequência (Hz)');
97 ylabel('Magnitude');
98 xlim([700 1000]); ylim([0 1]);
99 grid on;

```

3 Conclusão

Neste trabalho, estudamos o projeto de filtros digitais FIR utilizando diversas janelas: Hamming, Hanning, Blackman, Retangular e Kaiser. As simulações no MATLAB nos permitiram observar como a escolha da janela e a ordem do filtro impactam a resposta em frequência, a seletividade, a largura da banda de transição e os níveis de atenuação.

A variação da ordem M se mostrou muito importante para o desempenho dos filtros: valores maiores de M resultaram em uma banda passante mais bem definida e maior rejeição em faixas indesejadas. Janelas com cauda mais suave, como a Blackman, ofereceram um controle superior da atenuação, enquanto a Retangular apresentou mais lobos laterais.

Também aplicamos os filtros projetados a sinais compostos, o que demonstrou sua eficácia na filtragem seletiva de componentes de frequência específicas. Os resultados confirmam a teoria por trás dos métodos para desenvolvimento de filtros digitais eficientes.

Portanto, a técnica de janelamento se mostra uma abordagem versátil para projetar filtros FIR, sendo aplicável em sistemas de processamento digital de sinais.