Lista 4 - Processamento de Sinais

Emiliano Toledo - 2016106756

Outubro de 2020

1) Os filtros de Butterworth são especificados a partir de uma resposta de magnitude ao quadrado dada por

$$|H(j\omega)|^2 = \frac{1}{1+(\omega/\omega_c)^{2N}},$$

em que N é a ordem e ω_c é denominada frequência de corte, de ½ potência ou de $-3\,dB$. Considerando um filtro de Butterworth de ordem N = 7 e frequência de corte $f_c = 350\,Hz$: :

a) Determine, analiticamente, os valores dos polos, zeros e a função de transferência $H(s)$.

b) Faça um programa ou script do Matlab® que:
   - forneça os valores dos polos e zeros e a função de transferência $H(s)$;
   - plote o diagrama de polos e zeros;
   - plote as curvas de superfície de $|H(s)|$ e $\theta(s)$;
   - plote as respostas de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$.

   Verifique se os valores condizem com os calculados anteriormente. Corrija eventuais discrepâncias e apresente, como resposta, o código do programa e os gráficos gerados.

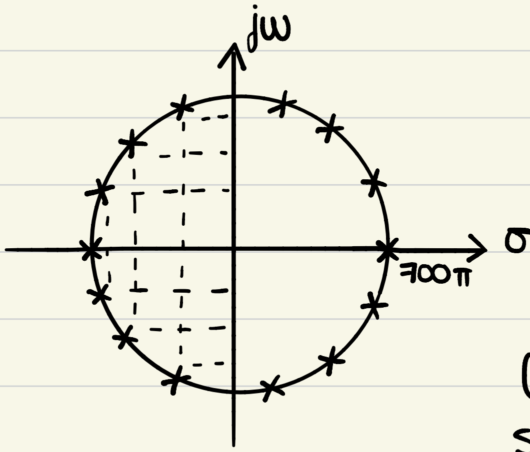
(a) $|H(j\omega)|^2 = \dfrac{1}{1+\left(\dfrac{\omega}{\omega_c}\right)^{2N}}$, $N = 7$

$\omega_c = 700\pi\ rad \cdot s^{-1}$

Genericamente: $s_k = \omega_c\, e^{\frac{j(k-1)\pi}{N}}$, $k = 1, \ldots, 2N$

Substituindo os valores:

$s_k = 700\pi\, e^{\frac{j(k-1)\pi}{7}} = 700\pi \cdot s_k'$



Obs. Os polos são simétricos.

$s_k' = \cos\left[\frac{(k-1)\pi}{7}\right] + j\,\text{sen}\left[\frac{(k-1)\pi}{7}\right]$

$k = 1, \ldots, 14$; Tomando $\omega_c = 1\ rad \cdot s^{-1}$ e escolhendo os do semiplano esquerdo: $k = 5$ até $k = 11$

$s_5' = -0.2225 + j0.9749$
$s_{11}' = -0.2225 + j0.9749$
$s_6' = -0.6235 + j0.7818$
$s_{10}' = -0.6235 - j0.7818$
$s_7' = -0.9010 + j0.4339$
$s_9' = -0.9010 - j0.4339$
$s_8' = -1$

Polos do semiplano esquerdo. Os do direito são obtidos para o resto dos valores de k.

$H(s) = \dfrac{1}{\prod_{k=5}^{11}(s - 700\pi s_k')}$

(b) Código:

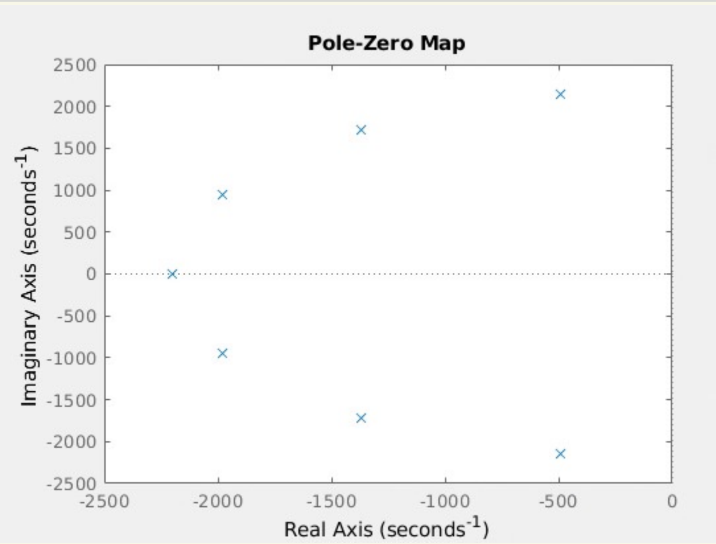
```
% Questao 1
%% Definicao da funcao de transferencia
format long

% Frequencia de Corte
wc = 2*pi*350; % Rad/s
N = 7; % Ordem

% Projeto do Filtro
[z_b1,p_b1,g_b1] = butter(N,wc,'s');
[num,den] = zp2tf(z_b1,p_b1,g_b1);
butter_1 = tf(num,den);

%% Informações
P = pole(butter_1);
Z = zero(butter_1);

disp(' ');
disp('Polos:');
display(P);
disp(' ');
disp('Zeros:');
disp(Z);
disp(' ');
disp('Função de Transferência:');
display(butter_1);

%% Polos e zeros
figure();
pzplot(butter_1);


%% Resposta em frequencia
figure();
bode(butter_1)
grid on
```

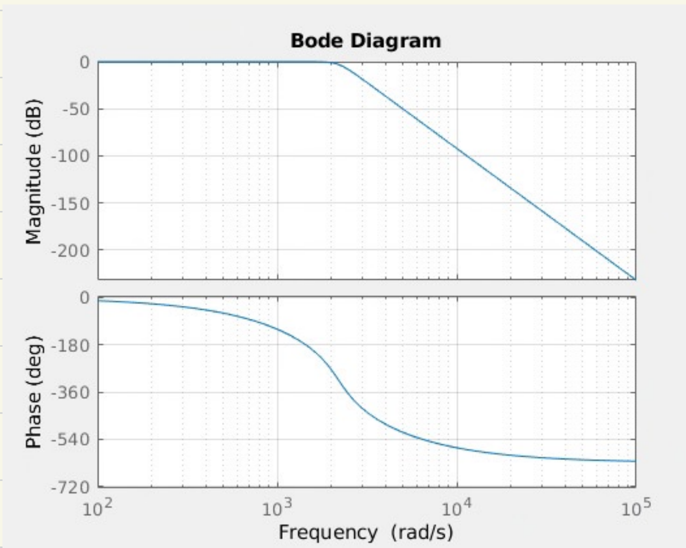
```
%% Superficie - Magnitude
% Dominio da Funcao
N_points = 1e2;
sigma = linspace(-1e4,1e4,N_points);
omega = linspace(-1e4,1e4,N_points);
[s_plane_sig,s_plane_om] = meshgrid(sigma,omega);

% Calculando as frequencias
z_mag = zeros(N_points,N_points);
z_pha = zeros(N_points, N_points);

for ctr1 = 1:N_points
    for ctr2 = 1:N_points
        s = sigma(ctr1)+1i*omega(ctr2);
        freq = evalfr(butter_1,s);
        z_mag(ctr1,ctr2) = abs(freq);
        z_pha(ctr1,ctr2) = (180/pi)*angle(freq);

    end
end

%% Desenhando Magnitude
figure();
surf(s_plane_sig,s_plane_om,z_mag)
title('Magnitude')
xlabel('j\omega')
ylabel('\sigma')
zlabel('H(j\omega')

%% Desenhando Fase
figure();
surf(s_plane_sig,s_plane_om,z_pha)
title('Fase')
xlabel('j\omega')
ylabel('\sigma')
zlabel('\theta(j\omega)')
```

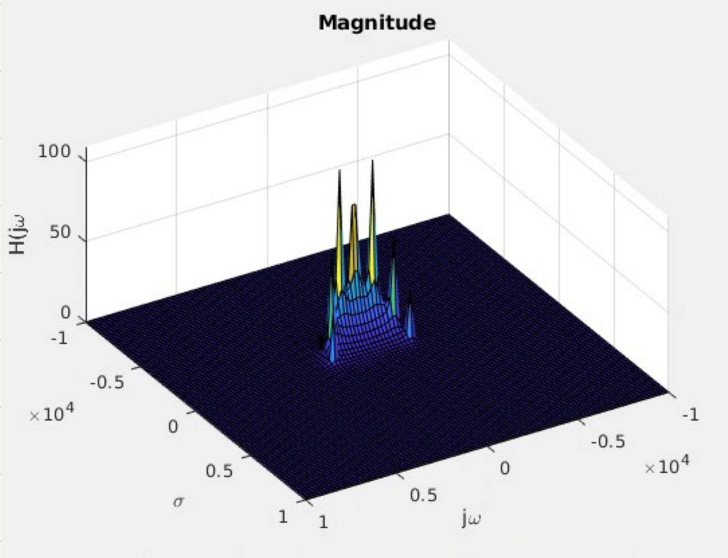
Imagens:

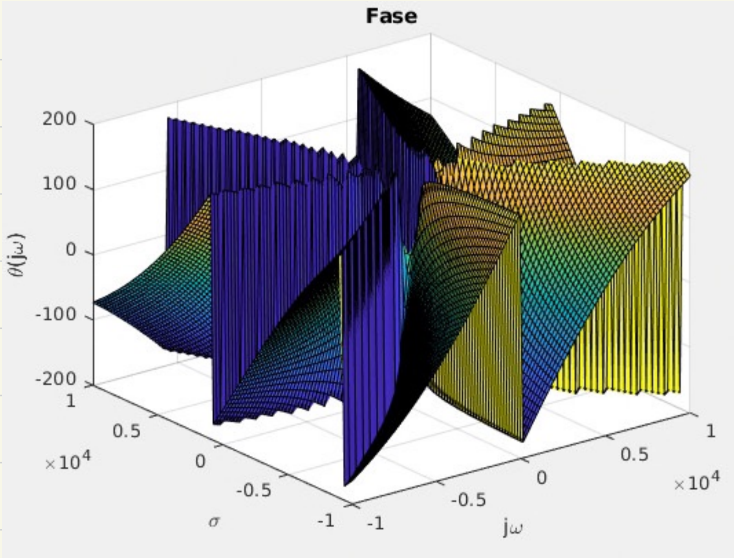
```
Função de Transferência:

butter_1 =

                                              2.487e23
  ---------------------------------------------------------------------------------------------
  s^7 + 9883 s^6 + 4.883e07 s^5 + 1.552e11 s^4 + 3.413e14 s^3 + 5.194e17 s^2 + 5.083e20 s + 2.487e23

Continuous-time transfer function.
```

```
Polos:

P =

   1.0e+03 *

  -0.489349091970980 + 2.143978456683036i
  -0.489349091970980 - 2.143978456683036i
  -1.371125686775234 + 1.719337229166736i
  -1.371125686775234 - 1.719337229166736i
  -2.199114857512729 + 0.000000000000000i
  -1.981334023560850 + 0.954160177126665i
  -1.981334023560850 - 0.954160177126665i


Zeros:
```









Zeros Matlab: Não há
Pólos Matlab:

-0.222520933956314 +	0.974927912181824i
-0.222520933956314 -	0.974927912181824i
-0.623489801858733 +	0.781831482468030i
-0.623489801858733 -	0.781831482468030i
-0.900968867902420 +	0.433883739117557i
-0.900968867902420 -	0.433883739117557i
-1.000000000000000 +	0.000000000000000i

Os valores gerados pelo Matlab sao os mesmos dos analíticos, diferindo apenas por uma questão de aproximação. Note que esses valores são resultados da expressão p_b1./(700*pi) (ver código)

2) Utilize o filtro projetado no exercício 1 para fazer a síntese de um filtro passa-faixa com os mesmos parâmetros mas com frequência central $f_0 = 5\,kHz$. Novamente, reporte os valores dos polos, zeros, função de transferência, o script e as figuras do diagrama de polos e zeros e da resposta em frequência.

Transformação realizada: PB → PF
Frequência central: $f_0 = 5\,kHz$

Seja $f_1 = 350\,Hz$, então: $f_2 = \dfrac{f_0^2}{f_1} = \dfrac{(5 \cdot 10^3)^2}{350} = 71.4286\,kHz$ //

$$B = 71.4286k - 0.35k = 71.0786 \cdot 10^3 \cdot 2\pi\ rad \cdot s^{-1} //$$

Da questão anterior:

$$H_{LP}(s) = \frac{1}{[s-(-0.2225 \pm j0.9749)][s-(-0.6235 \pm j0.7818)][s-(-0.9010 \pm j0.4339)][s+1]} //$$

$$H_{BP}(s) = H_{LP}\left(\frac{s^2 + (10^4\pi)^2}{s \cdot 142.5172\pi \cdot 10^3}\right) //$$

Código:

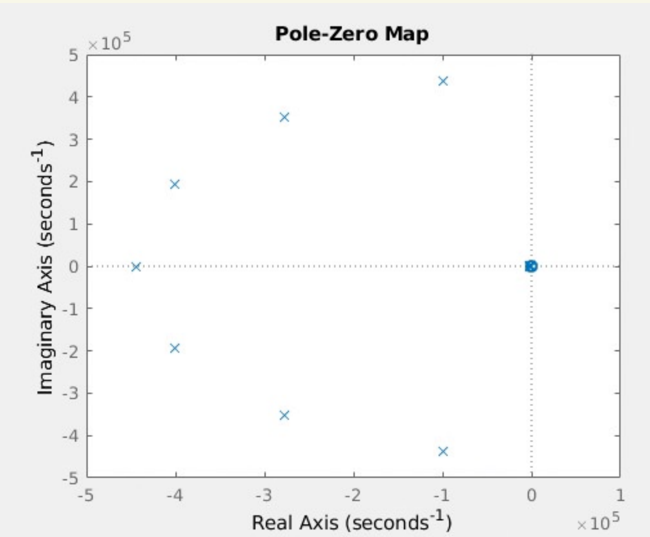
```
% Questao 2
%% Definicao de Filtro Butterworth PB prototipo
format long

% Parametros do filtro iguais ao anterior
N = 7; % Ordem
wc = 1; % Frequencia de corte para filtro prototipo

[z_but,p_but,g_but] = butter(N,wc,'s');
[num_but,den_but] = zp2tf(z_but,p_but,g_but);
butter_prototipo = tf(num_but,den_but);

%% Filtro PF

% Parametros
w_central = 5000*2*pi;
w_inferior = 700*pi;

% Definicao da banda
w_superior = (w_central^2)/w_inferior;
Bw = w_superior - w_inferior;

%% Projeto do Filtro
[z_pf,p_pf]=lp2bp(num_but,den_but,w_central,Bw);
fpf = tf(z_pf,p_pf);

%% Informações
P = pole(fpf);
Z = zero(fpf);

disp(' ');
disp('Polos:');
display(P);
disp(' ');
disp('Zeros:');
disp(Z);
disp(' ');
disp('Função de Transferência:');
display(fpf);

%% Polos e zeros
figure();
pzplot(fpf);
```

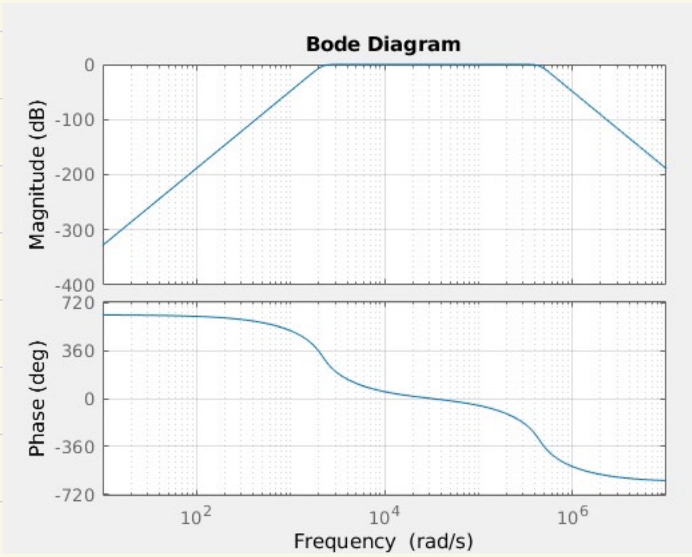
```
%% Resposta em Frequencia
figure();
bode(fpf)
grid on

%% Superficie - Magnitude
% Dominio da Funcao
N_points = 1e2;
sigma = linspace(-1e4,1e4,N_points);
omega = linspace(-1e4,1e4,N_points);
[s_plane_sig,s_plane_om] = meshgrid(sigma,omega);

% Calculando as frequencias
z_mag = zeros(N_points,N_points);
z_pha = zeros(N_points, N_points);

for ctr1 = 1:N_points
    for ctr2 = 1:N_points
        s = sigma(ctr1)+1i*omega(ctr2);
        freq = evalfr(fpf,s);
        z_mag(ctr1,ctr2) = abs(freq);
        z_pha(ctr1,ctr2) = (180/pi)*angle(freq);

    end
end

%% Desenhando Magnitude
figure();
surf(s_plane_sig,s_plane_om,z_mag)
title('Magnitude')
xlabel('j\omega')
ylabel('\sigma')
zlabel('H(j\omega)')

%% Desenhando Fase
figure();
surf(s_plane_sig,s_plane_om,z_pha)
title('Fase')
xlabel('j\omega')
ylabel('\sigma')
zlabel('\theta(j\omega)')
```

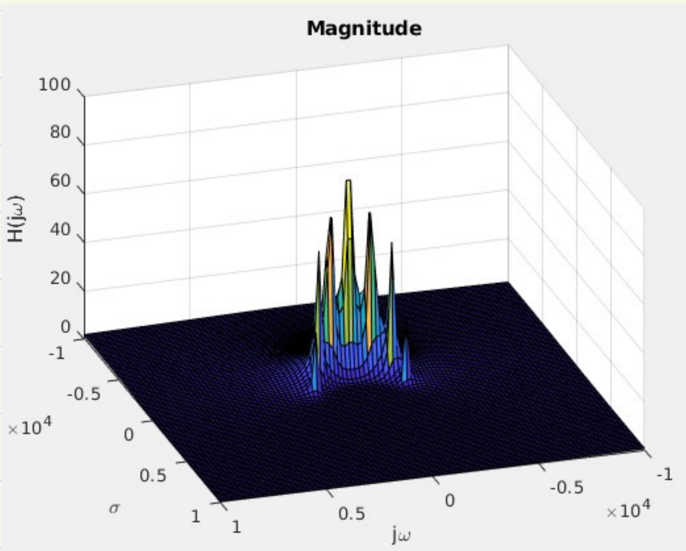
Imagens:

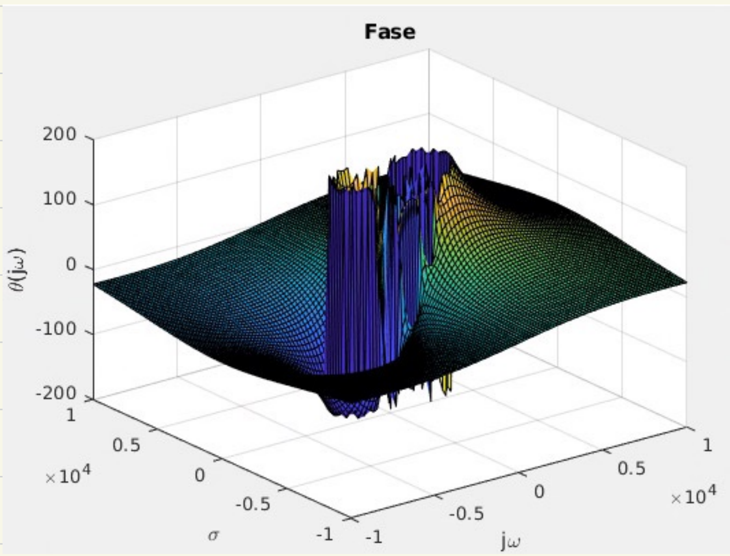
```
Função de Transferência:

fpf =

          3.543e39 s^7 - 2.439e25 s^6 - 1.112e30 s^5 - 1.55e26 s^4 - 4.286e34 s^3 - 1.578e26 s^2 - 3.679e39 s - 6.584e-42

  ---------------------------------------------------------------------------------------------------------------------

  s^14 + 2.007e06 s^13 + 2.021e12 s^12 + 1.312e18 s^11 + 5.904e23 s^10 + 1.846e29 s^9 + 3.74e34 s^8 + 3.905e39 s^7 + 3.691e43 s^6 + 1.798e47 s^5

                                                   + 5.676e50 s^4 + 1.245e54 s^3 + 1.893e57 s^2 + 1.855e60 s + 9.122e62

Continuous-time transfer function.
```

```
Polos:

P =

   1.0e+05 *

  -0.988927758919782 + 4.375486781679522i
  -0.988927758919782 - 4.375486781679522i
  -2.770824411680413 + 3.508982543428719i
  -2.770824411680413 - 3.508982543428719i
  -4.443788468592220 + 0.000000000000000i
  -4.003790933917016 + 1.947420112193297i
  -4.003790933917016 - 1.947420112193297i
  -0.004850366427796 + 0.021460328117703i
  -0.004850366427796 - 0.021460328117703i
  -0.013680018748309 + 0.017324427624948i
  -0.013680018748309 - 0.017324427624948i
  -0.022209887961244 + 0.000000000000000i
  -0.019934549440614 + 0.009696046359289i
  -0.019934549440614 - 0.009696046359289i

Zeros:
  -1.006262856434231 + 0.000000000000000i
  -0.503128460627626 + 0.871447483131488i
  -0.503128460627626 - 0.871447483131488i
   0.503128460627614 + 0.871447483131488i
   0.503128460627614 - 0.871447483131488i
   1.006262856434262 + 0.000000000000000i
   0.000000000000000 + 0.000000000000000i
```









O comportamento esperado foi atingido pelo filtro.

3) A resposta de magnitude ao quadrado de um filtro Chebyshev tipo I passa-baixas é definida por

$$|H_I(j\omega)|^2 = \frac{1}{1+\epsilon^2 C_{I,N}^2(\omega/\omega_p)},$$

em que ϵ determina o nível de *ripple* na faixa de passagem, ω_p é a frequência de borda da faixa de passagem e $C_{I,N}(\omega/\omega_p)$ é a função de Chebyshev tipo I de ordem *N*. Na forma normalizada, ou seja, para $\omega_p = 1\,rad/s$, $C_{I,N}(\omega)$ pode ser tabelado para cada *N*, tal como mostrado na tabela 1.

A resposta de magnitude ao quadrado de um filtro Chebyshev tipo II pode ser obtida a partir da tipo I fazendo-se $|H_{II}(j\omega)|^2 = 1 - |H_I(j\omega)|^2$ e substituindo-se ω/ω_p por ω_s/ω, O resultado é

$$|H_{II}(j\omega)|^2 = \frac{\epsilon^2 C_{II,N}^2(\omega_s/\omega)}{1+\epsilon^2 C_{II,N}^2(\omega_s/\omega)}$$

⋮

Considere um filtro Chebyshev tipo II <u>*normalizado*</u> em $\omega_s = 1\,rad/s$ de ordem *N* = 4 e $\alpha_s = 60\,dB$.

a) Determine a expressão para a resposta de magnitude ao quadrado, $|H_{II}(j\omega)|^2$.
b) Determine, analiticamente, os valores dos polos e zeros.
c) Determine a função de transferência $H(s)$.
d) Faça um programa ou script do Matlab® para projetar o filtro especificado e que:
   - forneça os valores dos polos e zeros e a função de transferência *H*(*s*);
   - plote o diagrama de polos e zeros;
   - plote as respostas de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$.

   Verifique se os valores obtidos condizem com os calculados anteriormente. Apresente, como resposta, o código do programa e os gráficos gerados.

(a) $\alpha s = 60dB$, então: $\epsilon = (\sqrt{10^6-1})^{-1} = 0.001$

Se $N=4$, seleciona-se o polinômio:

$$C_N(\omega) = 8(1/\omega)^4 - 8(1/\omega)^2 + 1$$

Para Chebyshev tipo II: $$|H(j\omega)|^2 = \frac{\epsilon^2 C_N^2(\omega s/\omega)}{1+\epsilon^2 C_N^2(\omega s/\omega)}$$

$$|H(j\omega)|^2 = \frac{\epsilon^2[8(1/\omega)^4-8(1/\omega)^2+1]^2}{1+\epsilon^2[8(1/\omega)^4-8(1/\omega)^2+1]^2}, \quad \text{seja } a=\frac{1}{\omega}$$

$$(8a^4-8a^2+1)(8a^4-8a^2+1)$$

$$64a^8-64a^6+8a^4-64a^6+64a^4-8a^2+8a^4-8a^2+1$$

$$64a^8-128a^6+80a^4-16a^2+1$$

$$|H(j\omega)|^2 = \frac{\epsilon^2(64a^8-128a^6+80a^4-16a^2+1)}{1+\epsilon^2(64a^8-128a^6+80a^4-16a^2+1)}, \quad a=\frac{1}{\omega}, \quad \epsilon=0.001$$

(b) Seja $W=\frac{s}{j}$, $\therefore a=\frac{j}{s}$, logo:

$$= \frac{\epsilon^2(64/s^8+128/s^6+80/s^4+16/s^2+1)}{1+\epsilon^2(64/s^8+128/s^6+80/s^4+16/s^2+1)}$$

$$= \frac{\epsilon^2(64+128s^2+80s^4+16s^6+s^8)}{\epsilon^2(64+128s^2+80s^4+16s^6+s^8)+s^8} \;//$$

Utilizando a função roots:

Zeros:

```
>> roots([e 0 16*e 0 80*e 0 128*e 0 64*e])

ans =

 -0.000000014414249 + 2.613125931200642i
 -0.000000014414249 - 2.613125931200642i
  0.000000014414248 + 2.613125928304860i
  0.000000014414248 - 2.613125928304860i
  0.000000010320881 + 1.082392203031871i
  0.000000010320881 - 1.082392203031871i
 -0.000000010320881 + 1.082392197552914i
 -0.000000010320881 - 1.082392197552914i
```

Tem-se o dobro de polos e zeros como esperado, pois se trata da função $H(s)\cdot H(-s)$. Deve-se selecionar os pólos e zeros do semiplano esquerdo.

Pólos:

```
>> e = 0.001;
>> e = 0.001^2;
>> roots([(1+e) 0 16*e 0 80*e 0 128*e 0 64*e])

ans =

 -0.278803183811681 + 0.120766788885021i
 -0.278803183811681 - 0.120766788885021i
  0.278803183811681 + 0.120766788885021i
  0.278803183811681 - 0.120766788885021i
 -0.108407525527823 + 0.273691047696113i
 -0.108407525527823 - 0.273691047696113i
  0.108407525527823 + 0.273691047696113i
  0.108407525527823 - 0.273691047696113i
```

$$(c)\ H(s) = \frac{K\cdot(s\pm j2.6131)(s\pm j1.0824)}{(s+0.2788\pm j0.1208)(s+0.1084\pm j0.2737)} = \frac{K\cdot(s^4+8s^2+8)}{s^4+0.7744s^3+0.2999s^2+0.0683s+0.008}$$

$$H(0) = \frac{K\cdot 8}{0.008} = \frac{1}{\sqrt{1+0.001^2}},\quad K = 0.001$$

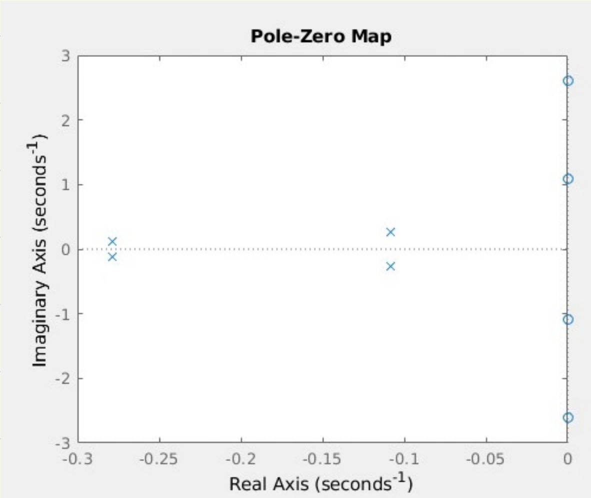
(d) Código:

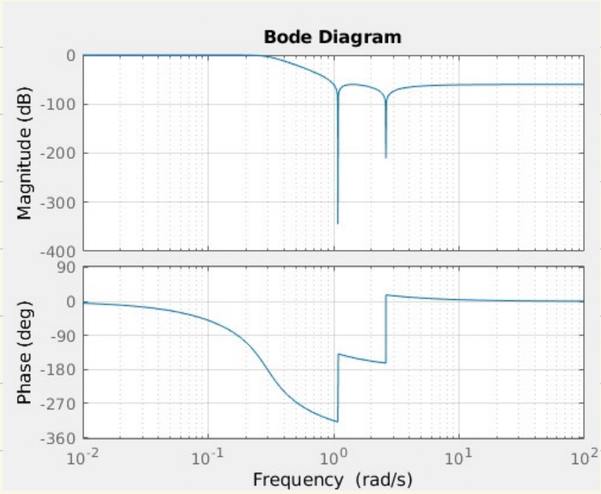
```
% Questao 3
%% Definicao do Filtro Chebyshev

% Parametros
N = 4; % Ordem
As = 60; % dB

[z_cb,p_cb,k_cb] = cheb2ap(N,As);
[num_cb,den_cb] = zp2tf(z_cb,p_cb,k_cb);
chebyshev_prototipo = tf(num_cb,den_cb);

%% Funcao de Transferencia, Polos e Zeros
disp('Zeros:')
display(z_cb)
disp('Polos:')
display(p_cb)
disp('Funcao de Transferencia:')
display(chebyshev_prototipo)

%% Polos e Zeros
figure();
pzplot(chebyshev_prototipo);

%% Resposta em Frequencia
figure();
bode(chebyshev_prototipo)
grid on
```

Imagens:

```
Funcao de Transferencia:

chebyshev_prototipo =

            0.001 s^4 + 0.008 s^2 + 0.008
  ---------------------------------------------------
  s^4 + 0.7744 s^3 + 0.2999 s^2 + 0.06834 s + 0.008

Continuous-time transfer function.
```

```
Zeros:

z_cb =

   0.000000000000000 + 1.082392200292394i
   0.000000000000000 - 1.082392200292394i
   0.000000000000000 + 2.613125929752753i
   0.000000000000000 - 2.613125929752753i

Polos:

p_cb =

  -0.108407537602265 - 0.273691081241343i
  -0.278803219270872 - 0.120766805595465i
  -0.278803219270872 + 0.120766805595465i
  -0.108407537602265 + 0.273691081241343i
```





A comparação de resultados mostra que os valores calculados diferem muito pouco dos determinados pelo Matlab, indicando, novamente, que diferiram por questão de precisão e/ou arredondamento. As funções de transferência estão condizentes.

4) Filtros Chebyshev tipo II são naturalmente especificados em função da frequência de borda da faixa de rejeição, ω_s . Para se fazer um projeto em termos da frequência de borda da faixa de passagem, ω_p, é necessário determinar qual será a ω_s correspondente, o que pode ser feito por meio da expressão

$$\omega_s = \omega_p \cosh\left\{(1/N)\, acosh\left[1/\left(\epsilon\sqrt{10^{\alpha_p/10}-1}\right)\right]\right\}.$$

em que α_p representa o ripple na faixa de passagem, em dB.

Considerando essas informações, faça um programa ou script do Matlab® para projetar um filtro Chebyshev passa-baixas tipo II com $\omega_p = 2\pi 500\, rad/s$, $\alpha_p = 0{,}5\, dB$ e $\alpha_s = 60\, dB$.

O programa deverá solicitar ao usuário a ordem desejada e retornar as seguintes informações:

a) Frequência de borda da faixa de rejeição ω_s corrigida.

b) Polos, zeros e diagrama correspondente.

c) Função de transferência $H(s)$.
d) As respostas de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$.
Apresente, como resposta, o código do programa e os gráficos gerados.

Código:

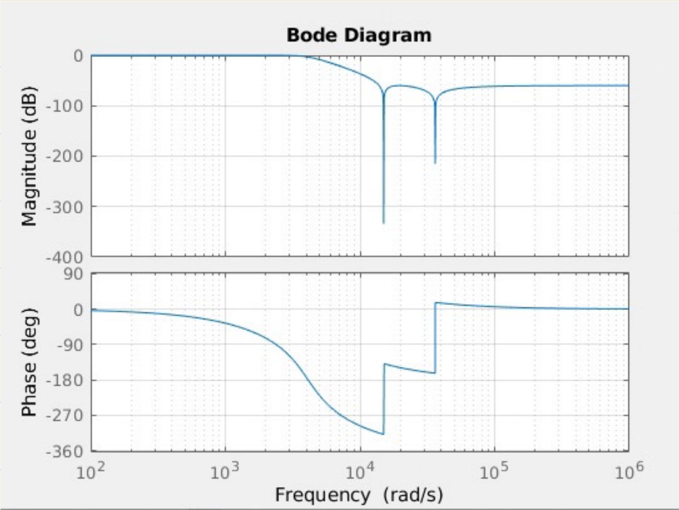
```
% Questao 4
%% Coletando informações do usuário

N = input('Entre com a ordem do filtro: ');

%% Projeto do filtro
format long

% Parâmetros do filtro
w_p = 1000 * pi; %rad/s
alpha_p = .5; %dB
alpha_s = 60; %dB

% Cálculo e projeto do filtro
e = 1/sqrt(10^(alpha_s/10) - 1);
w_s = w_p * cosh((1/N) * acosh(1/(e * sqrt(10^(alpha_p/10) - 1))));

[num,den] = cheby2(N,alpha_s,w_s,'s');
cheby = tf(num,den);

%% Frequencia de borda da faixa de rejeição
disp(' ');
disp('Frequência de borda da faixa de rejeição');
display(w_s);

%% Polos, zeros e diagrama correspondente
P = pole(cheby);
Z = zero(cheby);

disp(' ');
disp('Polos:')
disp(P);
disp(' ');
disp('Zeros:')
disp(Z);

figure();
pzplot(cheby);
```

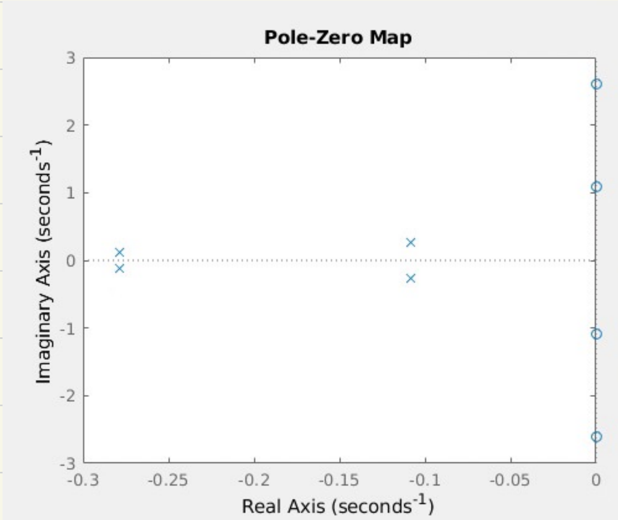
```
%% Função de transferência
disp(' ');
disp('Função de Transferência:')
display(cheby);

%% Resposta em Frequência
figure();
bode(cheby);
grid on
```

Exemplo para N=4:

```
Entre com a ordem do filtro: 4

Frequência de borda da faixa de rejeição

w_s =

     1.384446113318549e+04


Polos:
   1.0e+03 *

 -1.500843940878920 + 3.789105536745292i
 -1.500843940878920 - 3.789105536745292i
 -3.859880333002568 + 1.671951346245380i
 -3.859880333002568 - 1.671951346245380i


Zeros:
   1.0e+04 *

 -0.000000000000000 + 3.617732037058120i
 -0.000000000000000 - 3.617732037058120i
  0.000000000000000 + 1.498513674781119i
  0.000000000000000 - 1.498513674781119i
```

```
Função de Transferência:

cheby =

  0.001 s^4 + 6.334e-16 s^3 + 1.533e06 s^2 + 3.888e-07 s + 2.939e14
  -----------------------------------------------------------------
      s^4 + 1.072e04 s^3 + 5.748e07 s^2 + 1.813e11 s + 2.939e14

Continuous-time transfer function.
```





5) Considere o desenvolvimento de um filtro elíptico passa-altas com ganho unitário e frequência de borda da faixa de passagem $f_p = 2\,kHz$. Para garantir um nível mínimo de exatidão na aplicação, a distorção máxima aceitável na faixa de passagem é de 1,5%. O filtro deve fornecer uma atenuação mínima na faixa de rejeição de 99,8%, em uma frequência máxima de $1{,}35\,kHz$. Faça um programa ou script do Matlab® para projetar o filtro especificado e que:

- determine a ordem mínima N que atenda as especificações;
- forneça os valores dos polos, zeros e a função de transferência $H(s)$;
- plote o diagrama de polos e zeros;
- plote as respostas de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$.

Apresente, como resposta, o código do programa e os gráficos gerados.

$$\alpha_p = -20 \cdot \log\left(\frac{1-0.015}{1}\right) = 0.131275390\,dB$$

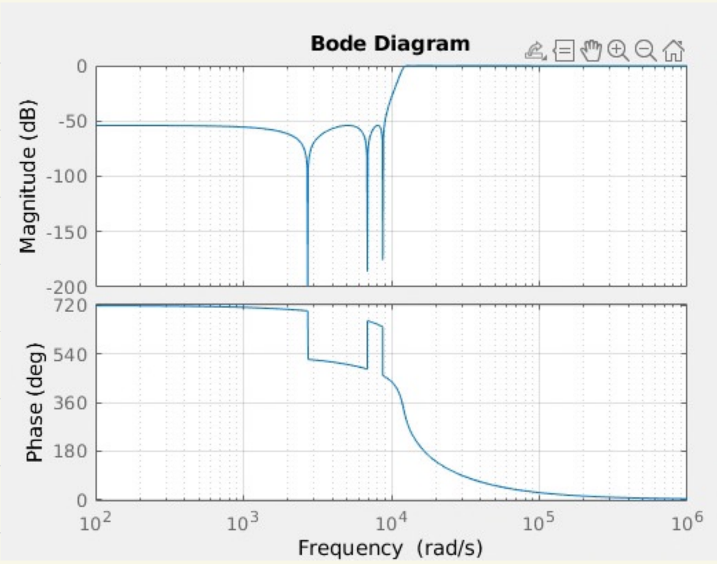
$$\alpha_s = -20 \cdot \log\left(\frac{1-0.998}{1}\right) = 53.9794001\,dB$$

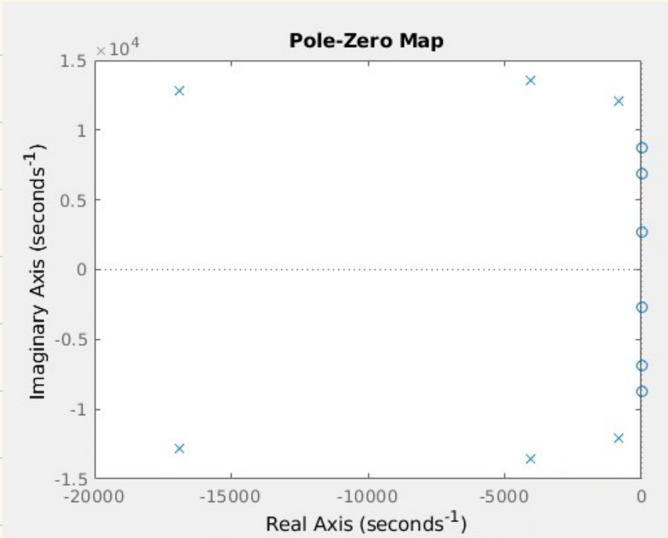
Código:

```
% Questao 5
%% Parâmetros do Filtro
w_p = 2e3 * 2 * pi; % Frequência de borda da faixa de passagem
w_s = 1.35e3 * 2 * pi; % Frequência máxima da faixa de rejeição
Rp = 0.131275390; % Distorção de 1.5% na faixa de passagem
Rs = 53.9794001; % Atenuação de 99.8% na faixa de rejeição

%% Projeto do Filtro
[N,w_p] = ellipord(w_p,w_s,Rp,Rs,'s'); % Determinação da ordem mínima
[num_elip,den_elip] = ellip(N,Rp,Rs,w_p,'high','s');
elip = tf(num_elip,den_elip);

%% Informações
P = pole(elip);
Z = zero(elip);

disp('Ordem mínima:');
display(N);
disp(' ');
disp('Polos:');
display(P);
disp(' ');
disp('Zeros:');
disp(Z);
disp(' ');
disp('Função de Transferência:');
display(elip);

%% Diagramas

% Polos e Zeros
figure();
pzplot(elip);

% Bode
figure();
bode(elip);
grid on;
```

```
Ordem mínima:

N =

     6


Polos:

P =

   1.0e+04 *

 -1.689672136362862 + 1.284206273244307i
 -1.689672136362862 - 1.284206273244307i
 -0.406674354154167 + 1.355017370011790i
 -0.406674354154167 - 1.355017370011790i
 -0.082660136407005 + 1.207424340669763i
 -0.082660136407005 - 1.207424340669763i


Zeros:
   1.0e+03 *

  0.000000000000023 + 8.755764787466632i
  0.000000000000023 - 8.755764787466632i
 -0.000000000000071 + 6.879237161769301i
 -0.000000000000071 - 6.879237161769301i
  0.000000000000053 + 2.718432608420572i
  0.000000000000053 - 2.718432608420572i
```

```
Função de Transferência:

elip =

  0.985 s^6 - 9.273e-12 s^5 + 1.294e08 s^4 - 0.00357 s^3 + 4.476e15 s^2 - 3.13e05 s + 2.641e22
  ----------------------------------------------------------------------------------------------
     s^6 + 4.358e04 s^5 + 1.141e09 s^4 + 1.81e13 s^3 + 2.429e17 s^2 + 1.676e21 s + 1.32e25

Continuous-time transfer function.
```



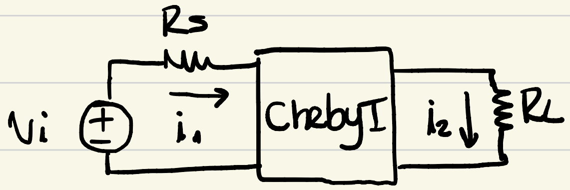


Considere o desenvolvimento de um filtro Chebyshev tipo I com ordem N = 3 utilizando um circuito passivo em escala LC.

a) Considerando o filtro normalizado em $\omega_p = 1\, rad/s$, consulte os dados da tabela 1, acima, e determine a expressão para a resposta de magnitude ao quadrado na forma polinomial,

$$|H(j\omega)|^2 = \frac{K^2}{1+\epsilon^2\left[\sum_{k=1}^{N} a_k \omega^k\right]^2}$$

a. Para $N=3$: $C(\omega) = 4\omega^3 - 3\omega$



Seja $R_S = R_L = 1\Omega$ para máxima transferência de potência. Então:

$$\begin{cases} V_{RL} = i_2 \cdot R_L \\ i_1 = i_2 \text{ (indutores curtados e capacitores abertos)} \\ V_i = V_{RS} + V_{RL} \end{cases}$$

$$V_i = R_S \cdot i_1 + R_L i_1$$
$$= (R_S + R_L)\, i_1$$
$$\therefore i_1 = i_2 = \frac{V_i}{R_S + R_L}$$

$$V_{RL} = R_L \cdot i_1 = \frac{R_L}{R_S + R_L} \cdot V_i$$

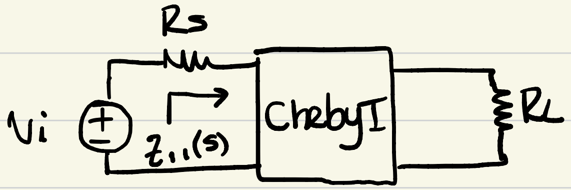
Ganho DC: $K = \dfrac{V_{RL}}{V_i} = \dfrac{R_S}{R_S + R_L}$ //

Finalmente: $|H(j\omega)|^2 = \dfrac{1}{1+\epsilon^2\left[4\omega^3 - 3\omega\right]^2} \cdot \dfrac{R_L^2}{(R_S + R_L)^2}$ //

b) Determine a expressão para a impedância de entrada $Z_{11}(s)$ (consulte o exemplo apresentado nos slides da aula 05_B).

c) Considerando $\alpha_p = 0{,}25\,dB$, $R_S = 0{,}5\,\Omega$ e $R_L = 1\,\Omega$, utilize o método do fracionamento contínuo para obter os dois possíveis circuitos em escada LC. Compare os resultados com os apresentados na tabela 10.3, disponível nos slides da aula 06_A. Corrija eventuais discrepâncias.

b. Tem-se:



$$Z_{11}(s) = R_S\left[\frac{1-A(s)}{1+A(s)}\right]^{\pm 1}$$

$$|A(j\omega)|^2 = 1 - 4\frac{R_S}{R_L}|H(j\omega)|^2 = 1 - 4\frac{R_S}{R_L}\cdot\frac{R_L^2}{(R_S+R_L)^2}\cdot\frac{1}{1+\epsilon^2[4\omega^3-3\omega]^2}$$

$$|A(j\omega)|^2 = 1 - \frac{4R_SR_L}{(R_S+R_L)^2}\cdot\frac{1}{1+\epsilon^2[4\omega^3-3\omega]^2}$$

Para $N=3$, $\alpha_p = 0.25\,dB$, $R_S = 0.5\,\Omega$ e $R_L = 1\,\Omega$:

$$\epsilon = \sqrt{10^{\frac{0.25}{10}}-1} = 0.243420881 \approx 0.2434$$

$$|A(j\omega)|^2 = 1 - \frac{4\cdot 0.5}{(1.5)^2}\cdot\frac{1}{1+(0.2434)^2\cdot(4\omega^3-3\omega)^2} = 1 - \frac{2}{2.25}\cdot\frac{1}{1+(0.2434)^2\cdot(4\omega^3-3\omega)^2}$$

Seja $\omega = \frac{s}{j}$ e $|A(j\omega)|^2 = A(s)A(-s)$:

$$A(s)A(-s) = 1 - \frac{(2/2.25)}{1+(0.2434)^2\cdot(4s^3/j - 3s/j)^2} = 1 - \frac{2/2.25}{1-(0.2434)^2\cdot[16s^6+24s^4+9s^2]}$$

$$A(s)A(-s) = \frac{1-(0.2434)^2\cdot[16s^6+24s^4+9s^2] - 2/2.25}{1-(0.2434)^2[16s^6+24s^4+9s^2]}$$

$$A(s)A(-s) = \frac{(0.2434)^2[16s^6+24s^4+9s^2]-0.1111}{(0.2434)^2[16s^6+24s^4+9s^2]-1}$$

$$A(s)A(-s) = \frac{0.9479s^6 + 1.4218s^4 + 0.5332s^2 - 0.1111}{0.9479s^6 + 1.4218s^4 + 0.5332s^2 - 1}$$

$$A(s)A(-s) = \frac{s^6 + 1.5s^4 + 0.5625s^2 - 0.1172}{s^6 + 1.5s^4 + 0.5625s^2 - 1.055} //$$

Utilizando a função "roots":

Zeros: $-0.1910 \pm j0.9271$
$+0.1910 \pm j0.9271$
± 0.3821

Pólos: $-0.3836 \pm j1.0916$
$+0.3836 \pm j1.0916$
± 0.7673

Selecionam-se os pólos e zeros no semiplano esquerdo:

$$A(s) = \frac{(s+0.1910 \pm j0.9271)(s+0.3821)}{(s+0.3836 \pm j1.0916)(s+0.7673)} //$$

Finalmente, a impedância buscada é:

$$Z_{11}(s) = R_s \left[\frac{1-A(s)}{1+A(s)}\right]^{\pm 1}$$

$$Z_{11}(s) = 1 \cdot \left[\frac{1 - \frac{(s+0.1910 \pm j0.9271)(s+0.3821)}{(s+0.3836 \pm j1.0916)(s+0.7673)}}{1 + \frac{(s+0.1910 \pm j0.9271)(s+0.3821)}{(s+0.3836 \pm j1.0916)(s+0.7673)}}\right]^{\pm 1} //$$

c. Utilizando a expressão definida na questão anterior e o Matlab para realizar as operações polinomialis:*

Para expoente +1: $Z_{11}(s) = \dfrac{0.3852s^2 + 0.4427s + 0.3424}{s^3 + 1.1493s^2 + 1.4847s + 0.6848}$ //

Para expoente −1: $Z_{11}(s) = \dfrac{s^3 + 1.1493s^2 + 1.4847s + 0.6848}{0.3852s^2 + 0.4427s + 0.3424}$ //

Obtenção dos Circuitos LC:

Expoente +1

Capacitor

$$Y_{11} = 2.5961s - \cancel{0.1137 \cdot 10^{-3}}^{\,0} + \frac{1.1915 \cdot 10^{12} s + 1.3697 \cdot 10^{12}}{0.3852s^2 + 0.4427s + 0.3424} \;//$$

Indutor

$$Z_{22} = 0.6466s - \cancel{8.2537 \cdot 10^{-5}}^{\,0} + \frac{6.8497 \cdot 10^{11}}{1.1915 \cdot 10^{12} s + 1.3697 \cdot 10^{12}} \;//$$

capacitor resistor

$$Y_{33} = 1.7396s + 2 \;//$$

Expoente −1

Indutor

$$Z_{11} = 2.5961s - \cancel{0.1137 \cdot 10^{-3}}^{\,0} + \frac{1.1915 \cdot 10^{12} s + 1.3697 \cdot 10^{12}}{0.3852s^2 + 0.4427s + 0.3424}$$

Capacitor

$$Y_{22} = 0.6466s - \cancel{8.2537 \cdot 10^{-5}}^{\,0} + \frac{6.8497 \cdot 10^{11}}{1.1915 \cdot 10^{12} s + 1.3697 \cdot 10^{12}}$$

Indutor Resistor

$$Z_{33} = 1.7396s + 2 \;//$$

Tomam-se os valores obtidos para serem comparados com a tabela 10.3 para filtros de ordem 3:

* Função "PolynomialReduce"

TABELA 10.3

N	C_1	L_2	C_3
2	1.7288	0.4104	
3	2.5965	0.6465	1.7402

VALORES CALCULADOS

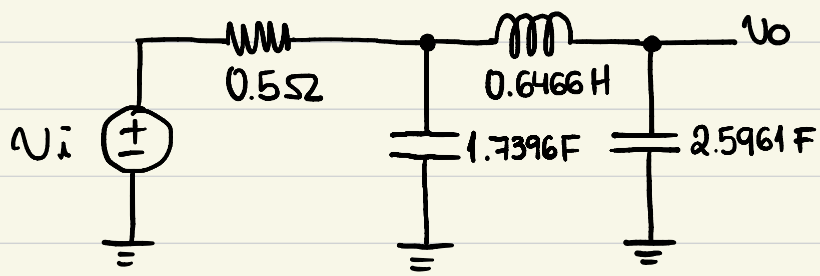
$$\boxed{\begin{aligned} C_1 &= 2.5961\,F \\ L_2 &= 0.6466\,F \\ C_3 &= 1.7396\,F \end{aligned}}$$

Os valores calculados são muito próximos aos tabelados, o que indica que o cálculo está correto. Também, Rs pode ser 0.5Ω ou 2Ω, logo, como o último termo foi 2, temos $1/2 = 0.5\Omega$ e $2 = 2\Omega$.

d) Faça os escalonamentos para que o filtro apresente uma resposta passa-altas com frequência de borda da faixa de passagem $f_p = 300\,Hz$ alimentando uma impedância de carga $R_L = 50\,\Omega$.

e) Monte o circuito em um simulador SPICE[1] e apresente o diagrama do circuito e os gráficos de resposta de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$. Corrija eventuais discrepâncias.

d. Circuito base PB:



Fatores de escala:

$$K_f = \frac{600\pi}{1} = 600\pi$$

$$K_z = \frac{50}{1} = 50$$

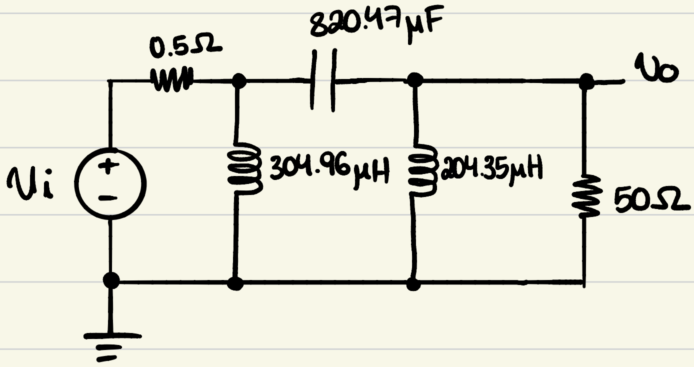
$$R_s' = R_s = 0.5\Omega$$

$$C_1 = \frac{1}{\Omega_0 \cdot L_2} = \frac{1}{600\pi \cdot 0.6466} = 820.4708\,\mu F$$

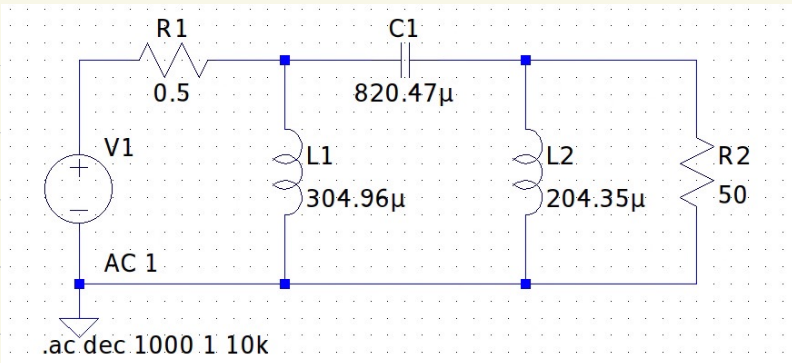
$$L_1 = \frac{1}{\Omega_0 \cdot C_3} = \frac{1}{600\pi \cdot 1.7396} = 304.9646\,\mu H$$

$$L_2 = \frac{1}{\Omega_0 \cdot C_1} = \frac{1}{600\pi \cdot 2.5961} = 204.3513\,\mu H$$

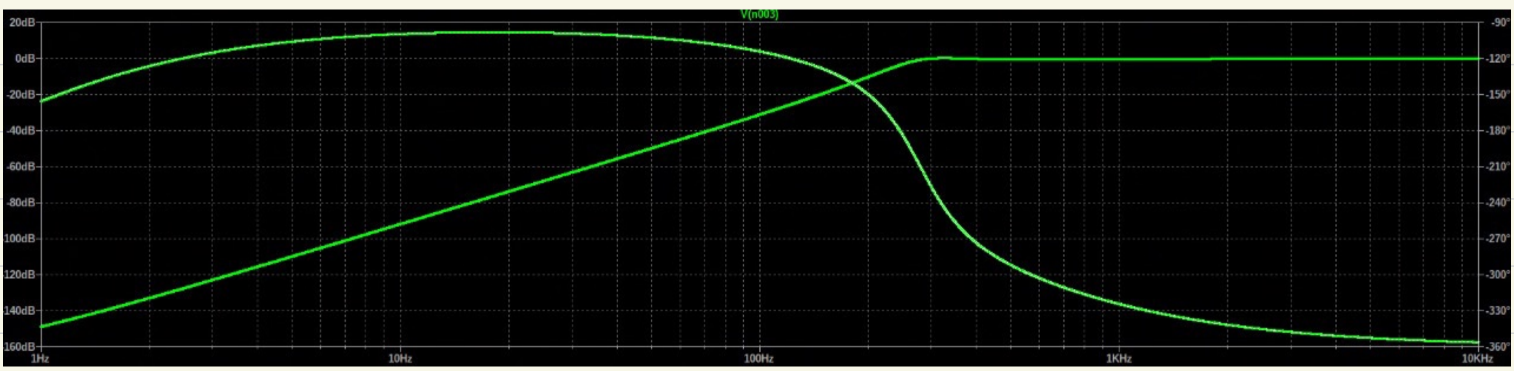
Circuito final:



e. Esquemático do LTSpice



Resposta em frequência



7) Considerando o que foi discutido a respeito das duas formas de implementação de filtros LIT de tempo contínuo, responda:
   a) Indique, pelo menos, três vantagens que a implementação com filtros ativos apresenta em relação à implementação passiva.
   b) Indique, pelo menos, duas áreas de aplicação em que a implementação passiva é desejável ou obrigatória.

a. • Filtros ativos possibilitam ganho superior a 0dB
   • Filtros ativos podem ser implementados de forma a não gerar acoplamento, eliminando a necessidade de estimar o efeito de carga de estágios subsequentes
   • Filtros ativos possuem maior diversidade de topologias
   • Filtros ativos podem dispensar o uso de indutores, o que os torna atrativos para aplicação em CHIPs

b. • Filtros passivos são necessários em áreas de altíssima frequência
   • Filtros passivos são utilizados em aplicações de alta potência

8) Faça o projeto de um filtro passa-faixa Chebyshev tipo II com as seguintes especificações: $f_0 = 10\,kHz$, $Bp = 1\,kHz$, $Bs = 9\,kHz$, $\alpha_p = 2\,dB$, $\alpha_s = 40\,dB$. Apresente o seu projeto da seguinte forma:
   a) Utilize o Matlab® para determinar e reportar a mínima ordem N que atenda as especificações.

a. $f_0 = 10kHz = \sqrt{f_1 \cdot f_2}$

$Bp = 1kHz = f_2 - f_1$

Logo:

$(10^4)^2 = f_1 \cdot (1k + f_1)$

$10^8 = f_1 \cdot 1k + f_1^2$

$f_1^2 + 10^3 f_1 - 10^8 = 0$

Utilizando a função roots:

$f_1' = 10^4 \cdot 0.9512 = 9.512\,kHz$ // ✓
$f_1'' = 10^4 \cdot (-1.0512) = -10.512\,kHz$ ✗
$\therefore f_2 = 9.512k + 1k = 10.512\,kHz$ //

Banda Passante: 9.512kHz até 10.512kHz //

$B_s = 9\,kHz$, então:

$(10^4)^2 = f_{1s} \cdot (9k + f_{1s})$
$10^8 = 9k f_{1s} + f_{1s}^2$
$f_{1s}^2 + 9k f_{1s} - 10^8 = 0$

Utilizando a função roots:

$f_{1s}' = 6.466\,kHz$ ✓
$f_{1s}'' = -15.466\,kHz$ ✗
$\therefore f_{2s} = 6.466k + 9k = 15.466\,kHz$ //

Banda de Rejeição: 6.466kHz até 15.466kHz //

Parâmetros do Matlab:
$R_p = 2$
$R_s = 40$
$W_0 = 10k \cdot 2\pi$
$W_p = [2\pi f_1 \;\; 2\pi f_2]$
$W_s = [2\pi f_{1s} \;\; 2\pi f_{2s}]$ //

b) Utilize o Matlab® para obter a função de transferência na forma original e em expansão em termos de 1ª e 2ª ordem.

c) Implemente o circuito em um simulador SPICE. Utilize um fator de escala de impedância $k_Z = 1000$.

Apresente, como resposta, o código do programa, o diagrama do circuito e os gráficos de resposta de magnitude $|H(j\omega)|$ e fase $\theta(\omega)$. Corrija eventuais discrepâncias.

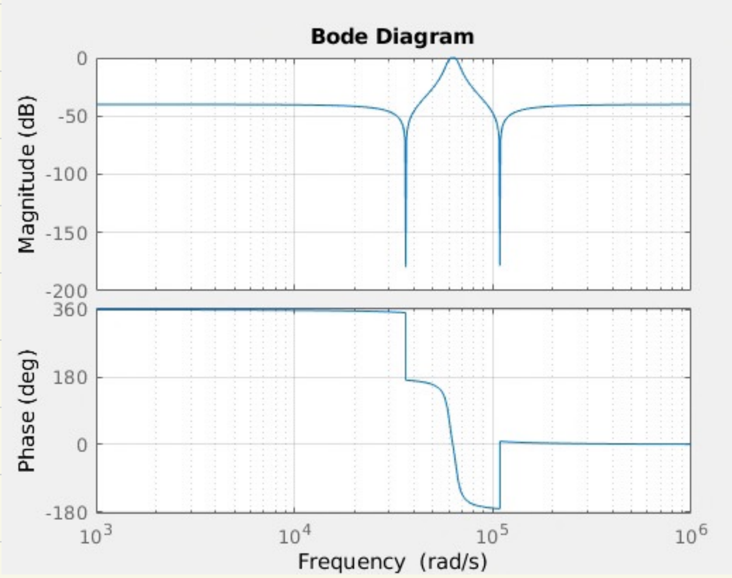
b. Código:

```
% Questão 8

%% Parâmetros do filtro
W0 = 10e3 * 2 * pi; % Rad/s
Wp1 = 9.512e3 * 2 * pi; % Rad/s
Wp2 = 10.512e3 * 2 * pi; % Rad/s
Ws1 = 6.466e3 * 2 * pi; % Rad/s
Ws2 = 15.466e5 * 2 * pi; % Rad/s
Wp = [Wp1 Wp2]; % Rad/s
Ws = [Ws1 Ws2]; % Rad/s
Rp = 2; % dB
Rs = 40; % dB

%% Projeto do filtro

[N,Ws] = cheb2ord(Wp,Ws,Rp,Rs,'s'); % Obter ordem
[z_cheb,p_cheb,k_cheb] = cheby2(N,Rs,Ws,'s');
[num_cheb,den_cheb] = zp2tf(z_cheb,p_cheb,k_cheb);
cheb_bp = tf(num_cheb,den_cheb);

%% Informacoes
bode(cheb_bp);
grid on

disp('Minima Ordem: ' );
display(N)

disp('Funcao de Transferencia:');
display(cheb_bp);
disp(' ');
disp('Expansao em termos de 1a e 2a ordem:');

[num_cheb1,den_cheb1] = zp2tf(z_cheb(3:4),p_cheb(3:4),sqrt(k_cheb));
[num_cheb2,den_cheb2] = zp2tf(z_cheb(1:2),p_cheb(1:2),sqrt(k_cheb));

cheb_bp_1 = tf(num_cheb1,den_cheb1);
cheb_bp_2 = tf(num_cheb2,den_cheb2);

display(cheb_bp_1);
display(cheb_bp_2);
```

Imagens:

```
Minima Ordem:

N =

     2

Funcao de Transferencia:

cheb_bp =

               0.01 s^4 + 1.31e08 s^2 + 1.558e17
  ---------------------------------------------------------
  s^4 + 1.015e04 s^3 + 7.947e09 s^2 + 4.006e13 s + 1.558e19

Continuous-time transfer function.
```

```
Expansao em termos de 1a e 2a ordem:

cheb_bp_1 =

    0.1 s^2 + 1.324e08
  ------------------------
  s^2 + 4867 s + 3.638e09

Continuous-time transfer function.


cheb_bp_2 =

    0.1 s^2 + 1.177e09
  ------------------------
  s^2 + 5281 s + 4.283e09

Continuous-time transfer function.
```

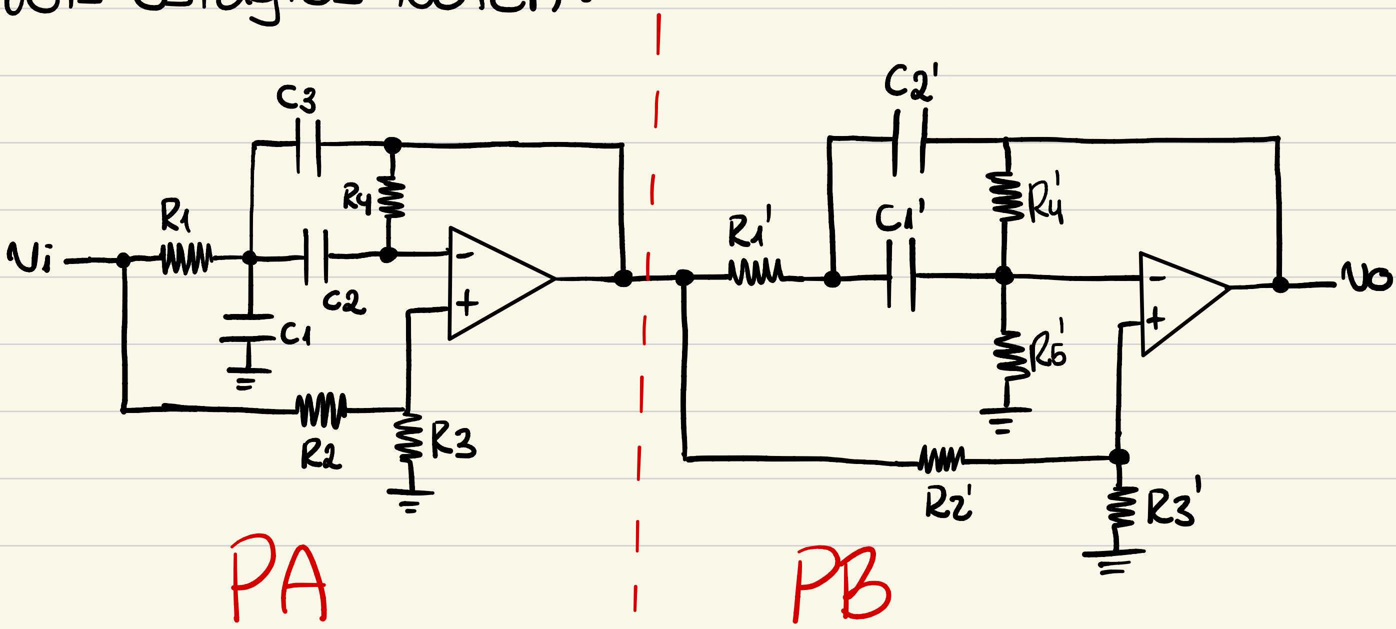


c. Primeiramente, analisam-se as funções de 1ª e 2ª ordem:

$$H_1(s) = \frac{0.1s^2 + 1.324\cdot 10^8}{s^2 + 4867s + 3.638\cdot 10^9}$$

$$H_2(s) = \frac{0.1s^2 + 1.177\cdot 10^9}{s^2 + 5281s + 4.283\cdot 10^9}$$

Para realizar o passa faixas, pode-se utilizar um PB e um PA. Para as duas FTs apresentadas, vale utilizar dois estágios Notch:



Para o Notch PA:

$$H(s) = K \cdot \frac{s^2 + \omega_z^2}{s^2 + \frac{\omega_0}{Q}s + \omega_0^2}.$$

$$\frac{0.1s^2 + 1.324 \cdot 10^8}{s^2 + 4867s + 3.638 \cdot 10^9} = \underbrace{0.1}_{K} \cdot \frac{s^2 + 1.324 \cdot 10^9}{s^2 + 4867s + 3.638 \cdot 10^9}$$

$$\omega_z = \sqrt{1.324 \cdot 10^9} = 36.3868 \cdot 10^3 //$$

$$\omega_0 = \sqrt{3.638 \cdot 10^9} = 60.3158 \cdot 10^3 //$$

$\omega_0 > \omega_z \rightarrow PA //$

$$Q = \frac{60.3158 \cdot 10^3}{4867} = 12.3928 //$$

Logo, utilizando o slide 16 da aula 6B:

$$\boxed{\begin{array}{l}
\omega_0^{norm} = 1 // \\
\omega_z^{norm} = \dfrac{36.3868 \cdot 10^3}{60.3158 \cdot 10^3} = 0.6033 // \\
K = (60.3158/36.3868)^2 - 1 = 1.7477 // \\
\\
R_1 = R_2 = 1\Omega // \\
R_3 = (2 + 1.7477) \cdot (12.3928)^2 = 575.5774\,\Omega // \\
R_4 = (2 + 1.7477)^2 \cdot (12.3928)^2 = 2.1571\,k\Omega // \\
C_1 = \dfrac{1.7477}{(2 + 1.7477) \cdot 12.3928} = 37.6299\,mF // \\
C_2 = C_3 = \dfrac{1}{(2 + 1.7477) \cdot 12.3928} = 21.5311\,mF //
\end{array}}$$

Analogamente, para o Notch PB:

$$H(s) = K \cdot \frac{s^2 + \omega_z^2}{s^2 + \frac{\omega_0}{Q} \cdot s + \omega_0^2}$$

$$\frac{0.1s^2 + 1.177 \cdot 10^9}{s^2 + 5281s + 4.283 \cdot 10^9} = 0.1 \frac{s^2 + 1.177 \cdot 10^{10}}{s^2 + 5281s + 4.283 \cdot 10^9}$$

K

$$\left.\begin{aligned} \omega_z &= \sqrt{1.177 \cdot 10^{10}} = 108.4896 \cdot 10^3 \\ \omega_0 &= \sqrt{4.283 \cdot 10^9} = 65.4446 \cdot 10^3 \end{aligned}\right\} \omega_z > \omega_0 \rightarrow PB$$

$$Q = \frac{65.4446 \cdot 10^3}{5281} = 12.3925$$

Utilizando o slide 13, da aula 6B:

$$\boxed{\begin{aligned} &\omega_0^{norm} = 1 \\ &\omega_z^{norm} = \frac{108.4896}{65.4446} = 1.6577 \\ &R_1' = R_3' = 1\,\Omega \\ &R_2' = \frac{(1.6577)^2}{2 \cdot (12.3925)^2} = 8.9467\,m\Omega \\ &R_4' = \frac{4 \cdot (12.3925)^2}{(1.6577^2 - 1)} = 351.4342\,\Omega \\ &R_5' = 4 \cdot (12.3925)^2 = 614.2962\,\Omega \\ &C_1' = C_2' = \frac{1}{2 \cdot 12.3925} = 40.3470\,mF \end{aligned}}$$

Finalmente, realiza-se os escalonamentos para implementar o filtro. Do enunciado: $K_Z = 1000$
$K_{fpa} = 60315.8$ e $K_{fpb} = 65444.6$

PA:

$R_1 = R_2 = 1 \cdot 1000 = 1k\Omega$ //

$R_3 = 575.58\,k\Omega$ //

$R_4 = 2.16\,M\Omega$ //

$$C_1 = \frac{1}{60315.8 \cdot 1000} \cdot 37.6299 = 623.88\,pF$$ //

$C_2 = C_3 = 356.97\,pF$ //

PB:

$R_1' = R_3' = 1k\Omega$ //

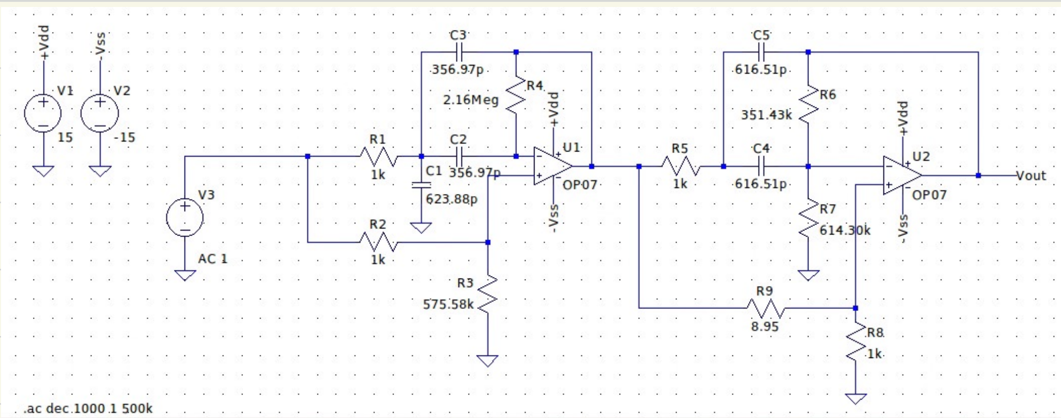
$R_2' = 8.95\,\Omega$ //

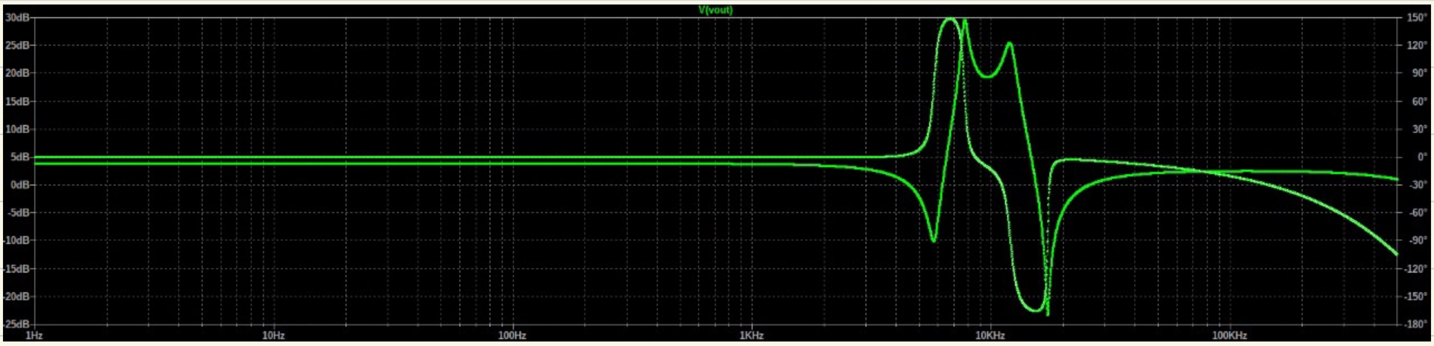
$R_4' = 351.43\,k\Omega$ //

$R_5' = 614.30\,k\Omega$ //

$$C_1' = C_2' = \frac{40.3470}{65444.6 \cdot 1000} = 616.51\,pF$$ //

Com os dados, implementa-se o circuito no LTSpice:





Percebe-se que o filtro tem o comportamento esperado aproximadamente correto, mas a simulação não foi igual ao Matlab. Provavelmente isso se deve a erros de arredondamento, mas ainda sim as frequências e a forma geral foram condizentes.