EECE5666 (DSP) : Homework-7 Solutions

Table of Contents

Default Plot Parameters

```
set(0,'defaultfigurepaperunits','points','defaultfigureunits','points');
set(0,'defaultaxesfontsize',10,'defaultaxeslinewidth',1.5);
set(0,'defaultaxestitlefontsize',1.4,'defaultaxeslabelfontsize',1.2);
```

Problem 7.1

We want to design a lowpass analog Chebyshev-I filter that has a 0.5 dB or better ripple at 10 Hz and at least 45 dB of attenuation at 20 Hz.

(a) Using the design procedure on Page 640 of the textbook (or that in Example 11.3) obtain the system function in a rational function form.

Solution: Follow the following steps. Perform numerical calculations using MATLAB below each step.

Step-0: Determine the analog passband ripple parameter ϵ and stopband attenuation parameter A.

```
Fp = 10; Fs = 20; % Band-edges in Hz
Omegap = Fp*2*pi; Omegas = Fs*2*pi; % Band-edges in radians
Ap = 0.5; As = 45; % Given log-magnitude (dB) parameters
[epsilon,A] = spec_convert(Ap,As,'rel','ana')
```

```
epsilon = 0.3493
A = 177.8279
```

Step-1: Compute the parameters α and β using (11.50):

$$\alpha = \frac{\Omega_s}{\Omega_p} = \frac{2\pi(20)}{2\pi(10)} = 2, \quad \beta = \frac{1}{\epsilon}\sqrt{A^2-1} = 509.0734.$$

```
alpha = Omegas/Omegap, beta = (1/epsilon)*sqrt(A^2-1)
```

```
alpha = 2
beta = 509.0734
```

Step-2: Compute order N using (11.49) and round upwards to the nearest integer:

$$N = \left\lceil \frac{\ln(\beta+\sqrt{\beta^2-1})}{\ln(\alpha+\sqrt{\alpha^2-1})} \right\rceil = 6.$$

```
N = ceil(log(beta+sqrt(beta^2-1))/log(alpha+sqrt(alpha^2-1)))
```

```
N = 6
```

Step-3: Set $\Omega_c = \Omega_p$ and compute a and b using (11.44) and (11.45):

$$\gamma = \left(1/\epsilon + \sqrt{1+1/\epsilon^2}\right)^{1/N} = 1.3441, \quad a = \frac{1}{2}(\gamma - \gamma^{-1}) = 0.3, \quad b = \frac{1}{2}(\gamma + \gamma^{-1}) = 1.044$$

```
Omegac = Omegap; gamma = (1/epsilon+sqrt(1+1/epsilon^2))^(1/N)
```

```
gamma = 1.3441
```

```
a = 0.5*(gamma-1/gamma), b = 0.5*(gamma+1/gamma)
```

```
a = 0.3000
b = 1.0440
```

Step-4: Compute the pole locations using (11.41) and (11.42):

$$\theta_k = \frac{\pi}{2} + \frac{2k-1}{2N}\pi;\ \sigma_k = (a\Omega_c)\cos(\theta_k);\ \Omega_k = (a\Omega_c)\sin(\theta_k);\ s_k = \sigma_k + \jmath\Omega_k; \quad k = 1, 2, \ldots, N$$

```
k = 1:N; thetak = pi/2+(2*k-1)/(2*N)*pi;
sigmak = a*Omegac*cos(thetak); Omegak = b*Omegac*sin(thetak);
sk = sigmak + 1j*Omegak; sk(1:3), sk(4:6)
```

```
ans = 1×3 complex
  -4.8789 +63.3635i -13.3294 +46.3853i -18.2083 +16.9782i
ans = 1×3 complex
 -18.2083 -16.9782i -13.3294 -46.3853i  -4.8789 -63.3635i
```

or $s_{1,6} = -4.8789 \pm \jmath 63.3635; \quad s_{2,5} = -13.3294 \pm \jmath 46.3853; \quad s_{3,4} = -18.2083 \pm \jmath 16.9782.$

Step-5: Compute the filter gain G and the system function $H(\jmath\Omega)$ from (11.43): Since N is even,

$$H(\jmath 0) = 1/\sqrt{1+\epsilon^2} = \frac{G}{(-s_1)(-s_2)(-s_3)(-s_4)(-s_5)(-s_6)} \quad \Rightarrow \quad G = \frac{s_1 s_2 s_3 s_4 s_5 s_6}{\sqrt{1+\epsilon^2}} = 5.5045 \times 10^9.$$

```
G = real(prod(sk)/sqrt(1+epsilon^2))
```

```
G = 5.5045e+09
```

Finally, the system function is given by (up to integer precision)

$$H(s) = \frac{55045(10^5)}{s^6 + 72.83s^5 + 8574.1s^4 + 394340s^3 + 18264(10^3)s^2 + 4234(10^5)s + 58306(10^5)}.$$

```
format short e; C = G, D = real(poly(sk)); D(1:4), D(5:7)
```

```
C =    5.5045e+09
ans = 1×4
    1.0000e+00    7.2833e+01    8.5741e+03    3.9434e+05
ans = 1×3
    1.8264e+07    4.2340e+08    5.8306e+09
```

(b) Verify your design using the **cheb1ord** and **cheby1** functions.

Solution: The design is obtained using the following script.

```
[N,Omegac] = cheb1ord(Omegap,Omegas,Ap,As,'s')
```

```
N =      6
Omegac =    6.2832e+01
```

```
format short e; [C,D] = cheby1(N,Ap,Omegac,'s'); C(end)
```

```
ans =    5.5045e+09
```

```
D(1:4), D(5:7)
```

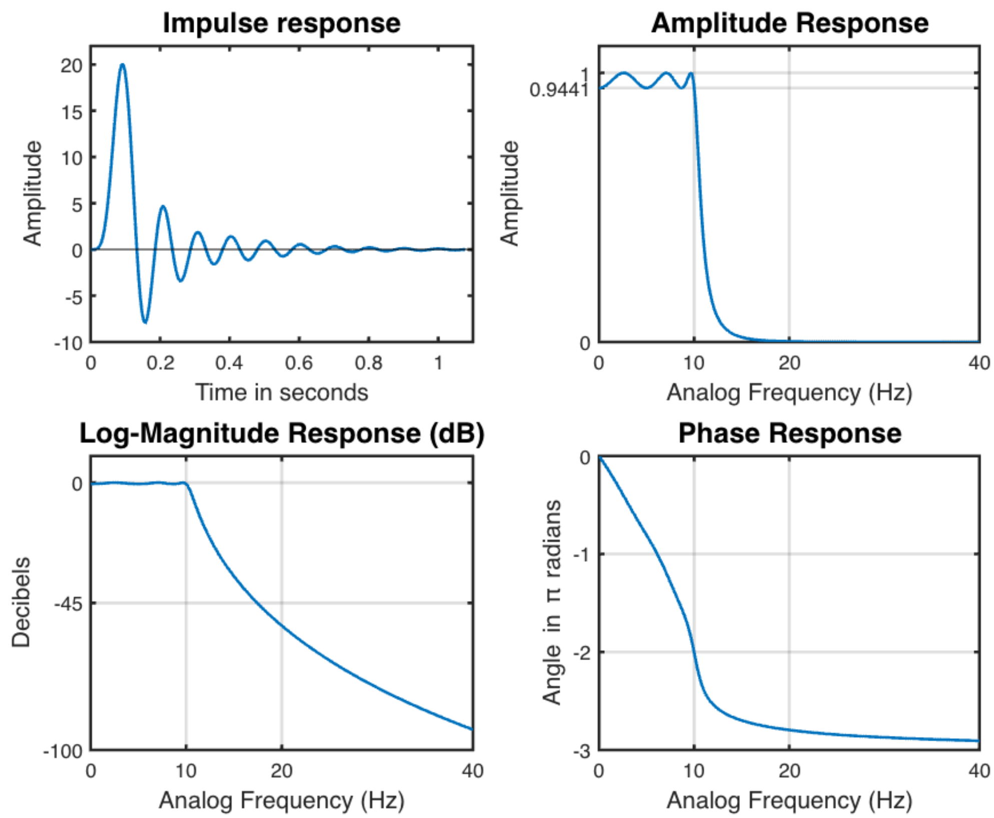
```
ans = 1×4
    1.0000e+00   7.2833e+01   8.5741e+03   3.9434e+05
ans = 1×3
    1.8264e+07   4.2340e+08   5.8306e+09
```

This design agrees with that in part (a).

(c) Provide plots of impulse response, amplitude response, log-magnitude response in dB, and phase response in one figure using two rows and two columns.

MATLAB script:

```
[hc,~,t] = impulse(C,D); % Impulse response
[Hdb, Hmag,Hpha,Omega] = freqs_m(C,D,40*2*pi); % Frequency response quantities
figure('position',[0,0,8,6]*72);
subplot(2,2,1); % Impulse response
plot(t,hc,'linewidth',1.5); axis([0,1.1,-10,22]); hold on;
plot([0,1.2],[0,0],'k','linewidth',0.5);
xlabel('Time in seconds'); ylabel('Amplitude'); title('Impulse response');
set(gca,'xtick',(0:0.2:1.2),'ytick',(-10:5:20));
subplot(2,2,2); % Amplitude Response
plot(Omega/(2*pi),Hmag,'linewidth',1.5); axis([0,40,0,1.1]);
xlabel('Analog Frequency (Hz)'); ylabel('Amplitude');
title('Amplitude Response');
set(gca,'xtick',[0,Fp,Fs,40],'ytick',[0,1/sqrt(1+epsilon^2),1]); grid;
subplot(2,2,3); % Log-magnitude response in dB
plot(Omega/(2*pi),Hdb,'linewidth',1.5); axis([0,40,-100,10]);
xlabel('Analog Frequency (Hz)'); ylabel('Decibels');
title('Log-Magnitude Response (dB)');
set(gca,'xtick',[0,Fp,Fs,40],'ytick',[-100,-As,0]); grid;
subplot(2,2,4); % Phase response
plot(Omega/(2*pi),unwrap(Hpha)/pi,'linewidth',1.5); axis([0,40,-3,0]);
xlabel('Analog Frequency (Hz)'); ylabel('Angle in \pi radians');
title('Phase Response');
set(gca,'xtick',[0,Fp,Fs,40],'ytick',(-3:1:0)); grid;
```



Problem 7.2

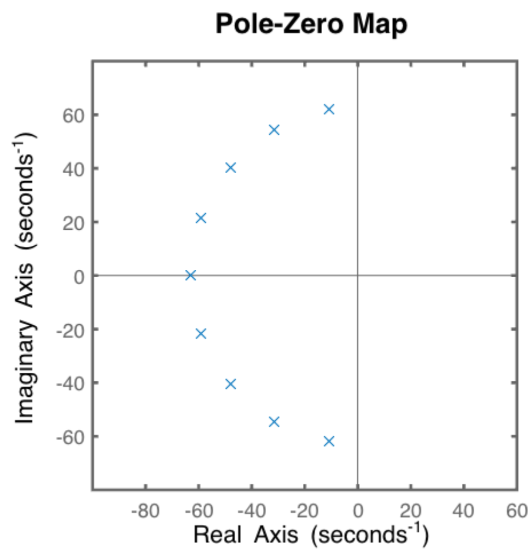
Text Problem 11.21 (Page 694)

Consider a 9^{th}-order analog Butterworth lowpass filter $H_c(s)$ with 3-dB cutoff frequency of 10 Hz.

(a) Determine and graph pole locations of $H_c(s)$.

Solution: We will use equations (11.14) and (11.15) to determine pole locations.

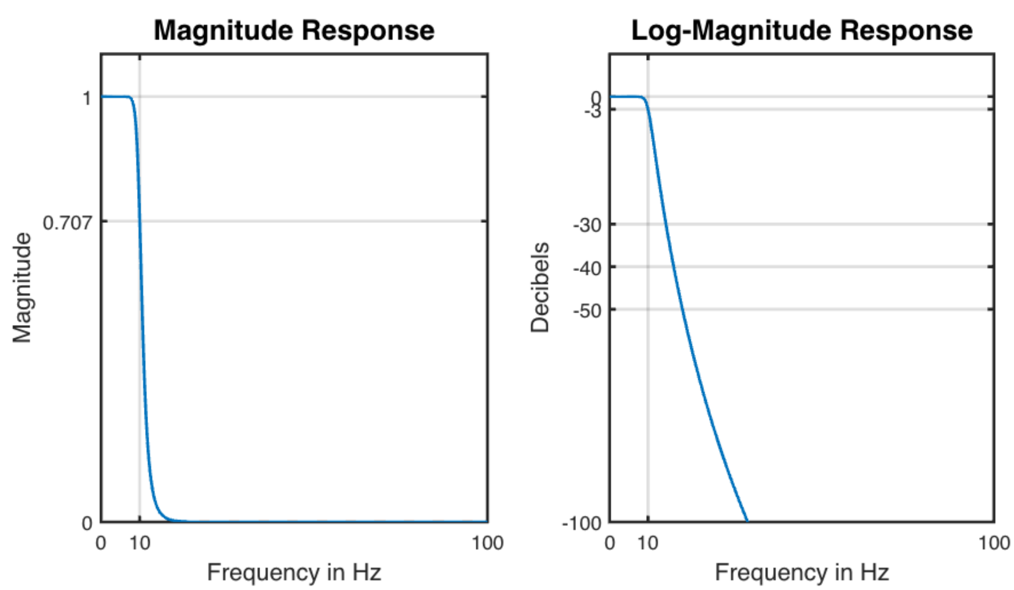
```
clc; close all; clear;
N = 9; Fc = 10; % Given order and 30dB cutoff frequency (in Hz);
Omegac = 2*pi*Fc;
k = 1:N; thetak = pi/2+(2*k-1)*pi/(2*N); % Pole angles in the LHP
sigmak = Omegac*cos(thetak); % Pole real locations
omegak = Omegac*sin(thetak); % Pole imaginary locations
sk = sigmak +1j*omegak; % s-plane pole locations
C = Omegac^N; D = real(poly(sk)); % Num and den polynomials
figure('position',[0,0,4,4]*72);
pzmap(C,D); axis([-100,60,-80,80]);
set(gca,'xtick',-80:20:60,'ytick',-60:20:60);
```



(b) Plot the magnitude and log-magnitude responses over $[0, 100]$ Hz range.

MATLAB script:

```
Fmax = 100; F = linspace(0,Fmax,1001); Om = 2*pi*F;
H = freqs(C,D,Om); Hmag = abs(H); Hdb = 20*log10(Hmag/max(Hmag));
figure('position',[0,0,8,4]*72);
subplot(1,2,1); % Magnitude plot
plot(F,Hmag,'linewidth',1.5); axis([0,Fmax,0,1.1]);
xlabel('Frequency in Hz'); ylabel('Magnitude'); title('Magnitude Response');
set(gca,'xtick',[0,10,100],'ytick',[0,0.707,1]); grid;
subplot(1,2,2); % Log-magnitude plot
plot(F,Hdb,'linewidth',1.5); axis([0,Fmax,-100,10]);
xlabel('Frequency in Hz'); ylabel('Decibels');
title('Log-Magnitude Response');
set(gca,'xtick',[0,10,100],'ytick',[-100,-50,-40,-30,-3,0]); grid;
```



(c) Determine frequencies at which the attenuation is 30 dB, 40 dB, and 50 dB.

Solution: We use the fact that the magnitude-squared response of a Butterworth filter is given by

$$|H_B(j\Omega)|^2 = \frac{1}{1+(\Omega/\Omega_c)^{2N}} \quad \Rightarrow \quad \Omega = \Omega_c \sqrt[2N]{\frac{1}{|H_B(j\Omega)|^2} - 1} \quad \text{or} \quad F = F_c \sqrt[2N]{\frac{1}{|H_B(j\Omega)|^2} - 1}.$$

Thus, for each given attenuation, we will first compute the magnitude-squared value and then use the above formula.

```
Hdb = [30,40,50]; Hmagsq = 10.^(-Hdb/10);
Fdb = Fc*(1./Hmagsq-1).^(1/(2*N));
fprintf('30 db frequency: %5.2f Hz \n',Fdb(1));
```

```
30 db frequency: 14.68 Hz
```

```
fprintf('40 db frequency: %5.2f Hz \n',Fdb(2));
```

```
40 db frequency: 16.68 Hz
```

```
fprintf('50 db frequency: %5.2f Hz \n',Fdb(3));
```

```
50 db frequency: 18.96 Hz
```

Thus the respective frequencies are 14.68 Hz, 16.68 Hz, and 18.96 Hz.

Problem 7.3

Text Problem 11.31 (Page 695)

A lowpass digital filter's specifications are given by:

$$\omega_p = 0.4\pi, \quad A_p = 0.5\text{dB}, \quad \omega_s = 0.55\pi, \quad A_s = 50\text{dB}.$$

(a) Using bilinear transformation and Chebyshev-I approximation approach obtain a system function $H(z)$ in the rational function form that satisfies the above specifications.

Solution: We will make use of the **bilinear**, **cheb1ord**, and **cheby1** functions for this design.

```
clc; close all; clear;
% Given Specifications
omegap = 0.4*pi; omegas = 0.55*pi; Ap = 0.5; As = 50;
% Step-1: Choose Td
Td = 2;
% Step-2: Obtain analog band-edge frequencies
Omegap = (2/Td)*tan(omegap/2); Omegas = (2/Td)*tan(omegas/2);
% Step-3: Design analog Chebyshev-I approximation
[N,Omegac] = cheb1ord(Omegap,Omegas,Ap,As,'s'); N
```

```
N =      8
```

```
[C,D] = cheby1(N,Ap,Omegac,'s');
% Step-4: Obtain digital Chebyshev-I filter
[b,a] = bilinear(C,D,1/Td); format short; display(b'*1e4),
```

```
    3.4709
   27.7672
   97.1852
  194.3704
  242.9630
  194.3704
```

```
   97.1852
   27.7672
    3.4709
```

```
display(a')
```

```
    1.0000
   -3.8656
    8.2625
  -11.6939
   11.7756
   -8.5442
    4.3559
   -1.4343
    0.2381
```

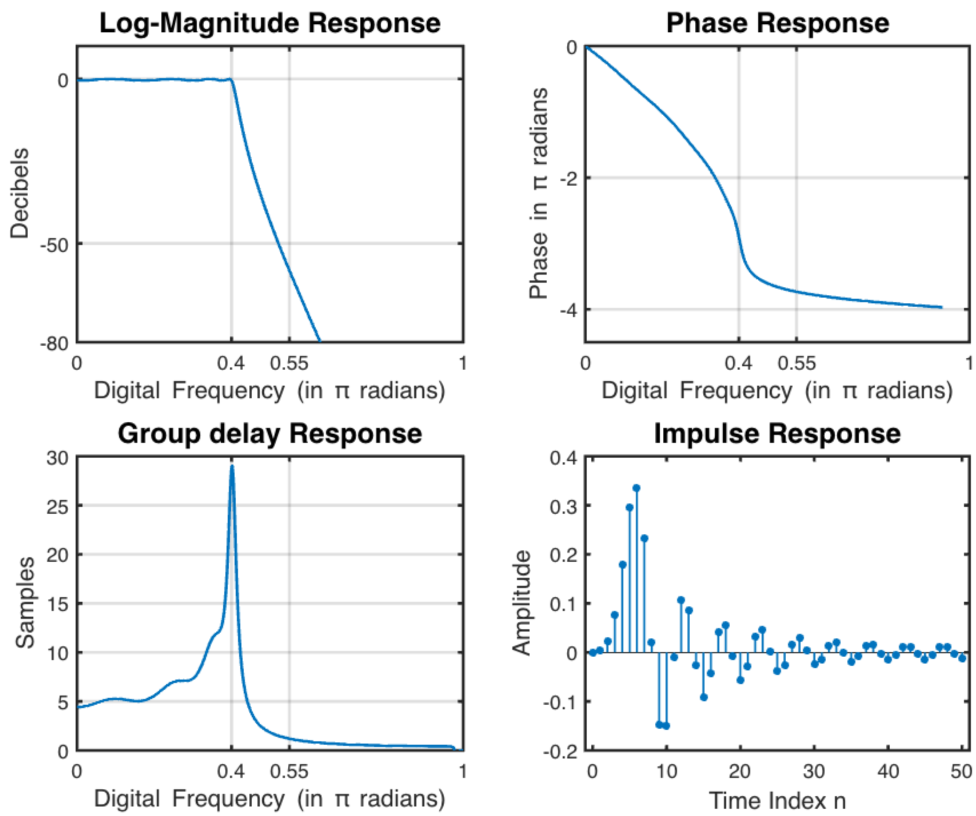
Hence the digital filter system function is an 8th-order rational function

$$H(z)=\frac{10^{-4}\left(3.47+27.77z^{-1}+97.185z^{-2}+194.37z^{-3}+243z^{-4}+194.37z^{-5}+97.185z^{-6}+27.77z^{-7}+3.47z^{-8}\right)}{1-3.87z^{-1}+8.26z^{-2}-11.69z^{-3}+11.78z^{-4}-8.54z^{-5}+4.36z^{-6}+-1.43z^{-7}+0.2381z^{-8}}.$$

(b) Provide design plots in the form of log-magnitude, phase delay, group-delay, and impulse responses.

MATLAB script:

```
f = linspace(0,1,1001); [A,~,Hpha] = zerophase(b,a,f*pi);
Hdb = 20*log10(abs(A)); Hgd = grpdelay(b,a,f*pi);
n = 0:50; hn = filter(b,a,[1,zeros(1,50)]);
figure('position',[0,0,8,6]*72);
subplot(2,2,1); % Log-magnitude response
plot(f,Hdb,'linewidth',1.5); axis([0,1,-80,10]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Decibels');
title('Log-Magnitude Response');
set(gca,'xtick',[0,omegap,omegas,pi]/pi,'ytick',[-80,-50,0]); grid;
subplot(2,2,2); % Phase delay response
plot(f,Hpha/pi,'linewidth',1.5); axis([0,1,-4.5,0]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Phase in \pi radians');
title('Phase Response');
set(gca,'xtick',[0,omegap,omegas,pi]/pi,'ytick',(-4:2:0)); grid;
subplot(2,2,3); % group delay response
plot(f,Hgd,'linewidth',1.5); axis([0,1,0,30]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Samples');
title('Group delay Response');
set(gca,'xtick',[0,omegap,omegas,pi]/pi,'ytick',(0:5:30)); grid;
subplot(2,2,4); % Impulse response
stem(n,hn,'filled','markersize',3,'linewidth',1);
axis([-1,51,-0.2,0.4]); title('Impulse Response');
xlabel('Time Index n'); ylabel('Amplitude');
```



(c) Determine the exact band-edge frequencies for the given specifications.

Solution: To determine these frequencies, we will search for the locations of $A_p = 0.5$ dB and $A_s = 50$ dB values to the nearest frequency index using the **find** function.

```
indp = find(Hdb < -Ap,1,'first'); fp_exact = f(indp-1);
inds = find(Hdb < -As,1,'first'); fs_exact = f(inds);
fprintf('Exact Passband edge = %4.3f; Exact stopband edge = %4.3f',...
    fp_exact,fs_exact);
```

```
Exact Passband edge = 0.400; Exact stopband edge = 0.522
```

Thus, the exact passband edge is $\omega_{\text{p,exact}} = 0.4\pi$ radians as expected but the exact stopband edge is $\omega_{\text{s,exact}} = 0.522\pi$ radians. The passband specifications are met while those in stopband are exceeded.

Problem 7.4

Text Problem 11.38

A highpass filter specifications are given by:

$$\omega_s = 0.6\pi, \quad A_s = 40\text{dB}, \quad \omega_0 = 0.8\pi, \quad A_p = 1\text{dB}.$$

(a) Using the Butterworth approximation obtain a system function $H(z)$ in the cascade function form that satisfies the above specifications.

Solution: We will use the **buttord** and **butter** function to obtain this design.

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

```
% Given Specifications
fs = 0.6; As = 40; fp = 0.8; Ap = 1;
[N,fc] = buttord(fp,fs,Ap,As)
```

```
N = 7
fc = 0.7709
```

```
[b,a] = butter(N,fc,'high');
[sos,G] = tf2sos(b,a)
```

```
sos = 4×6
    1.0000   -0.9922         0    1.0000    0.4532         0
    1.0000   -2.0144    1.0144    1.0000    0.9434    0.2547
    1.0000   -2.0034    1.0034    1.0000    1.0657    0.4174
    1.0000   -1.9901    0.9901    1.0000    1.3114    0.7441
G = 2.0235e-04
```

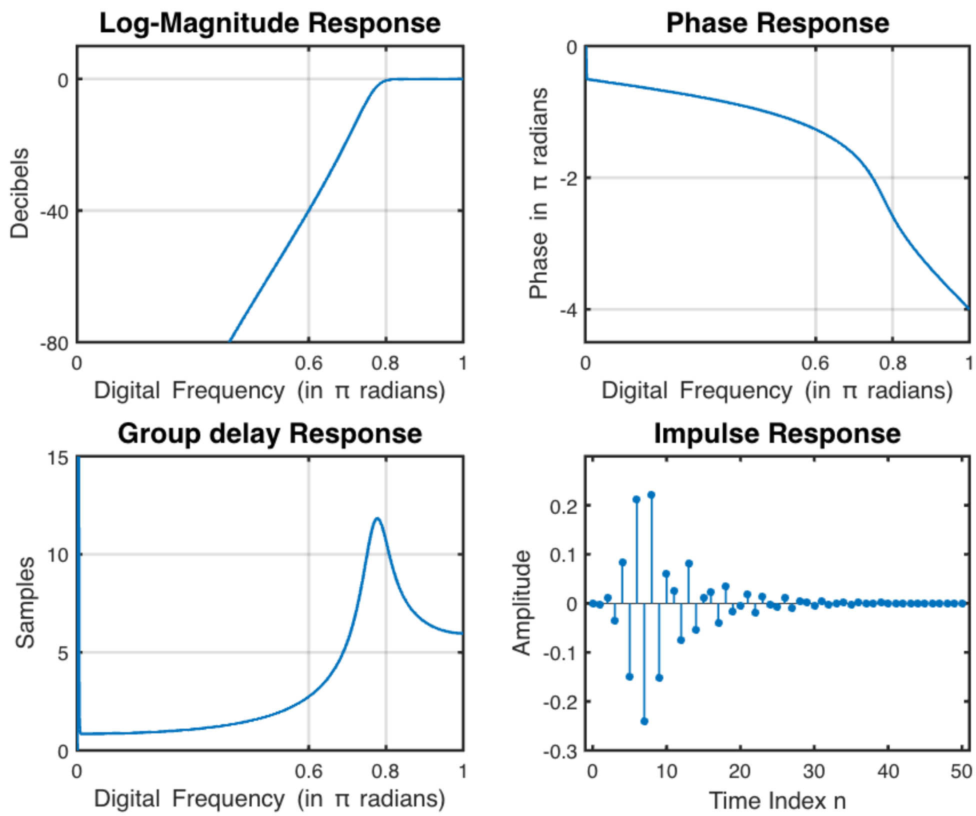
Hence the system function $H(z)$ of the designed digital filter is

$$H(z) = 2.0234(10^{-4})\left(\frac{1-0.9922z^{-1}}{1+0.4532z^{-1}}\right)\left(\frac{1-2.0144z^{-1}+1.0144z^{-2}}{1+0.9434z^{-1}+0.2547z^{-2}}\right)\left(\frac{1-2.0034z^{-1}+1.0034z^{-2}}{1+1.0657z^{-1}+0.4174z^{-2}}\right)\times$$
$$\left(\frac{1-1.9901z^{-1}+0.9901z^{-2}}{1+1.3114z^{-1}+0.7441z^{-2}}\right).$$

(b) Provide design plots in the form of log-magnitude, phase delay, group-delay, and impulse responses.

Solution: The following script plots various design plots.

```
f = linspace(0,1,1001); [A,~,Hpha] = zerophase(sos,f*pi);
Hdb = 20*log10(abs(A)/max(A)); Hgd = grpdelay(b,a,f*pi);
n = 0:50; hn = filter(b,a,[1,zeros(1,50)]);
figure('position',[0,0,8,6]*72);
subplot(2,2,1); % Log-magnitude response
plot(f,Hdb,'linewidth',1.5); axis([0,1,-80,10]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Decibels');
title('Log-Magnitude Response');
set(gca,'xtick',[0,fs,fp,1],'ytick',[-80,-40,0]); grid;
subplot(2,2,2); % Phase delay response
plot(f,Hpha/pi,'linewidth',1.5); axis([0,1,-4.5,0]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Phase in \pi radians');
title('Phase Response');
set(gca,'xtick',[0,fs,fp,1],'ytick',(-4:2:0)); grid;
subplot(2,2,3); % group delay response
plot(f,Hgd,'linewidth',1.5); axis([0,1,0,15]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Samples');
title('Group delay Response');
set(gca,'xtick',[0,fs,fp,1],'ytick',(0:5:15)); grid;
subplot(2,2,4); % Impulse response
stem(n,hn,'filled','markersize',3,'linewidth',1);
axis([-1,51,-0.3,0.3]); title('Impulse Response');
xlabel('Time Index n'); ylabel('Amplitude');
```



(c) Determine the exact band-edge frequencies for the given specifications.

Solution: To determine these frequencies, we will search for the locations of $A_\mathrm{p} = 0.5$ dB and $A_\mathrm{s} = 50$ dB values to the nearest frequency index using the **find** function.

```
indp = find(Hdb < -Ap,1,'last'); fp_exact = f(indp);
inds = find(Hdb < -As,1,'last'); fs_exact = f(inds);
fprintf('Exact stopband edge = %4.3f; Exact passband edge = %4.3f',...
    fs_exact,fp_exact);
```

```
Exact stopband edge = 0.600; Exact passband edge = 0.790
```

Thus, the exact stopband edge is $\omega_{\mathrm{s,exact}} = 0.6\pi$ radians as expected but the exact passband edge is $\omega_{\mathrm{p,exact}} = 0.79\pi$ radians. The stopband specifications are met while those in passband are exceeded.

Problem 7.5

Text Problem 11.43 (Page 698)

A digital filter is specified by the following band parameters:

$$\begin{aligned} &\text{Band-1:}[0, 0.3\pi], && \text{Attn.} = 50\text{dB}, \\ &\text{Band-2:}[0.4\pi, 0.5\pi], && \text{Attn.} = 1\text{dB}, \\ &\text{Band-3:}[0.6\pi, \pi], && \text{Attn.} = 50\text{dB}. \end{aligned}$$

(a) Using Chebyshev II approximation, obtain a system function $H(z)$ in the rational function form that satisfies the above specifications.

Solution: From the given specifications, the required filter is a bandpass filter. We will use the **cheb2ord** and **cheby2** functions to obtain the needed design.

```
clc; close all; clear;
% Given Specifications
fs = [0.3,0.6]; fp = [0.4,0.5]; Ap = 1; As = 50;
% Calculation of Chebyshev-II Filter parameters
[N,fc] = cheb2ord(fp,fs,Ap,As); N
```

```
N = 4
```

```
% Digital Chebyshev-II Bandpass Filter Design
[b,a] = cheby2(N,As,fc); format short; b'
```

```
ans = 9×1
    0.0068
   -0.0054
    0.0053
   -0.0055
    0.0114
   -0.0055
    0.0053
   -0.0054
    0.0068
```

```
a'
```

```
ans = 9×1
    1.0000
   -1.2238
    3.5354
   -2.8788
    4.2798
   -2.2237
    2.1128
   -0.5604
    0.3536
```

Thus, the resulting bandstop filter is an 8th-order filter with system function

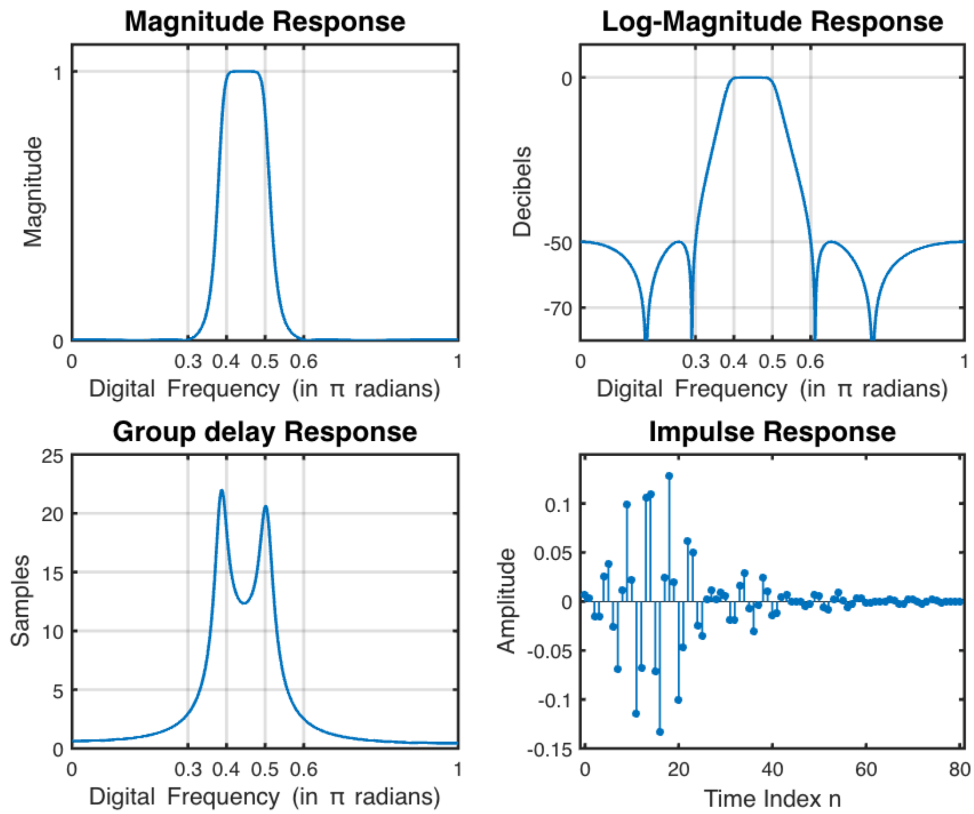
$$H(z) = \frac{0.0068 - 0.0054z^{-1} + 0.0053z^{-2} - 0.0055z^{-3} + 0.0114z^{-4} - 0.0055z^{-5} + 0.0053z^{-6} - 0.0054z^{-7} + 0.0068z^{-8}}{1 - 1.2238z^{-1} + 3.5354z^{-2} - 2.8788z^{-3} + 4.2798z^{-4} - 2.2237z^{-5} + 2.1128z^{-6} - 0.5604z^{-7} + 0.3536z^{-8}}.$$

(b) Provide design plots in the form of magnitude, log-magnitude, group-delay, and impulse responses.

MATLAB script:

```
f = linspace(0,1,1001); [A,~,~] = zerophase(b,a,f*pi);
Hdb = 20*log10(abs(A)); Hgd = grpdelay(b,a,f*pi);
n = 0:80; hn = filter(b,a,[1,zeros(1,80)]);
fcutoff = [0,fs(1),fp,fs(2),1];
figure('position',[0,0,8,6]*72);
subplot(2,2,1); % Magnitude response
plot(f,abs(A),'linewidth',1.5); axis([0,1,0,1.1]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Magnitude');
title('Magnitude Response'); set(gca,'xtick',fcutoff,'ytick',[0,1]); grid;
subplot(2,2,2); % Log-Magnitude response
plot(f,Hdb,'linewidth',1.5); axis([0,1,-80,10]);
```

```
xlabel('Digital Frequency (in \pi radians)'); ylabel('Decibels'); grid;
title('Log-Magnitude Response'); set(gca,'xtick',fcutoff,'ytick',[-70,-50,0]);
subplot(2,2,3); % group delay response
plot(f,Hgd,'linewidth',1.5); axis([0,1,0,25]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Samples');
title('Group delay Response'); set(gca,'xtick',fcutoff,'ytick',(0:5:30)); grid;
subplot(2,2,4); % Impulse response
stem(n,hn,'filled','markersize',3,'linewidth',1); axis([-1,81,-0.15,0.15]);
title('Impulse Response'); xlabel('Time Index n'); ylabel('Amplitude');
```



(c) Determine the exact band-edge frequencies for the given attenuation.

Solution: To determine these frequencies, we will search for the locations of $A_p = 0.5$ dB and $A_s = 50$ dB values to the nearest frequency index using the **`find`** function.

```
inds1 = find(round(Hdb) > -As,1,'first'); fs1_exact = f(inds1-1);
indp1 = find(Hdb > -Ap,1,'first'); fp1_exact = f(indp1+1);
indp2 = find((Hdb) > -Ap,1,'last'); fp2_exact = f(indp2-1);
inds2 = find(round(Hdb) > -As,1,'last'); fs2_exact = f(inds2+1);
fprintf('Exact band-edge frequencies are: Fs1 = %4.3f, Fp1 = %4.3f, Fp2 = %4.3f, Fs2 =
%4.3f',...
    fs1_exact,fp1_exact,fp2_exact,fs2_exact);
```

```
Exact band-edge frequencies are: Fs1 = 0.300, Fp1 = 0.393, Fp2 = 0.496, Fs2 = 0.600
```

Thus, the exact stopband edges are 0.3π and 0.6π radians as expected but the exact passband edges are 0.393π and 0.496π radians. The stopband specifications are met while those in the passband are exceeded.

Problem 7.6

Text Problem 11.66 (Page 701)

A digital filter is specified by the following band parameters:

$$\begin{aligned} &\text{Band-1:}[0, 0.2\pi], && \text{Attn.} = 1\text{dB}, \\ &\text{Band-2:}[0.35\pi, 0.5\pi], && \text{Attn.} = 50\text{dB}, \\ &\text{Band-3:}[0.65\pi, \pi], && \text{Attn.} = 1\text{dB}. \end{aligned}$$

(a) Using Butterworth approximation, obtain a system function $H(z)$ in the cascade form that satisfies the above specifications.

Solution: From the given specifications, the required filter is a bandstop filter. We will use the **`buttord`** and **`butter`** functions to obtain the needed design.

```
clc; close all; clear;
% Given Specifications
fp = [0.2,0.65]; fs = [0.35,0.5]; Ap = 1; As = 50;
% Calculation of Butterworth Filter parameters
[N,fc] = buttord(fp,fs,Ap,As); N
```

```
N = 6
```

```
% Digital Butterworth Bandstop Filter Design
[b,a] = butter(N,fc,'stop'); [sos,G] = tf2sos(b,a);
format short; display(round(sos,4));
```

```
    1.0000   -0.4818    1.0059    1.0000   -0.0441    0.2424
    1.0000   -0.4859    1.0028    1.0000   -0.5481    0.2885
    1.0000   -0.4760    1.0031    1.0000    0.3322    0.4315
    1.0000   -0.4786    0.9941    1.0000   -0.9404    0.5150
    1.0000   -0.4745    0.9972    1.0000    0.5928    0.7650
    1.0000   -0.4843    0.9969    1.0000   -1.2428    0.8133
```

Thus, the resulting bandstop filter is a 12th-order filter with six second-order sections. The cascaded system function is

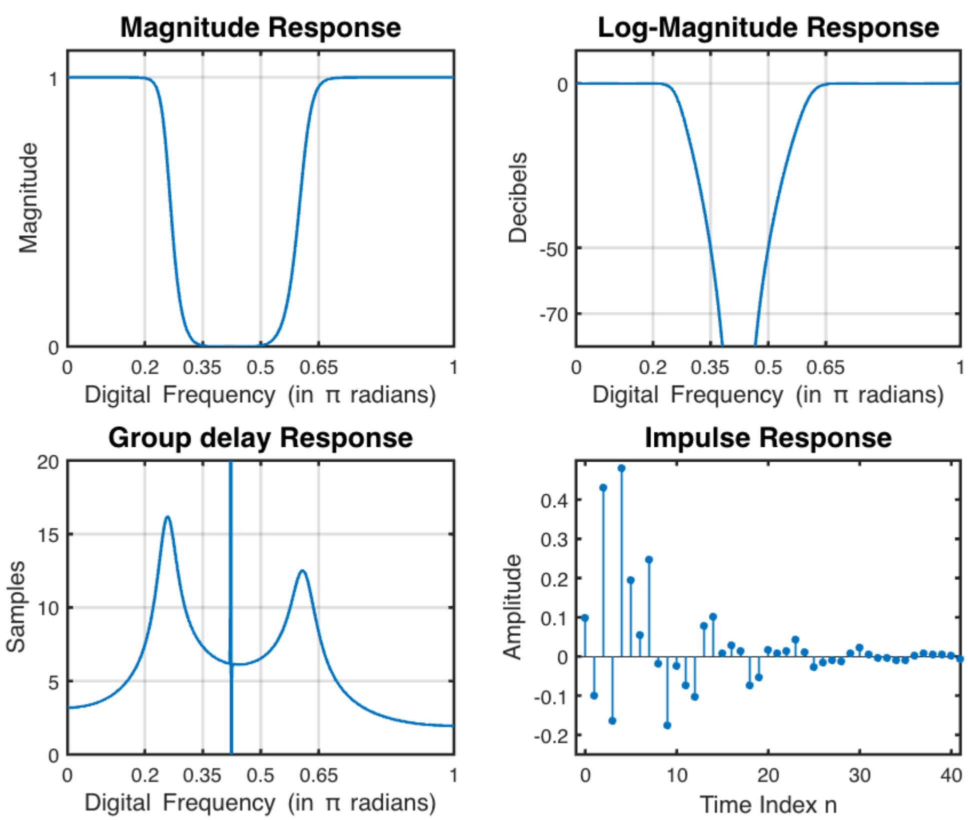
$$\begin{aligned} H(z) = 0.0982 \times &\left(\frac{1-0.4798z^{-1}+1.0047z^{-2}}{1-0.0441z^{-1}+0.2424}\right)\left(\frac{1-0.4842z^{-1}+1.0036z^{-2}}{1-0.5481z^{-1}+0.2885}\right)\left(\frac{1-0.4758z^{-1}+1.0010z^{-2}}{1+0.3322z^{-1}+0.4315}\right) \\ \times &\left(\frac{1-0.4805z^{-1}+0.9954z^{-2}}{1-0.9404z^{-1}+0.5150}\right)\left(\frac{1-0.4762z^{-1}+0.9964z^{-2}}{1+0.5928z^{-1}+0.7650}\right)\left(\frac{1-0.4846z^{-1}+0.9990z^{-2}}{1-1.2428z^{-1}+0.8133}\right). \end{aligned}$$

(b) Provide design plots in the form of magnitude, log-magnitude, group-delay, and impulse responses.

Solution: The following script plots various design plots.

```
f = linspace(0,1,1001); [A,~,~] = zerophase(b,a,f*pi);
Hdb = 20*log10(abs(A)); Hgd = grpdelay(b,a,f*pi);
n = 0:80; hn = filter(b,a,[1,zeros(1,80)]);
fcutoff = sort([0,fp,fs,1]);
figure('position',[0,0,8,6]*72); subplot(2,2,1); % Magnitude response
plot(f,abs(A),'linewidth',1.5); axis([0,1,0,1.1]);
```

```
xlabel('Digital Frequency (in \pi radians)'); ylabel('Magnitude');
title('Magnitude Response'); set(gca,'xtick',fcutoff,'ytick',[0,1]); grid;
subplot(2,2,2); % Log-Magnitude response
plot(f,Hdb,'linewidth',1.5); axis([0,1,-80,10]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Decibels');
title('Log-Magnitude Response'); set(gca,'xtick',fcutoff,'ytick',[-70,-50,0]); grid;
subplot(2,2,3); % group delay response
plot(f,Hgd,'linewidth',1.5); axis([0,1,0,20]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Samples');
title('Group delay Response'); set(gca,'xtick',fcutoff,'ytick',(0:5:30)); grid;
subplot(2,2,4); % Impulse response
stem(n,hn,'filled','markersize',3,'linewidth',1); axis([-1,41,-0.25,0.5]);
title('Impulse Response'); xlabel('Time Index n'); ylabel('Amplitude');
```



(c) Determine the exact band-edge frequencies for the given attenuation.

Solution: To determine these frequencies, we will search for the locations of $A_p = 1$dB and $A_s = 50$dB values to the nearest frequency index using the **find** function.

```
indp1 = find(Hdb < -Ap,1,'first'); fp1_exact = f(indp1);
indp2 = find(Hdb < -Ap,1,'last'); fp2_exact = f(indp2);
inds1 = find(Hdb < -As,1,'first'); fs1_exact = f(inds1);
inds2 = find(Hdb < -As,1,'last'); fs2_exact = f(inds2);
fprintf('Exact band-edge frequencies are: fp1 = %4.3f, fs1 = %4.3f, fs2 = %4.3f, fp2 =
%4.3f',...
```

```
    fp1_exact,fs1_exact,fs2_exact,fp2_exact);
```

```
Exact band-edge frequencies are: fp1 = 0.243, fs1 = 0.350, fs2 = 0.500, fp2 = 0.631
```

Thus, the exact stopband edges are 0.35π and 0.5π radians as expected but the exact passband edges are 0.243π and 0.631π radians. The stopband specifications are met while those in the passband are exceeded.

Problem 7.7

Text Problem 11.70 (Page 702)

An analog signal $x_c(t) = 5\sin(2\pi 250t) + 10\sin(2\pi 300t)$ is to be processed using the effective continuous-time system of Figure 6.18 in which the sampling frequency is 1 kHz.

(a) Design a minimum-order IIR digital filter that will suppress the 300 Hz component down to 50 dB while pass the 250 Hz component with attenuation of less than 1 dB. The digital filter should have an equiripple passband and stopband. Determine the system function of the filter and plot its log-magnitude response in dB.

Solution: In this problem, the digital filter to be designed is an elliptic prototype so that the overall effective analog filter is also elliptic. Note that the band-edge frequencies are not given. However, we desire a minimum-order filter. This means that the transition bandwidth should be as large as possible while passing the 250 Hz component and suppressing the 300 Hz component. Thus, in the effective analog filter the passband edge should be at 250 Hz and the stopband edge should be at 300 Hz. A larger passband edge and/or a smaller stopband edge will result in a smaller transition bandwidth and a possibly larger filter order. Further note that the filter must be a lowpass filter because although a bandpass or a bandstop filter can satisfy the given requirements, they will have higher order.

Through sampling the above band-edge frequencies translate into corresponding band-edge frequencies for the digital filter to be designed. Hence, we have $\omega_p = 2\pi(250/1000) = 0.5\pi$ and $\omega_s = 2\pi(300/1000) = 0.6\pi$. Now using the given $A_p = 1$dB and $As = 50$dB we design the IIR elliptic lowpass filter.

```
clc; close all; clear;
fp = 0.5; fs = 0.6; Ap = 1; As = 50; % Given specifications
[N,fn] = ellipord(fp,fs,Ap,As); N
```

```
N = 6
```

```
[b,a] = ellip(N,Ap,As,fn)
```

```
b = 1×7
    0.0481    0.1381    0.2542    0.3010    0.2542    0.1381    0.0481
a = 1×7
    1.0000   -1.1660    2.2689   -1.8296    1.5033   -0.6909    0.2401
```

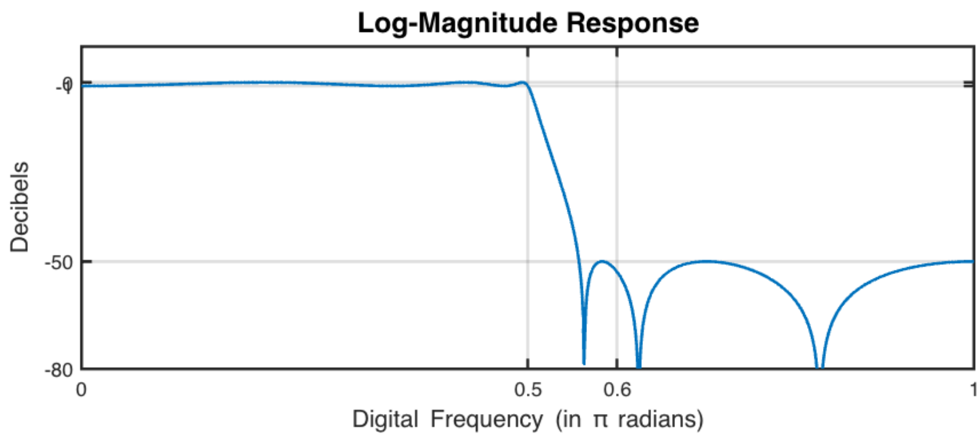
Thus we obtain a 6th-order elliptic filter with system function

$$H(z) = \frac{0.0481 + 0.1381z^{-1} + 0.2542z^{-2} + 0.3010z^{-3} + 0.2542z^{-4} + 0.1381z^{-5} + 0.0481z^{-6}}{1 - 1.1660z^{-1} + 2.2689z^{-2} - 1.8296z^{-3} + 1.5033z^{-4} - 0.6909z^{-5} + 0.2401z^{-6}}.$$

The log-magnitude response is computed using the following script.

```
f = linspace(0,1,1001); [A,~,Hpha] = zerophase(b,a,f*pi); Hdb = 20*log10(abs(A));
figure('position',[0,0,8,3]*72);
plot(f,Hdb,'linewidth',1.5); axis([0,1,-80,10]);
xlabel('Digital Frequency (in \pi radians)'); ylabel('Decibels')
```

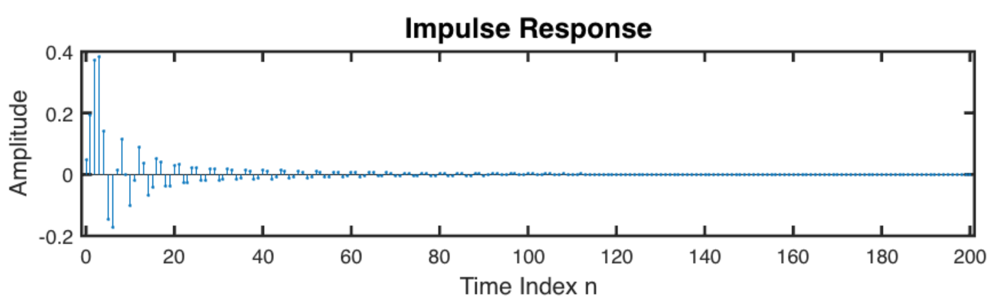
```
title('Log-Magnitude Response');
set(gca,'xtick',[0,fp,fs,1],'ytick',[-80,-50,-1,0]); grid;
```



(b) Process the signal $x_c(t)$ through the effective analog system. Generate sufficient samples so that the output response $y_c(t)$ goes into steady state. Plot the steady state $y_{ss}(t)$ and comment on the filtering result.

Solution: Since the fundamental periods of 250 and 300 Hz components are 4 and 3.333 ms we will need 80 ms of time interval to show about 20 cycles of the signal. To determine how long the transient response lasts we will determine the impulse response of the filter.

```
n = 0:200; hn = filter(b,a,[1,zeros(1,200)]);
figure('position',[0,0,8,2]*72);
stem(n,hn,'filled','markersize',1); axis([-1,201,-0.2,0.4]);
xlabel('Time Index n'); ylabel('Amplitude'); title('Impulse Response');
```

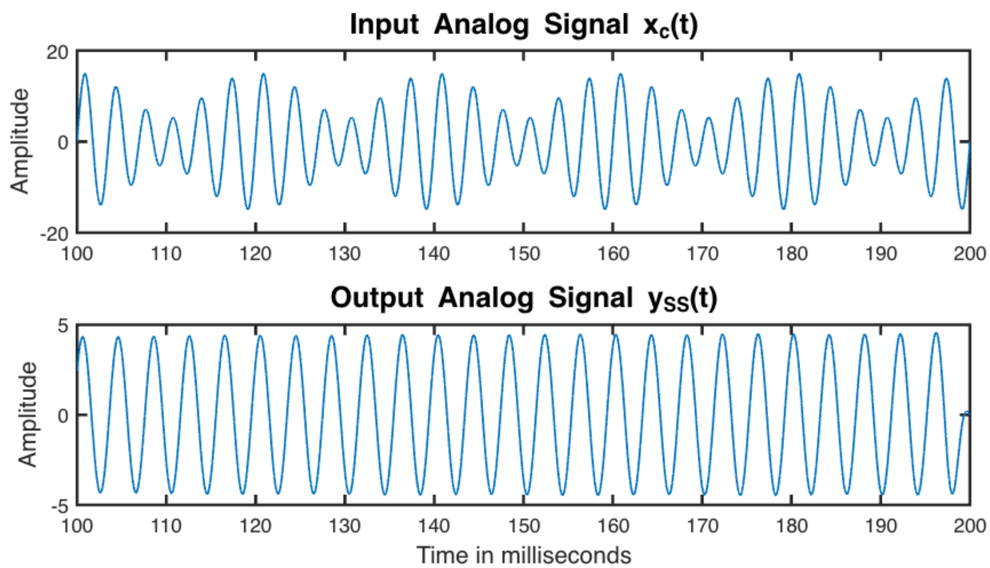


It appears that the steady state is reached in about 100 samples or in 100 ms. To show the response beyond steady state we will consider additional 100 ms. Hence, we will consider time interval of $[0, 200]$ ms.

```
t = linspace(0,200,10001)*1e-3; % Finer sampling of time for analog signal
xt = 5*sin(2*pi*250*t)+10*sin(2*pi*300*t); % Analog signal
Fs = 1000; Ts = 1/Fs; n = 0:200; nTs = n*Ts; % Sampling instances
xn = 5*sin(2*pi*250*nTs)+10*sin(2*pi*300*nTs); % Discrete-time signal
yn = filter(b,a,xn); % Filtered discrete-time signal
yt = interpft(yn,length(t));
```

Since the transient response ends (for engineering purposes) in 100 ms, we will plot over $100 \le t, \text{ms} \le 200.$

```
figure('position',[0,0,8,4]*72);
subplot(2,1,1); plot(t(5001:end)*1000,xt(5001:end),'linewidth',1);
% xlabel('Time in milliseconds');
title('Input Analog Signal x_c(t)'); ylabel('Amplitude');
subplot(2,1,2); plot(t(5001:end)*1000,yt(5001:end),'linewidth',1);
xlabel('Time in milliseconds'); ylabel('Amplitude');
title('Output Analog Signal y_{SS}(t)');
```



Comment: Observe that the output signal has 25 cycles in 100 ms which is the 250 Hz signal. Thus, the digital filter eliminated the 300 Hz component.

(c) Repeat parts (a) and (b) by designing an equiripple FIR filter. Compare the orders of the two filters and their filtering results.

Solution: The following script implements this design and implementation.

```
[deltap,deltas] = spec_convert(Ap,As,'rel','abs');
[M,fo,ao,W] = firpmord([fp,fs],[1,0],[deltap,deltas]); M
```

```
M = 34
```

```
[~,delta] = firpm(M,fo,ao,W);
fprintf('Required ripple: %g, Designed ripple: %g',deltap,delta);
```

```
Required ripple: 0.0575011, Designed ripple: 0.0548128
```

We will decrease $M \to M-1$ and check if the specifications are still satisfied.

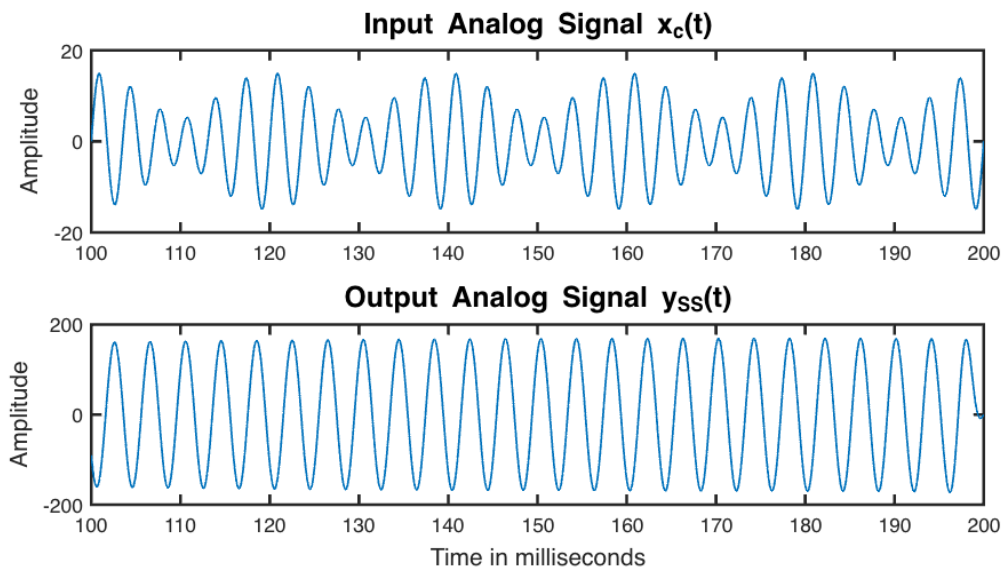
```
M = M-1; [~,delta] = firpm(M,fo,ao,W);
fprintf('Required ripple: %g, Designed ripple: %g',deltap,delta);
```

```
Required ripple: 0.0575011, Designed ripple: 0.0678677
```

So clearly, the previous value of $M = 34$ was correct.

```
M = M+1; [b,delta] = firpm(M,fo,ao,W);
yn = filter(b,a,xn); % Filtered discrete-time signal
yt = interpft(yn,length(t));
```

```
figure('position',[0,0,8,4]*72);
subplot(2,1,1); plot(t(5001:end)*1000,xt(5001:end),'linewidth',1);
% xlabel('Time in milliseconds');
title('Input Analog Signal x_c(t)'); ylabel('Amplitude');
subplot(2,1,2); plot(t(5001:end)*1000,yt(5001:end),'linewidth',1);
xlabel('Time in milliseconds'); ylabel('Amplitude');
title('Output Analog Signal y_{SS}(t)');
```



Comment: Again, we have the output signal that has 25 cycles in 100 ms which is the 250 Hz signal. Thus, the digital filter eliminated the 300 Hz component. However, time-shift from input to output is different therefore, the output signal looks different.

Problem 7.8

Text Problem 11.71 (Page 703)

Consider the following bandpass digital filter specifications:

$$\begin{aligned} \text{Stopband-1} &: [0, 0.4\pi], & \text{Attn.} &= 40 \text{ dB} \\ \text{Passband} &: [0.45\pi, 0.55\pi], & \text{Attn.} &= 0.5 \text{ dB} \\ \text{Stopband-2} &: [0.65\pi, \pi], & \text{Attn.} &= 50 \text{ dB} \end{aligned}$$

(a) Design a minimum order FIR filter to satisfy the above specifications. Plot its magnitude, log-magnitude (dB), and group-delay responses in one figure using 3 rows and 1 column.

MATLAB script: The minimum order FIR filter is obtained using the Parks-McClellan approach.

```
clc; close all; clear;
fs1 = 0.4; fp1 = 0.45; fp2 = 0.55; fs2 = 0.65; % Band-edge frequencies
As1 = 40; Ap = 0.5; As2 = 50; % Mag response specs in dB
% Convert mag-resp dB specs to absolute specs
[~,deltas1] = spec_convert(Ap,As1,'rel','abs')
```

```
deltas1 = 0.0103
```

```
[deltap,deltas2] = spec_convert(Ap,As2,'rel','abs')
```

```
deltap = 0.0288
deltas2 = 0.0033
```

```
Weights = deltap./[deltas1,deltap,deltas2], % correct weighting array
```

```
Weights = 1×3
    2.7970    1.0000    8.8448
```

```
f = [fs1,fp1,fp2,fs2]; A = [0,1,0];
[M,f0,A0,~] = firpmord(f,A,[deltas2,deltap,deltas2]);
[~,dev] = firpm(M,f0,A0,Weights); dev, deltap,M,
```

```
dev = 0.0170
deltap = 0.0288
M = 76
```

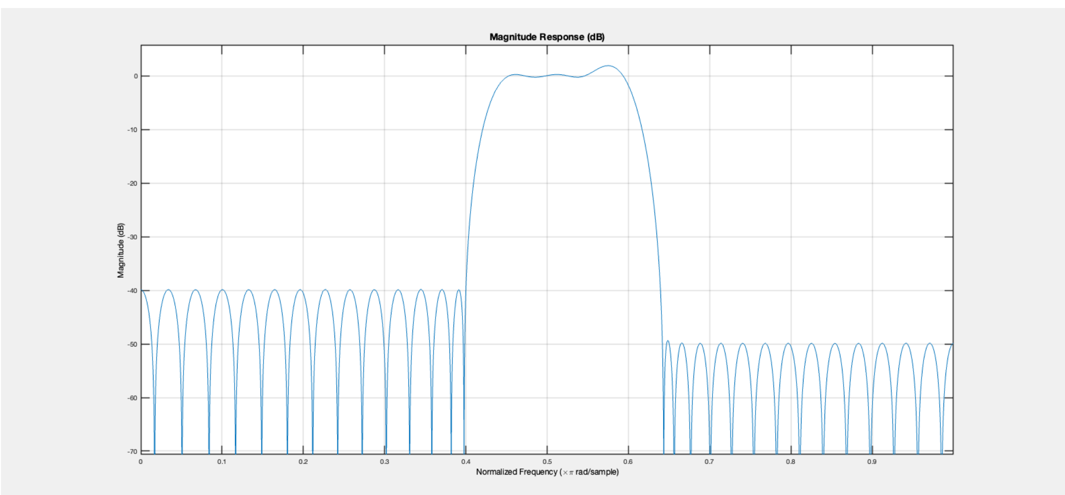
Since `dev < deltap`, we will decrease M. The optimum value (i.e., for `dev` just below `deltap`) was found to be $M = 66$.

```
M = M-10; [h,dev] = firpm(M,f0,A0,Weights); dev, deltap,M,
```

```
dev = 0.0283
deltap = 0.0288
M = 66
```

At this stage, we should check the frequency response to verify that the specifications are satisfied. A quick check can be done using the `fvtool` function.

```
fvtool(h,1);
```



The dB specifications in the passband and two stopbands are clearly satisfied. However, the transition band response is not acceptable since the response at some frequencies is more than $0\ \text{dB}$. We need a monotone decreasing response. This happened because the transition bandwidths are not equal, and the PM algorithm does not constraint the response in the transition bands. To resolve this issue, we will have to equalize the transition bandwidth to the smallest of the two. If we increase the upper passband f_{p_2} then we are letting some frequencies in the upper transition band to pass through. If we decrease f_{s_2} then we are suppressing some frequencies in the upper transition band, which is more reasonable. Thus, we set set $f_{\text{s}_2} = 0.6$ and redesigned the filter to the new specifications. The minimum order to satisfy all requirements (including monotone transition band responses) was $M = 78$ as shown below.

```
fs2new = 0.6; % New upper stopband frequency
f = [fs1,fp1,fp2,fs2new];
```

```
[M,f1,A0,~] = firpmord(f,A,[deltas2,deltap,deltas2]);
[~,dev] = firpm(M,f1,A0,Weights); dev, deltap,M,
```

```
dev = 0.0341
deltap = 0.0288
M = 76
```

```
M = M+1; [~,dev] = firpm(M,f1,A0,Weights); dev, deltap,M,
```

```
dev = 0.0292
deltap = 0.0288
M = 77
```

```
M = M+1; [~,dev] = firpm(M,f1,A0,Weights); dev, deltap,M,
```

```
dev = 0.0280
deltap = 0.0288
M = 78
```

Can we do better than this and obtain a lower order? To investigate this, we will now increase f_{s_2} in steps of 0.01 and recheck the minimum order. For $f_{s_2} = 0.61$, we obtained $M = 72$. For $f_{s_2} = 0.62$, we obtained $M = 70$. For $f_{s_2} = 0.63$, we obtained $M = 67$. Any further attempts resulted either in higher order or in non-monotone transition band responses.

```
fs2new = 0.63; % New upper stopband frequency
f = [fs1,fp1,fp2,fs2new];
[M,f1,A0,weights] = firpmord(f,A,[deltas2,deltap,deltas2]);
[~,dev] = firpm(M,f1,A0,Weights); dev, deltap,M,
```

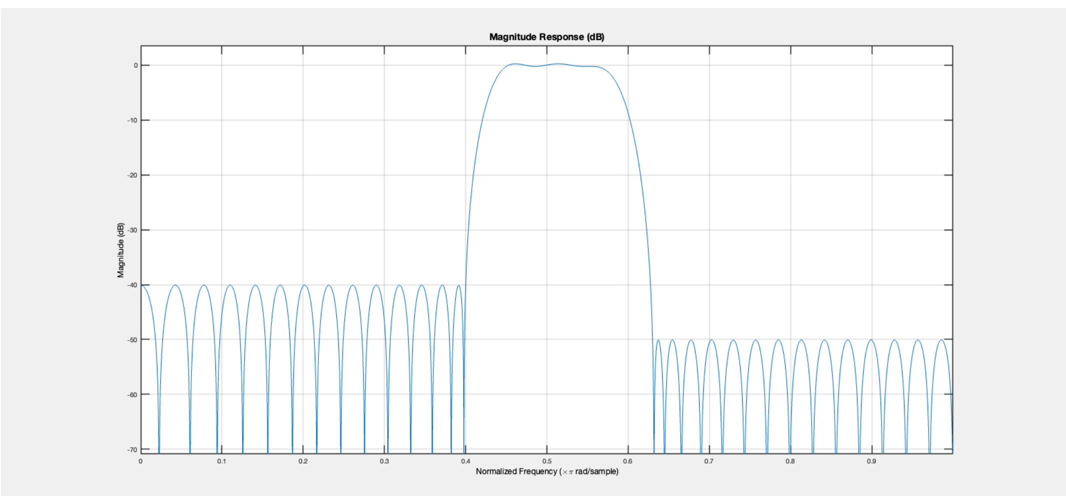
```
dev = 0.0178
deltap = 0.0288
M = 76
```

```
M = M-9; [h,dev] = firpm(M,f1,A0,Weights); dev, deltap,M,
```

```
dev = 0.0276
deltap = 0.0288
M = 67
```

We will do a quick check using the `fvtool` function.

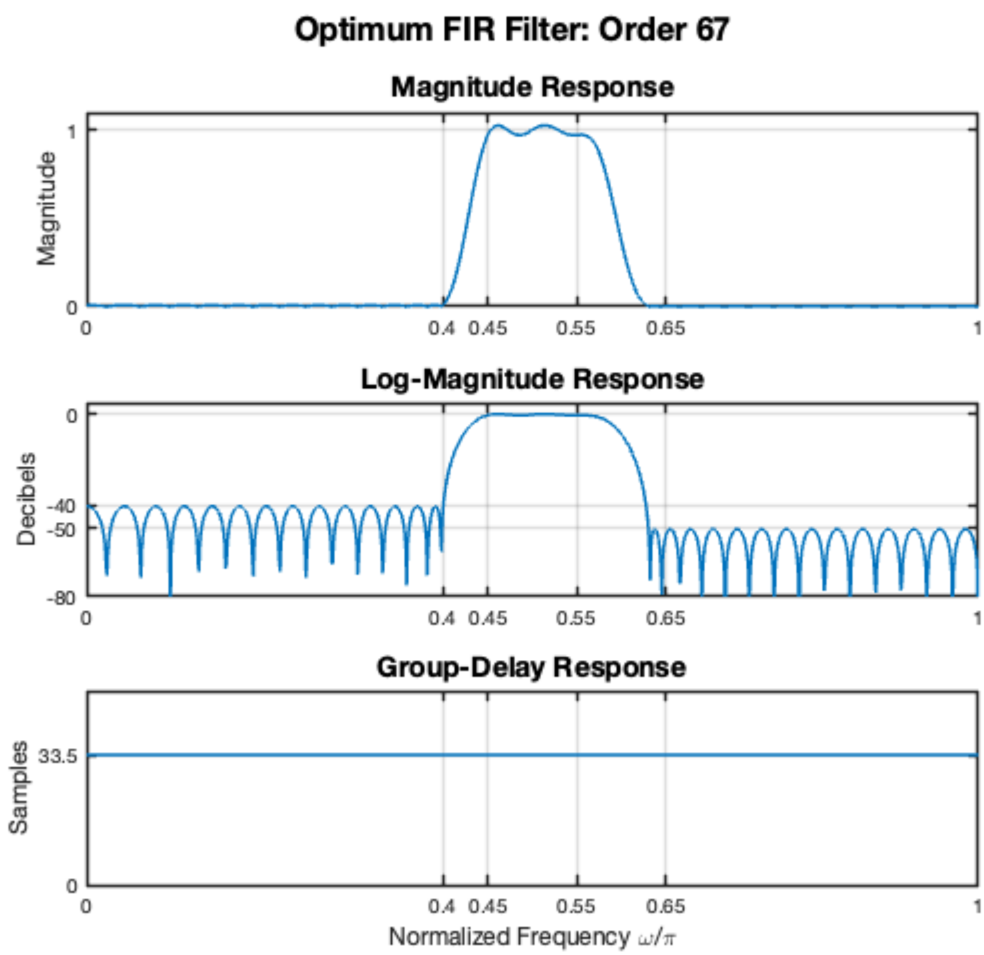
```
fvtool(h,1);
```



Clearly, all specs are satisfied, and the response is monotonically decreasing (with at least 60 dB of attenuation) in the transition band.

Design Plots:

```
om = linspace(0,1,1001)*pi; H = freqz(h,1,om);
Hmag = abs(H); HdB = 20*log10(Hmag/max(Hmag));
Hgrd = grpdelay(h,1,om);
figure('Position',[0,0,8,7]*72);
subplot(3,1,1); % Magnitude response
plot(om/pi,Hmag,'LineWidth',1.5); axis([0,1,0,1.1]);
ylabel('Magnitude'); title('Magnitude Response');
set(gca,'ytick',[0,1],'xtick',f0); grid;
subplot(3,1,2); % Log-magnitude response in dB
plot(om/pi,HdB,'LineWidth',1.5); axis([0,1,-80,5]);
ylabel('Decibels'); title('Log-Magnitude Response');
set(gca,'ytick',[-80,-50,-40,0],'xtick',f0); grid;
subplot(3,1,3); % Group delay response
plot(om/pi,Hgrd,'LineWidth',1.5); axis([0,1,0,50]);
ylabel('Samples'); title('Group-Delay Response');
xlabel('Normalized Frequency \omega/\pi');
set(gca,'ytick',[0,M/2],'xtick',f0); grid;
sgtitle('Optimum FIR Filter: Order 67','fontsize',16,'fontweight','bold');
```



(b) Design a minimum order IIR filter to satisfy the above specifications. Plot its magnitude, log-magnitude (dB), and group-delay responses in one figure using 3 rows and 1 column. From your plots determine the exact band-edge frequencies.

MATLAB script: The minimum order IIR filter is obtained via elliptic prototype. Since there are two stopband attenuations, we will design for the larger minimum stopband attenuation $A_{s_2} = 50\ \text{dB}$.

```
[N,omp] = ellipord([fp1,fp2],[fs1,fs2],Ap,As2)
```

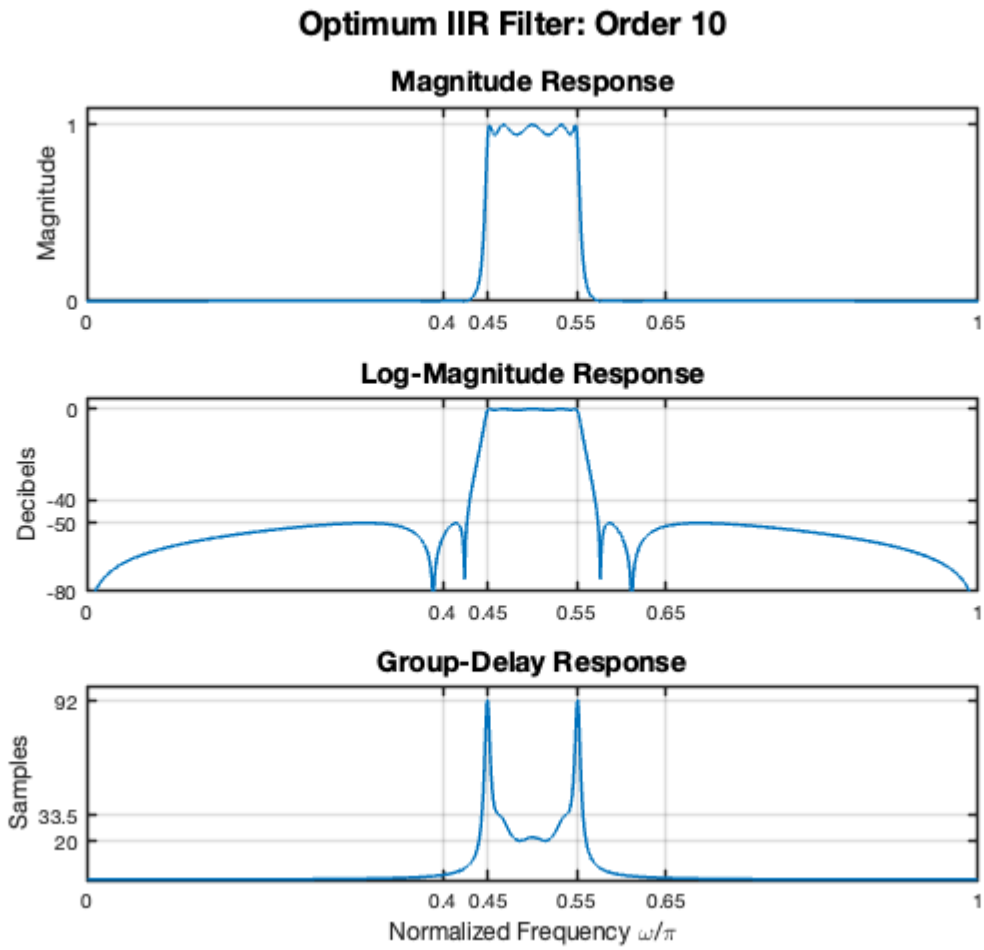
```
N = 5
omp = 1×2
    0.4500    0.5500
```

```
[b,a] = ellip(N,Ap,As2,omp);
```

Thus, the desired elliptic IIR filter has order equal to $2N = 10$, which is much smaller than the optimum FIR filter.

Design Plots:

```
om = linspace(0,1,1001)*pi; H = freqz(b,a,om);
Hmag = abs(H); HdB = 20*log10(Hmag/max(Hmag));
Hgrd = grpdelay(b,a,om);
minHgrd = round(min(Hgrd(451:550)));
maxHgrd = round(max(Hgrd(451:550)));
figure('Position',[0,0,8,7]*72);
subplot(3,1,1); % Magnitude response
plot(om/pi,Hmag,'LineWidth',1.5); axis([0,1,0,1.1]);
ylabel('Magnitude'); title('Magnitude Response');
set(gca,'ytick',[0,1],'xtick',f0); grid;
subplot(3,1,2); % Log-magnitude response in dB
plot(om/pi,HdB,'LineWidth',1.5); axis([0,1,-80,5]);
ylabel('Decibels'); title('Log-Magnitude Response');
set(gca,'ytick',[-80,-50,-40,0],'xtick',f0); grid;
subplot(3,1,3); % Group delay response
plot(om/pi,Hgrd,'LineWidth',1.5); axis([0,1,0,100]);
ylabel('Samples'); title('Group-Delay Response');
xlabel('Normalized Frequency \omega/\pi');
set(gca,'ytick',[minHgrd,M/2,maxHgrd],'xtick',f0); grid;
sgtitle('Optimum IIR Filter: Order 10','fontsize',16,'fontweight','bold');
```



(c) Compare the two filter designs in terms of their responses and orders.

Comparison: Comparing the orders and response plots of the two designs, we observe the following

1. The optimal FIR filter order is 67 while that for the IIR filter is 10.
2. The group delay of the FIR filter is constant equal to 33.5 while that for the IIR filter is highly nonlinear, ranging from a minimum of 20 samples to the maximum of 92 samples at the passband cutoff frequencies.

Additionally, steps needed in obtaining the optimal FIR filter design were substantially more compared to those for the optimum IIR filter design, which was straightforward.