Tutorial-5

IIR Filters

Objective:

Design of IIR filters using Butterworth, Chebyshev, Elliptical approximations, and direct method.

1. Design an IIR low pass filter with passband edge at 1000Hz and stopband edge at 1500Hz for a sampling frequency of 8000Hz. The filter is to have a passband ripple of 0.5 dB and a stopband ripple below 30 dB. Plot the frequency response and compare the performance of the filters designed with Butterworth, Chebyshev I, Chebyshev II and Elliptic approximations and using both impulse invariant and bilinear transformations.

```
clear; clc; close all;
Fs=8000;
wp=1000*2/Fs; ws=1500*2/Fs;
wpa=1000*2*pi; wsa=1500*2*pi;
Rp=0.5; Rs=30;
 % Butterworth filter
% 1. Bilinear transformation
[Nb1,wn_b1]=buttord(wp,ws,Rp,Rs);
[num_but1,den_but1]=butter(Nb1,wn_b1);

% 2. Impulse invariant transformation
[Nb2,wn_b2]=buttord(wpa,wsa,Rp,Rs,'s');
[num_but2a,den_but2a]=butter(Nb2,wn_b2,'s');
[num_but2,den_but2]=impinvar(num_but2a,den_but2a,Fs);

f=0:10:Fs/2;
hb1=freqz(num_but1,den_but1,f,Fs);
hb2=freqz(num_but2,den_but2,f,Fs);
plot(f,20*log10(abs(hb1)),f,20*log10(abs(hb2)));
Title('Butterworth filter');axis([0 4000 -60 1]);
figure; plot(f,abs(hb1),f,abs(hb2));
Title('Butterworth filter');

% Chebyshev I filter
% 1. Bilinear transformation
[Nc11,wn_c11]=cheb1ord(wp,ws,Rp,Rs);
[num_c11,den_c11]=cheby1(Nc11,Rp,wn_c11);

% 2. Impulse invariant transformation
[Nc12,wn_c12]=cheb1ord(wpa,wsa,Rp,Rs,'s');
[num_c12a,den_c12a]=cheby1(Nc12,Rp,wn_c12,'s');
[num_c12,den_c12]=impinvar(num_c12a,den_c12a,Fs);
```

```
f=0:10:Fs/2;
hc11=freqz(num_c11,den_c11,f,Fs);
hc12=freqz(num_c12,den_c12,f,Fs);
figure;plot(f,20*log10(abs(hc11)),f,20*log10(abs(hc12)
));
Title('Chebyshev I filter');axis([0 4000 -60 1]);
figure; plot(f,abs(hc11),f,abs(hc12));
Title('Chebyshev I filter');

% Chebyshev II filter
% 1. Bilinear transformation
[Nc21,wn_c21]=cheb2ord(wp,ws,Rp,Rs);
[num_c21,den_c21]=cheby2(Nc21,Rs,wn_c21);

% 2. Impulse invariant transformation
[Nc22,wn_c22]=cheb2ord(wpa,wsa,Rp,Rs,'s');
[num_c22a,den_c22a]=cheby2(Nc22,Rs,wn_c22,'s');
[num_c22,den_c22]=impinvar(num_c22a,den_c22a,Fs);

f=0:10:Fs/2;
hc21=freqz(num_c21,den_c21,f,Fs);
hc22=freqz(num_c22,den_c22,f,Fs);
figure;plot(f,20*log10(abs(hc21)),f,20*log10(abs(hc22)
));
Title('Chebyshev II filter');axis([0 4000 -60 1]);
figure; plot(f,abs(hc21),f,abs(hc22));
Title('Chebyshev II filter');

% Elliptic filter
% 1. Bilinear transformation
[Ne1,wn_e1]=ellipord(wp,ws,Rp,Rs);
[num_e1,den_e1]=ellip(Ne1,Rp,Rs,wn_e1);

% 2. Impulse invariant transformation
[Ne2,wn_e2]=ellipord(wpa,wsa,Rp,Rs,'s');
[num_e2a,den_e2a]=ellip(Ne2,Rp,Rs,wn_e2,'s');
[num_e2,den_e2]=impinvar(num_e2a,den_e2a,Fs);

f=0:10:Fs/2;
he1=freqz(num_e1,den_e1,f,Fs);
he2=freqz(num_e2,den_e2,f,Fs);
figure;plot(f,20*log10(abs(he1)),f,20*log10(abs(he2)))
;
Title('Elliptic filter');axis([0 4000 -60 1]);
figure; plot(f,abs(he1),f,abs(he2));
Title('Elliptic filter');
```

2. Design an IIR low pass filter for the above specifications using direct method. *Use MATLAB function* yulewalk.

```
clear; clc; close all;
Fs=8000;
wp=1000*2/Fs; ws=1500*2/Fs;
wpa=1000*2*pi; wsa=1500*2*pi;
Rp=0.5; Rs=30;
% Direct method
f = [0 wp ws 1];
m = [1 1 0 0];

[b,a] = yulewalk(10,f,m);
[h,w] = freqz(b,a,128);
plot(f,m,w/pi,abs(h),'--')
legend('Ideal','yulewalk Designed')
title('Comparison of Frequency Response Magnitudes')
```

3. **Design a FIR filter for the same specifications and compare the characteristics of IIR filters with that of FIR filters.**

4. **Generate a signal s(n) with three sinusoidal components at 5, 15 and 30Hz and sampled at 100Hz. Design an bandpass elliptic filter to keep the 15Hz sinusoid and eliminate the 5 and 30Hz harmonics.**

<u>Sample Solution</u>

```
fs=100; t=(1:100)/fs;
s= sin(2*pi*5*t)+sin(2*pi*15*t)+sin(2*pi*30*t);
plot(t,s); grid;
title('Time domain waveform (5,15,30 Hz)');

%  Design filter to keep 15 Hz and remove others
wp1=10/50; wp2=20/50; ws1=5/50; ws2=25/50;
wp=[wp1,wp2]; ws=[ws1,ws2]; rp=0.1; rs=40;

[n,wn]=ellipord(wp,ws,rp,rs)
[b,a] = ellip(n,rp,rs,wn);
freqz(b,a);

[H,w] = freqz(b,a);
plot(w*fs/(2*pi),abs(H)); grid;

sf = filter(b,a,s);
subplot(211); plot(t,sf); grid;
title('Filtered signal 15Hz');
```

```
S= fft(s,512); SF=fft(sf,512);
f=(0:255)/256*(fs/2);
subplot(212); plot(f, abs([S(1:256)', SF(1:256)'])); grid
```

5. Develop a sine-cosine generator and plot the first 50 samples of its two output sequences. Scale the outputs so that they both have maximum amplitude of ±1. Take 24 samples in one cycle.

6. Execute the filterDesigner program by typing **filterDesigner** at MATLAB command prompt. Study the characteristics of the different filters.