EE253 Prof. Essam Marouf
DSP I Matlab Handout # 7

FIR Filter Design: Comparison of Optimal and Window Methods

Problem (adapted from O&S)
a) use Matlab as a CAD tool(use the Matlab function remez) to design and order 74 equiripple (optimal or Parks-McClellan) bandpass filter having the following specs

$H_d(e^{j\omega}) =$	0	$W(\omega) =$	1	$0 \le \lvert\omega\rvert \le 0.3\pi$
	1		1	$0.35\pi \le \lvert\omega\rvert \le 0.6\pi$
	0		0.2	$0.7\pi \le \lvert\omega\rvert \le \pi$

b) If the FIR window method is to be used instead, choose an appropriate window type that yields the smallest filter order. Determine the filter order M and plot the corresponding filter response. Compare results of the window method with those of the Parks-McClellan method

a) Minimax or optimal design of a bandpass filter

```
% Script File: FIR_Optim_1.m
% ------------------------------
j=sqrt(-1);
M= 74;                                              % Given Filter Order
MM=37;
                                                    % Filter edges
wL1=0.3*pi;
wU1=0.35*pi;
wL2=0.6*pi;
wU2=0.7*pi;
f= [0  wL1/pi  wU1/pi  wL2/pi  wU2/pi  1];          % normalized to Fs/2 in Matlab

                                                    % Filter Design
Hd= [0  0  1  1  0  0];                             % desired (ideal) response
weight= [1  1  0.2];                                % wheights in different bands
h=remez(M,f,Hd,weight);                             % optimal design in Matlab

n=[0:1:M];
stem(n,h);                                          % Impulse response
axis([0 M -.4 .4]);
xlabel('Sample Number, n'),
ylabel('Impulse Response, h(n)')
title('Minimax design of a bandpass filter)
axis; pause

nn=512;
a=1;
[H,w]=freqz(h,a,nn);                                % Frequency response
plot(w/pi, 20*log10(abs(H)),'-r')
axis([0 1 -80 20]);
xlabel('Normalized Frequency, w/pi = f/fs')
ylabel('Magnitude Response, |H(f)| dB')
axis; pause
```

```
                                                % Compute and plot the error
                                                % Determine ranges for error
plot
index1=find(w <= wL1);
index2=find((w >= wU1)&(w <= wL2));
index3=find(w >= wU2);
%
                                  % Remove the linear phase component of H
MM=M/2;
As1=H(index1).*exp(j*MM*w(index1));
Ap=H(index2).*exp(j*MM*w(index2));
As2=H(index3).*exp(j*MM*w(index3));
plot(w(index1)/pi,-As1,'-',w(index2)/pi,1-Ap,'-',w(index3)/pi,-As2,'-')
axis([0 1 -.06 .06]);
xlabel('Normalized Frequency, w/pi = f/2fs'),
ylabel('Error'), axis;
```

---

b) Window design

Need to estimate the filter order first.

In the window method, one can not independently control the ripple in individual bands. So the design must be based on the requirement that is more restrictive.

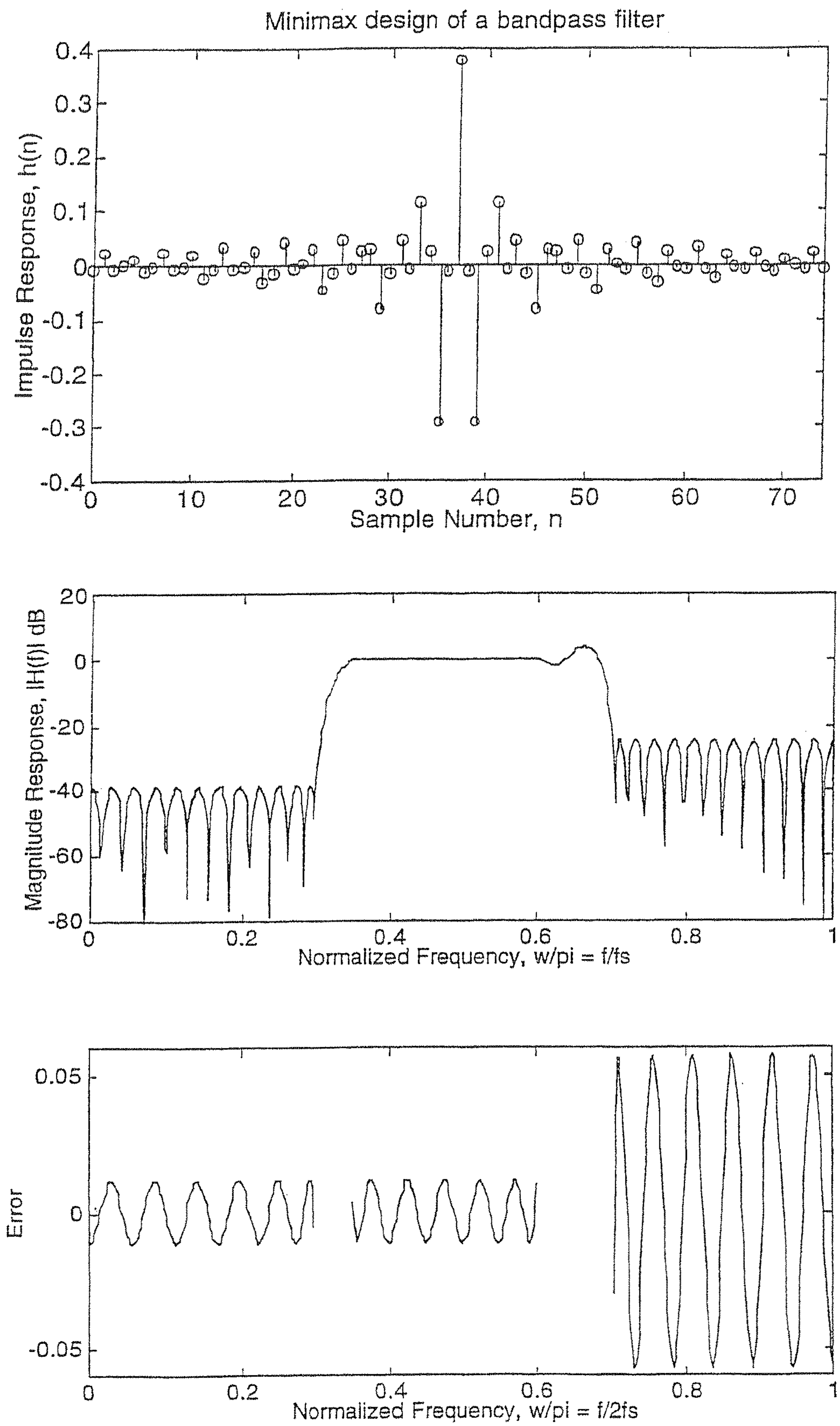
Results above indicate a passband ripple that is nearly equal to the ripple in the first stopband, which corresponds to an attenuation As of about 38.8 dB. Thus, a Hanning window with As= 44 dB is appropriate, or to optimize the filter order, a Kaiser window that provides the required As = 38. 8 with lower filter order is better. The order M for the Kaiser window can be estimated from M ~= (As - 8)/(2.285*dw)

```
% Script File: FIR_Optim_2.m
% ---------------------------------
j=sqrt(-1);
wL1=.3*pi;
wU1=.35*pi;
wL2=.6*pi;
wU2=.7*pi;
f= [0  wL1/pi  wU1/pi  wL2/pi  wU2/pi  1];
%
As=38.8;
dw = (wU1-wL1);                       % use the sharper transition
M= ceil((As-8)/(2.285*dw))            % estimate filter order
MM=M/2;                               % M is even here; Type I filter
%
beta=0.5842*(As-21)^0.4+0.07886*(As-21);
wn=kaiser(M+1,beta);                  % Kaiser window
%
wc1=.5*(wL1+wU1);                     % lower edge of passband
wc2=.5*(wL2+wU2);                     % upper edge of passband
w0=(wc1+wc2)/2;                       % center frequency
B=wc2-wc1;                            % bandwidth
%
n=[0:1:M];
h=2*cos(w0*(n-MM)).* (sin(0.5*B*(n-MM)+1.0E-10)./(pi*(n-MM)+1.0E-10));
h=h.*wn';                             % apply the Kaiser window
%
```

```
% The following was not required, but is included here for your information
% Check: Matlab as a CAD tool gives (fir1 is Matlab implementation of window design);
% an even order M is required to use fir1

h1=fir1(M, [wc1 wc2]/pi, wn);
%
                                              % Plot and compare  results
stem(n,h), hold on, plot(n,h1,'xr'), hold off
axis([0 M -.4 .4]);
xlabel('Sample Number, n'),
ylabel('Impulse Response, h(n)')
title('Window design of a bandpass filter')
axis; pause
%
                                              % Frequency Response
nn=512;a=1;
[H,w]=freqz(h1,a,nn);
plot(w/pi, 20*log10(abs(H))),
axis([0 1 -80 20]);
xlabel('Normalized Frequency, w/pi = f/fs')
ylabel('Magnitude Response, |H| dB')
axis; pause
%
                                              % Compute and plot the error
%
index1=find(w <= wL1);
index2=find((w >= wU1)&(w <= wL2));
index3=find(w >= wU2);
As1=H(index1).*exp(j*MM*w(index1));           % remove the linear phase component
Ap=H(index2).*exp(j*MM*w(index2));
As2=H(index3).*exp(j*MM*w(index3));
%
plot(w(index1)/pi,-As1,'-',w(index2)/pi,1-Ap,'-',w(index3)/pi,-As2,'-')
axis([0 1 -.015 .015]);
xlabel('Normalized Frequency, f/fs'),
ylabel('Error'),
axis;
```

Note that the error behavior is not any more equiripple. As is the case for all window designs, tha maximum error occurs near the transition edges, and becomes smaller away from the edges. Note also that a higher filter order (M= 86) compared to the minimax design (M=74). Note also that the specs are more than met at wc2.


Minimax design of a bandpass filter
Impulse Response, h(n)
Sample Number, n
Magnitude Response, |H(f)| dB
Normalized Frequency, w/pi = f/fs
Error
Normalized Frequency, w/pi = f/2fs

EE253, FIR Filter Design: Comparison of Optimal and Window Methods

M =

86