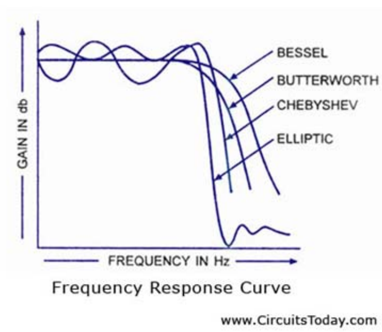
EE 419 - Project 10
Optimal FIR and Alternative IIR Filter Designs With Matlab



REMINDERS:
1) When you are finished, be sure to always copy your m-files and "C" source code files to a place that you can retrieve them again in the future. (Who knows??... You might just need to do some of this stuff again,...maybe even in the "real world"!)

1) [Matlab] FIR vs. IIR Filter Design Comparison

In this exercise you will use your Matlab filter design and analysis programs to investigate differences between optimal FIR filters, and IIR filters designed with the most common analog prototypes. You will use the same Low-pass filter design specifications for attenuations and transition bandwidths for all of the filters; and will first try to achieve these specifications using an optimal (equi-ripple) FIR filter design based on the Parks-McClellan Remez exchange algorithm. Then, you will create and analyze different IIR filters that meet the same specifications, using the Butterworth, Chebyshev (Type 1 and Type 2), and Elliptical designs. The required filter lengths and resulting performance and phase characteristics will vary depending on the type of filter (FIR vs IIR) and the analog prototype filter on which each IIR filter is based.

You will not be designing the filters "by hand". Rather, you will use the filter design tools available in Matlab for such designs. For the Parks-McClellan FIR design and for each type of IIR filter prototype, Matlab provides ready-made functions for determining the required filter orders (or filter lengths) and design frequencies needed to meet a given set of ripple / attenuation specifications with a particular filter design. Matlab also has functions to create the filter difference equation coefficients for the Parks-McClellan FIR and all the different types of IIR filter designs.

For example, in Matlab, the command to create FIR filter coefficients (impulse response) based the Parks-McClellan optimization method has the form:

```
B=firpm(N,F,A)
```

Which returns a **length N+1** linear phase (real, symmetric coefficients) FIR filter which has the best approximation to the desired frequency response described by **F** and **A** in the min-max sense. The maximum error is minimized.

N is the filter order (Filter length M = N+1)
F is a vector of <u>normalized</u> frequency band edges in pairs, in ascending order between 0 and 1.
[*1 corresponds to the Nyquist frequency or half the sampling frequency.* (**F** $=2*f_{analog}/f_{sample}$)]

A is a real vector the same size as **F** which specifies the desired amplitude of the frequency response of the resultant filter **B** at each frequency in the F vector.

The desired magnitude response is the line connecting the points (F(k),A(k)) and (F(k+1),A(k+1)) for odd k. FIRPM treats the bands between F(k+1) and F(k+2) for odd k as "transition bands" or "don't care" regions. Thus the desired amplitude response is piecewise linear approximation with transition bands.

Example: Length 31 lowpass filter, with passband from 0 - 0.1 cycles/sample, and stopband from 0.2 – 0.5 cyc./samp. (Transition band is from 0.1 – 0.2 cycles /sample)

```
h=firpm(30,[0 .1 .2 .5]*2,[1 1 0 0]);
```

[from Matlab: **help firpm**]

Matlab also has a function for predicting the filter order needed to achieve the desired frequency response shape and the allowed ripple specifications for a Parks-McClellan optimal FIR filter:

```
[N,Fo,Ao,W] = firpmord(F,A,DEV,Fs)
```

which finds the approximate filter order N, normalized frequency band edges Fo, frequency band magnitudes Ao and weights W to be used by the FIRPM function as follows:

```
b = firpm(N,Fo,Ao,W)
```

[see Matlab: **help firpmord**]

Matlab also provides additional functions to determine the required filter order and digital filter coefficients for each IIR filter type. [See help files for: `buttord()`, `butter()`, `cheb1ord()`, `cheby1()`, `cheb2ord()`, `cheby2()`, `ellipord()`, `ellip()`].

For this study, you are to create composite magnitude AND PHASE response plots (with multiple results overlay plotted on the same axes) for the different resulting filter designs. For the overlay plots, be sure to use different line types and/or colors for the different plots and a "legend" for the figure so that you can distinguish the different results. (see Matlab: `help plot and help legend`). Create both linear and log (dB) scale magnitude response plots for your comparisons. Then, compare your results by filling in the Table below for each case, with the results determined by your analysis programs or by making measurements from your plots. Be sure to use a large number of frequency response data points in the plots to ensure accurate measurements. You may assume a sampling frequency of S = 1 KHz for your analysis.

Finally, determine how you would implement the different IIR filter designs so that you minimized potential problems of coefficient rounding and arithmetic overflow in a fixed-point DSP implementation [*i.e.* divide up the design into cascaded 2nd-order stages.] Determine the difference equation coefficients for each 2nd order stage in the implementation. Show the coefficient values as unscaled floating point values with 4 decimal place precision.

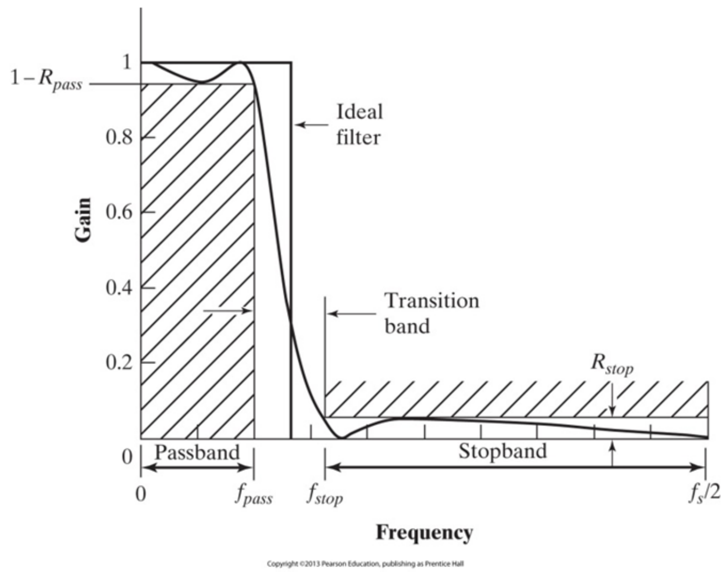
All designs should use the following design specification requirements:

Filter Length: As needed to meet specs
Real Filter Coefficients
Maximum Ripples: $\delta_p = \delta_s = 0.013$
Passband Edge Frequency: $F_p = 0.1333$ cyc/sample
Stopband Edge Frequency: $F_s = 0.2667$ cyc/sample
Transition Bandwidth: $\Delta F_T = 0.1333$ cyc/sample

Be sure to note that ripples and attenuations are not specified the same for IIR filters as they are for FIR filters; as IIR filter designs typically have a peak amplitude of 1.0 (as opposed to $1+\delta_p$ as in FIR filters). Therefore, the passband and stopband attenuations for IIR filters are given by:

R_{pass} = Passband Ripple $\qquad$ R_{stop} = Stopband Ripple

$A_{p(dB)} = -20\log(1-R_{pass})$ and $A_{s(dB)} = -20\log(R_{stop})$



The alternative filter designs to be designed, evaluated, and compared are:

a) FIR Parks-McClellan
b) IIR Butterworth Filter
c) IIR Chebyshev Type 1 Filter
d) IIR Chebyshev Type 2 Filter
e) IIR Elliptical Filter

For each design, you will:

1) Estimate the required filter length (filter order + 1) to meet the specifications using the available Matlab functions.
2) Design the filter using the appropriate Matlab function for the desired filter type.
3) Analyze the resulting filter using your Matlab analysis functions, and verify whether or not it met all of the design specifications. If not, revise the design (probably with a longer filter length), and repeat the process until all specifications are met.
4) Record the final filter design difference equation coefficients (Table 1), and the performance of the filter for each design specification parameter (Table 2).

For the IIR filter designs, you will also:

5) Convert the filter designs into cascaded 2nd-order section implementations. Group the poles and zeros appropriately, and order the sections to minimize possible overflow problems if the sections were implemented using scaled integer coefficients. (You do not have to choose scale factors; assume you can keep the 2nd–order filter coefficients in unscaled floating point form).

Submit:

a) **Listing of the Matlab commands** used to create each design. (Analysis steps do not need to be shown).

b) Completed Table 1 (below) with the Difference Equation coefficients for each of the different filter designs (assuming a Direct-Form 1 implementation with a single section for all filters). Also **determine the minimum number of multiplication operations** that would be needed **to implement each filter** using a floating-point processor, if you were able to implement any filter structure you wanted.

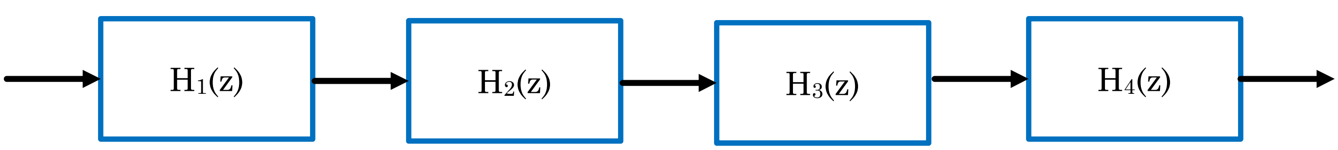
c) Pole-Zero Plots for each design (individual plot for each design).

d) **Two Composite Magnitude Response Plots in a single Figure,** showing all 5 designs plotted in the same figure **using a linear magnitude scale**. (No phase response plot). Use 2 subplots: one subplot with the magnitude scale ranging from $0 < |H| < 0.05$ and frequencies from 0 – 0.5 cycles/sample (to see the full frequency range and details of the stop band ripple); and a second subplot with a magnitude range of $0.95 < |H| < 1.05$ and frequencies from 0 – 0.25 cyc/sample (to see passband ripple details). [See the Matlab `axis( )` command.]

e) Two Composite Magnitude Response Plots in a single Figure, showing all 5 designs **using a dB magnitude scale**. (No phase plots) Use 2 subplots: one subplot with the dB magnitude scale ranging from -50 dB < |H| < +10 dB and frequencies from 0 – 0.5 cycles/sample (to see the full amplitude range and details of the stop band ripple); and a second subplot with a dB magnitude range of -0.5 dB < |H| < + 0.5 dB and frequencies from 0 – 0.25 cyc/sample (to see passband ripple details).

f) **Completed Table 2 (below) with the resulting performance for each filter design.** Determine the passband and stopband edge frequencies based on the frequency at which the specified maximum ripple values δ_s and $\delta_p = 0.013$ are crossed in the transition band for all filter designs. Also determine the actual values achieved by each filter for ripples at the specified target passband and stopband edge frequencies. Finally, determine the maximum passband and minimum stopband attenuations occurring anywhere in each band (passband: $0 < F < Fp$; stopband: $Fs < F < 0.5$).

g) **Label the poles and zeros in the Pole/Zero diagrams for the IIR filters with the 2nd–order section number** that they would be assigned to in a cascade 2nd-order implementation. Assume the 2nd order filter sections are ordered from left to right ($H_1(z)$ first, then $H_2(z)$, *etc.*) as shown below. For example, the two poles and two zeros that would be assigned to section $H_1(z)$ would be labelled with the number "1" in the Pole/Zero Diagram. Poles and zeros for section $H_2(z)$ would be labelled with the number "2".



a) Based on the pole/zero locations, which IIR filter type do you think would be more susceptible to arithmetic overflow problems when implemented with scaled integer filter coefficients in cascaded 2nd order sections? **Explain your reasoning.**

b) Conclusions (in words) of your comparison between filter types (optimal FIR, Butterworth IIR, Chebyshev 1 & 2 IIR, Elliptical IIR). Identify relative differences in the performance of each approach, and discuss the tradeoffs involved in selecting each filter type. Consider the filter lengths, specifications achieved, and phase characteristics of each filter design.

Filter Coeff	FIR Parks-McClellan	Filter Coeff	IIR – Butterworth	IIR – Chebyshev 1	IIR – Chebyshev 2	IIR – Elliptical
B_0		B_0				
B_1		B_1				
B_2		B_2				
B_3		B_3				
B_4		B_4				
B_5		B_5				
B_6		B_6				
B_7		B_7				
B_8		B_8				
B_9		B_9				
B_{10}						
B_{11}		A_0	1.0	1.0	1.0	1.0
B_{12}		A_1				
B_{13}		A_2				
B_{14}		A_3				
B_{15}		A_4				
B_{16}		A_5				
B_{17}		A_6				
B_{18}		A_7				
		A_8				
A_0	1.0	A_9				
Minimum # of Multiplies Needed		Minimum # of Multiplies Needed				

Table 1 – Filter Difference Equation Coefficients
(Fill in only those blocks needed for your filter designs)

Design Method	Filter Order	Maximum Passband Attenuation Ap (dB) $\lvert H\rvert_{max}/(1-\delta_{pmax})$ @$F < F_p$	Passband Ripple δp @F_p=.133	Stopband Ripple δs @F_s=.267	Minimum Stopband Attenuation As (dB) $\lvert H\rvert_{max}/(\delta_{s\,max})$ @$F > F_s$	Passband Edge Freq. Fp (cyc/spl) @$\lvert H\rvert=1-\delta_p$ (δ_p=.013)	Stopband Edge Freq. Fs (cyc/spl) @$\lvert H\rvert=\delta_s$ (δs=.013)	Transition Band Width ΔF_T (cyc/spl)
Specification	-	FIR: 0.226 IIR: 0.114	.0130	.0130	FIR: 37.8 IIR: 37.7	.133	.267	.133
FIR Parks-McClellan								
IIR Butterworth								
IIR Chebyshev 1								
IIR Chebyshev 2								
IIR Elliptical								

Table 2 – Filter Performance Comparison