Digital Signal Processing

ECEN 5763
Fall 2001

Design of Digital FIR Filters With Matlab
Using Parks-McClellen Method and the Window Design Method

Purpose

The purpose of this assignment is for you to learn more about designing optimum digital FIR filters using the Parks-McClellen Method (Remez exchange algorithm).

Introduction to FIR Design with Matlab (partially repeated from the previous assignment)

Matlab contains various routines for designing and analyzing digital IIR and FIR filters (and other z-domain functions). Most of these are part of the Signal Processing Toolbox, which must be available for you to complete this assignment. A selection of these functions is listed below. Not all of the listed functions will be useful in this assignment. For additional information, type *help signal* at the Matlab prompt to get a list of all the functions in the Signal Processing Toolbox. You should also refer to the on-line help for each function you consider using as this handout is not intended to be a tutorial in their use.

When using the Parks-McClellen method, you will use the *remez* routine. Although there is a routine for estimating the order necessary for the Parks-McClellen design (*remezord*), use the equations given in the textbook instead. The Matlab on-line help (*helpdesk*, accessed from the Matlab command prompt) contains several good examples of using *remez*. Look in particular at the on-line Acrobat manual (pdf) document titled *Signal Processing Toolbox User Guide*.

freqz	z-domain frequency response
buttord, cheb1ord, ellipord	IIR analog/digital filter order calculation
butter, cheby1, ellip	IIR analog/digital filter calculation
fir1	FIR digital design (window) - standard response
fir2	FIR digital design (window) - arbitrary response
Firls	FIR digital design - arbitrary response w/t_band
remez	Parks-McClellan optimal FIR digital design
remezord	Parks-McClellan filter order calculation
abs, angle, unwrap	magnitude, phase, unwrapped phase
fft, ifft	forward and inverse discrete Fourier transform
bilinear	analog to digital bilinear transformation
impinvar	analog to digital impulse invariance
invfreqz	digital filter fit to frequency response
zplane	pole/zero plotting routine

The FIR Filter Design Problem(s)

Use Matlab to design the optimum digital FIR filters specified below. Perform the indicated analysis. These are the same filters you designed in the previous IIR assignment.

Problem FIR-1: Use Matlab to design the *lowpass* filter specified below using the Parks-McClellen method such that the specifications are met or exceeded. If the specifications are not met on your first attempt, redesign the filter until a satisfactory solution is found. You may need to alter the length and/or the error weighting to get an appropriate solution. It's not often necessary to vary the filter specifications you provide to *remez* in order to meet the stated specification, but you may consider this as well if necessary.

-1 dB or less at 1200 Hz and below (maximum passband ripple)	-50 dB or more at 1500 Hz and above (minimum stopband attenuation)
sampling frequency: 10,000 Hz	

For your filter, verify that the specifications were met at all critical frequencies by performing the following analysis. Plot the overall magnitude frequency response using a dB amplitude scale (*freqz* will create magnitude and phase plots together on the same figure). Plot the transition band(s) in dB with sufficient frequency resolution and numeric precision to verify the specifications were met (the technique discussed class last week or the Matlab *zoom* command may be helpful). Write directly on the transition band plot(s) the actual dB response value at each critical frequency (passband edge and stopband edge). Create a pole/zero plot (*zplane*). Comment very briefly on the placement of poles/zeros as related to realization of a lowpass response. Finally, create an overall phase plot (if one was not already presented) to demonstrate that the filter is linear phase.

Note that varying the error weighting between bands is quite useful to meet the desired specifications and to introduce a tradeoff between stopband attenuation and necessary filter length. For example, increasing the error weighting in the stopband should be tried as an alternative to (before) increasing the filter length. Try to adjust the available parameters in order to minimize the required filter length.

Use methods to minimize the number of plots you must print, including the *subplot* command or, as a preferred alternative, cut and paste through the Windows clipboard to a word processor document.

What was the filter length required to meet the specifications *before* and *after* you tried to optimize the order? What error weighting did you use?

Estimate the order of the filter required using the *FIR Window Design Procedure* discussed in class (evaluate the order using the *formula* from the table of window properties in the text). You do not need to actually design the filter. What window function did you choose? Is this order significantly different than what was required for Parks-McClellen?

Problem FIR-2: Using the same basic procedure as described above, design a *bandpass* filter using the Parks-McClellen method to meet these specifications:

-50 dB or more from 0 to 1,200 Hz	-1 dB or less from 2,000 Hz to 4,000 Hz
sampling frequency: 16,000 Hz	-50 dB or more above 6,000 Hz

Consult the help file (try using *help remez* and/or *helpdesk,* and searching for *remez,* at the Matlab command line) to see exactly how to specify a multiband filter to Matlab when using Parks-McClellen.

Carefully verify that the specifications were met at all critical frequencies as in the previous problem. Plot the overall magnitude frequency response using a dB amplitude scale. Plot the transition band(s) in dB with sufficient frequency resolution and numeric precision to verify the specs were met. Write directly on the transition band plot(s) the actual dB response value at each critical frequency (passband edge and stopband edge). Create a pole/zero plot. Finally, create an overall phase plot.

As above, try to minimize the filter length. What was the required filter length *before* and *after* minimization? What error weighting did you use to design the filter?

Comment very briefly on the placement of the poles/zeros as their placement relates to the production of a bandpass response in this filter design.

Problem FIR-3: Finally, repeat the LPF design in **FIR-1**, this time using the Window Design Method. Use an appropriate Kaiser window for the design. Show your calculations for estimated Kaiser window parameter and window length, as well as the cutoff frequency for Matlab.

Carefully verify that the specifications were met at all critical frequencies as in the previous problems. If your first attempt at this design does not meet the specifications, iterate your design as necessary until the specifications are met. Once you can meet the specifications, document your design. In particular, plot the overall magnitude frequency response using a dB amplitude scale. Plot the transition band(s) in dB with sufficient frequency resolution and numeric precision to verify the specs were met. Write directly on the transition band plot(s) the actual dB response value at each critical frequency (passband edge and stopband edge). Create a pole/zero plot. Finally, create an overall phase plot.

Once you have a filter that meets the specifications and you have documented your design, try to optimize the design by iterating on the filter length and cutoff frequency provided to Matlab (this technique was discussed in class and was the subject of a handout). What are the filter length and cutoff frequency for the optimized design? Produce a magnitude and phase plot illustrating your optimized design.

Comment very briefly on the placement of the poles/zeros as their placement relates to the production of a lowpass response in this filter design.

The Report

Turn in your filter designs and the corresponding analyses, complete with plots and all requested information from the questions above. Everything should be clearly labeled and organized *in the order in which it was requested*. Neatness is important! *Type* your answers or *print* clearly as requested directly on the plots. Use methods to reduce the number of pages in your assignment – put multiple plots on a single page. You are strongly encouraged to use Microsoft Word to produce your document as this allows direct importing of Matlab plots and greatly reduces the number of pages of plots that you must print and hand in.

This assignment is due on Monday, October 29, at the beginning of class.