COMP ENG 4TL4 – DIGITAL SIGNAL PROCESSING

Lab #4: Filter Design and Analysis

Objective:

To gain experience in filter design and analysis within MATLAB.

Assessment:

- Your grade for this lab will be based on your ability to use filter design tools from the MATLAB Signal Processing Toolbox and to analysis and implement the resultant filters, and on your reporting of the results. The report should contain any mathematical calculations or derivations carried out, MATLAB plots of results with brief descriptions, the MATLAB code with which you obtained your results, and answers to specific questions below.
- Clearly label all plots and their axes (points for style will be deducted otherwise)
- DON'T COPY OTHERS BLINDLY!!
- Please attend the lab section to which you have been assigned.
- You should complete this lab with one lab partner. If there are an odd number of students, then one group of three will be created by the TA.
- Each pair of students should complete one lab report together, which is to be submitted one week from the date of the lab. Those students in the "At Home" section must submit their reports one week from the day on which they demonstrated their lab to Jeff Bondy.

Pre-lab:

- Carefully read through this lab description, so that you know what is required.
- Read through the lecture notes (and bring them with you) so that you know how to answer the questions.
- Familiarize yourself with the MATLAB commands that may be required for this lab – see the list at the end of this lab description for some hints.

1. The truncation method of FIR design:

a. Follow the math from Lecture #22 to design linear-phase lowpass filters with the cutoff-frequency $\omega_c = \pi/3$ using the truncation method for filter orders $M = 20$, 50 and 150.

b. Demonstrate the Gibbs phenomenon for these filters.

c. Confirm that these filters have linear phase.

d. Show that the MATLAB Signal Processing Toolbox function `firls()` function, which designs linear-phase FIR filters using least-squares error minimization, produces identical impulse responses to the truncation method for each of the filter orders $M = 20$, 50 and 150.

e. Create a signal 1000 samples long of white Gaussian noise. Convolve this signal with each of the filter impulse responses obtained in part a (or part c). Now try using the MATLAB function `filter()`. What similarities and differences are there between the outputs of `filter()` and `conv()`?

2. The window method of FIR design:

a. Design linear-phase lowpass filters with the cutoff-frequency $\omega_c = \pi/3$ using the window method with a Dolph-Chebyshev window for filter orders $M = 20$, 50 and 150. The MATLAB Signal Processing Toolbox function `fir1()` can be used to implement the window method and the MATLAB Signal Processing Toolbox function `chebwin()` can be used to obtain the desired Dolph-Chebyshev windows.

b. Note that you will need to specify the amount of sidelobe attenuation for your Dolph-Chebyshev windows. Try some different values, and demonstrate what effect this has on:

   i. the passband ripple and stopband attenuation, and

   ii. the slope of the transition region,

   for <u>one</u> of the filters ($M = 20$, 50 or 150).

c. Do you observe the Gibbs phenomenon in the window method?

3. The Parks-McClellan optimization method of FIR design:

a. Design linear-phase lowpass filters with the cutoff-frequency $\omega_c = \pi/3$ using the Parks-McClellan method for filter orders $M = 20$, 50 and 150. The MATLAB Signal Processing Toolbox function `remez()` can be used to implement the Parks-McClellan method.

b. Note that you will not be able to specify the ideal passband and stopband cutoff frequencies exactly in the `remez()` function. Try specifying a few different passband and stopband cutoff frequencies that vary in how close they are to the ideal lowpass filter, and see what effect this has on the actual frequency responses.

c. How do the resulting filter frequency responses compare to those obtained from the truncation and window methods?

4. IIR filter design and analysis:

a. Design Chebyshev Type II lowpass IIR filters with the cutoff-frequency $\omega_c = \pi/3$ using the MATLAB Signal Processing Toolbox function `cheby2()` for filter orders $M = N = 20$, 50 and 150. Note that you will need to specify the stopband attenuation – make it the same as one of the values that you used for the Dolph-Chebyshev window sidelobe attenuation in section 2b above.

b. Determine the impulse responses of these filters. Do they appear to be infinite?

c. Compare and contrast the IIR filter frequency responses (both the <u>magnitude</u> and the <u>phase</u>) with the FIR frequency responses from *section 2* above. Explain the differences in light of filter theory as it relates to causal FIR and IIR filters.

d. Using the `filter()` function, calculate the output of the IIR filter for the white Gaussian noise signal from section 1e above. Now estimate the output by convolving the input signal with a very long impulse response that you obtained from part b above. How good is the estimate?

Potentially useful MATLAB commands

Note that this is not an exhaustive list! You are not required to incorporate all of these in your scripts.

help <topic>	helpwin	figure	plot	stem
hist	subplot	hold on	xlabel	ylabel
legend	title	function	clear	close
clc	filter	firls	fir1	chebwin
cheby2	remez	conv	freqz	impz
abs	angle	unwrap	randn	length