Digital Signal Processing (30.115)
EPD Term 7

Lab Exercise 3

Due time: 6pm, 20th April 2020

Design of Digital Filters I

3.1 MATLAB Commands Used

The MATLAB commands you will encounter in this exercise are as follows:

General Purpose Commands

disp length

Operators and Special Characters

: . + - * / ;
% .* ./ > ==

Two-Dimensional Graphics

axis grid plot title xlabel
ylabel

Signal Processing Toolbox

blackman	butter	buttord	chebwin	cheb1ord
cheb2ord	cheby1	cheby2	ellip	ellipord
fir1	fir2	firpm	firpmord	freqz
hanning	hamming	kaiser		

latc2tf	poly2rc	residue	residuez
tf2latc	zp2sos		

For additional information on these commands, see the MathWorks Online documentation.

3.2 Design filters with known transfer function

Project 3.1 FIR Cascade Realization

The factored form of a causal FIR transfer function H(z) of order M − 1, as given below can be determined from its polynomial form. To generate the factored form, Program P3_1 that uses the function zp2sos can be employed.

$$H(z) = h[0] \prod_k (1 + \beta_{1k} z^{-1} + \beta_{2k} z^{-2})$$

```
%   Program P3_1
%   Conversion of a rational transfer function
%   to its factored form
num  = input('Numerator coefficient  vector = ');
den = input('Denominator coefficient  vector = ');
[A, B]  = eqtflength(num, den);
[z,p,k] = tf2zp(A, B);
sos = zp2sos(z,p,k)
```

Q3.1.1 Provide inputs as the filter coefficients in numerator and denominator. Explore the functions in this program and comments what are tf2zp and zp2sos doing?

Q3.1.2 Using Program P3_1 to develop a cascade realization of the following FIR transfer function:

$$H1(z) = 2 + 10z^{-1} + 23z^{-2} + 34z^{-3} + 31z^{-4} + 16z^{-5} + 4z^{-6}$$

Sketch the block diagram of the cascade realization.
Use Matlab to plot the phase response. Is H1(z) a linear-phase transfer function?

Q3.1.3 Using Program P3_1 to develop a cascade realization of the following FIR transfer function:

$$H2(z) = 6 + 31z^{-1} + 74z^{-2} + 102z^{-3} + 74z^{-4} + 31z^{-5} + 6z^{-6}$$

Sketch the block diagram of the cascade realization.
Use Matlab to plot the phase response. Is H2(z) a linear-phase transfer function?

Sketch a new cascade structure of H2(z) with only 4 multipliers.

Project 3.2 IIR Cascade Realizations

Q3.2.1 Using Program P3_1 to design an IIR filter cascade form with the following causal IIR filter with transfer function:

$$H_1(z) = \frac{3 + 8\,z^{-1} + 12\,z^{-2} + 7\,z^{-3} + 2\,z^{-4} - 2\,z^{-5}}{16 + 24\,z^{-1} + 24\,z^{-2} + 14\,z^{-3} + 5\,z^{-4} + \,z^{-5}}$$

Sketch the block diagram of the cascade realization. Each sub-system is implemented using direct form I.

Q3.2.2 Using Program P3_1 to design an IIR filter cascade form with the following causal IIR filter with transfer function:

$$H_2(z) = \frac{2 + 10\,z^{-1} + 23\,z^{-2} + 34\,z^{-3} + 31\,z^{-4} + 16\,z^{-5} + 4\,z^{-6}}{36 + 78\,z^{-1} + 87\,z^{-2} + 59\,z^{-3} + 26\,z^{-4} + 7\,z^{-5} + \,z^{-6}}$$

Sketch the block diagram of the cascade realization. Each sub-system is implemented using direct form II.

Design of Digital FIR Filters

4.1 **MATLAB functions may be used**

The MATLAB commands you will encounter in this exercise are as follows:

General Purpose Commands

disp length

Operators and Special Characters

: . + - * / ;
% .* ./ > ==

Language Constructs and Debugging

else function if

Elementary Matrices and Matrix Manipulation

fliplr nargin pi :

Elementary Functions

abs ceil cos log10 sin sqrt

Data Analysis
min

Two-Dimensional Graphics

axis grid plot title xlabel
ylabel

Signal Processing Toolbox

blackman	butter	buttord	chebwin	cheb1ord
cheb2ord	cheby1	cheby2	ellip	ellipord
fir1	fir2	firpm	firpmord	freqz
hanning	hamming	kaiser		

For additional information on these commands, see the MathWorks Online Documentation.

4.2 Digital FIR Filter Design by Windows method and PM method

Conceptually the simplest approach to FIR filter design is to simply truncate to a finite number of terms from the infinite-length impulse response coefficients obtained by computing the inverse discrete-time Fourier transform of the desired ideal frequency response. However, a simple truncation results in an oscillatory behavior in the respective magnitude response of the FIR filter, which is more commonly referred to as the Gibbs phenomenon.

The Gibbs phenomenon can be reduced by windowing the doubly infinite-length impulse response coefficients by an appropriate finite-length window function. The functions **fir1** and **fir2** can be employed to design windowed FIR digital filters in MATLAB. Both functions yield a linear-phase design.

4.2.1

The function fir1 can be used to design conventional lowpass, highpass, bandpass, and bandstop linear-phase FIR filters. The command

b = fir1(N, w_n)

returns the impulse response coefficients in vector b, arranged in ascending powers of z^{-1} , of a lowpass or a bandpass filter of order N for an assumed sampling frequency of 2Hz. For lowpass design, the normalized cutoff frequency is specified by a scalar Wn, a number between 0 and 1. For bandpass design, Wn is a two-element vector [w_{n1}, w_{n2}] containing the specified passband edges where $0 < w_{n1} < w_{n2} < 1$.

4.2.2

The command b = fir1(N, w_n,’high’) with N an even integer, is used for designing a highpass filter.

4.2.3

The command b = fir1(N, w_n,’stop’) with w_n a two-element vector, is employed for designing a bandstop FIR filter. **If none is specified, the Hamming window** is employed as a default.

4.2.4

The command b = fir1(N, Wn, taper) makes use of the specified window coefficients of length N+1 in the vector taper. However, the window coefficients must be generated a priori using an appropriate MATLAB function such as blackman, hamming, hanning, chebwin, or kaiser. The commands to use are of the following forms:

taper = blackman(N)   taper = hamming(N)   taper = hanning(N)
taper = chebwin(N)   taper = kaiser(N, beta)

4.2.5

The function fir2 can be used to design linear-phase FIR filters with arbitrarily shaped magnitude responses. In its basic form, the command is

b = fir2(N, fpts, mval)

which returns in the vector b of length N+1 the impulse response coefficients, arranged in ascending powers of z^{-1} . fpts is the vector of specified frequency points, arranged in an increasing order, in the range 0 to 1 with the first frequency point being 0 and the last frequency point being 1. As before, the sampling frequency is assumed to be 2Hz. mval is a vector of specified magnitude values at the specified frequency points and therefore must also be of the same length as fpts. The Hamming window is used as a default. To make use of other windows, the command to use is

b = fir2(N, fpts, mval,taper)

where the vector taper contains the specified window coefficients.

4.2.6

A more widely used linear-phase FIR filter design is based on the Parks–McClellan algorithm, which results in an optimal FIR filter with an equiripple weighted error E(ω). It makes use of the Remez optimization algorithm and is available in MATLAB as the function firpm. This function can be used to design any type of single-band or multiband filter, the differentiator, and the Hilbert transformer. In its basic form, the command

b=firpm(*N*,fpts,mval)

returns a vector *b* of length *N*+1 containing the impulse response coefficients of the desired FIR filter in ascending powers of z^{-1}. fpts is the vector of specified frequency points, arranged in increasing order, in the range 0 to 1 with the first frequency point being 0 and the last frequency point being 1. As before, the sampling frequency is assumed to be 2 Hz.

The desired magnitudes of the FIR filter frequency response at the specified band edges are given by the vector mval, with the elements given in equal-valued pairs. The desired magnitudes between two specified consecutive frequency points f(k) and f(k+1) are determined according to the following rules. For k is odd, the magnitude is a line segment joining the points {mval(k), fpts(k)} and {mval(k+1), fpts(k+1)}, whereas, for k is even, it is unspecified with the frequency range [fpts(k), fpts(k+1)] being a transition or “don’t care” region. The vectors fpts and mval must be of the same length with the length being even.

Project 4.1 Estimation of Order of FIR Filter

Q4.1.1 Using the function kaiserord, write a code to estimate the order of a linear-phase lowpass FIR filter with the following specifications: passband edge = 2 kHz, stopband edge = 2.5 kHz, passband ripple δ_p = 0.005, stopband ripple δ_S=0.005, and sampling rate of 10 kHz.

Q4.1.2 Repeat **Q4.1.1** for the following cases: (a) sampling rate of 20 kHz, (b) δ_p = 0.002 and δ_S = 0.002, and (c) stopband edge = 2.3 kHz. Compare the filter length obtained in each case with that obtained in Q4.1.1. Comment on the effect of the sampling rate, ripples, and the transition bandwidth on the filter order.

Q4.1.3 Repeat Question **Q4.1.1** using the function firpmord and estimate the filter length.

Project 4.2 FIR Filter Design using windows

Q4.2.1 Using the function `fir1`, design a linear-phase FIR lowpass filter meeting the specifications given in **Q4.1.1** and plot its gain and phase responses. Use the order estimated using Kaiser's formula in **Q4.1.1**, show the filter coefficients in a tabular form. Plot the frequency response to verify the design.

Project 4.3 FIR Filter Design using Parks-McClellan method

Q4.3.1 Design an FIR lowpass filter using the function firpm. The filter specifications are: $\omega_p = 0.2\pi$, $\omega_s = 0.3\pi$, and stopband ripple is $A_s = 50$ dB. Required passband ripple is 25dB. Subfunctions "db2delta.m" and "freqz_m.m"are given in eDimension. Plot the frequency response to verify the design.

Note:
db2delta.m - Conversion of dB specs to delta specs.
freqz_m.m – A modified version of the Matlab function "freqz.m".