CMPE 422/Spring 08/Project 3: Least Squares based filter design

Samir Chettri

January 3, 2008

Abstract

In class we will study principles filter design (chapters 6, 9 and 10 of our class text by Ambardar). Some of these filter design principles are easy (conceptually and computationally) and others are not. This project considers a deceptively simple method to design filters involving simple matrix algebra - which is known to all engineering students - namely the method of Least Squares.

Students will first do some simple derivations followed by actual numerical calculations for filter design in MATLAB. They will hand in both their derivations and program that does the calculations. Upto fifty percent of the points for the project will be given for derivations and upto fifty percent for the MATLAB programs. The maximum grade a student can get on this project is one hundred points.

Keywords: Least Squares, FIR filter design, MATLAB.

Contents

1 The problem

Our textbook, on page 151, writes a transfer function as a ratio of two polynomials in z:

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1} + \ldots + b_N z^{-N}}{1 + a_1 z^{-1} + \ldots + a_D z^{-D}}. \tag{1}$$

$U(z)$ is the z–transform of the input signal and $Y(z)$ is the z–transform of the output signal. We also assume that the system is causal. Hence the difference equation in the time domain (Ambardar, p. 62) is:

$$y[k] = -a_1 y[k-1] - \ldots - a_D y[k-D] + b_0 u[k] + \ldots + b_N u[k-N]. \tag{2}$$

In matrix–vector form, equation (2) is written as:

$$y[k] = \begin{bmatrix} -y[k-1] & \ldots & y[k-D] & u[k] & \ldots & u[k-N] \end{bmatrix} \begin{bmatrix} a_1 \\ . \\ . \\ . \\ b_0 \\ . \\ . \\ . \\ b_N \end{bmatrix} \tag{3}$$

Now consider the frequency response of a digital system (Ambardar, p. 232); if the input frequency is ω_i, then the output amplitude will be A_i times the input amplitude and the output phase is also modified (shifted) by an angle ϕ_i. Apply this to the previous equations.

Let the input at time k be

$$u_1[k] = \cos[k\omega_1 t_s],$$

where, t_s is the sampling period. The output will then be

$$y_1[k] = A_1 \cos[k\omega_1 t_s + \phi_1].$$

If we set the current sample index to 0 (this is arbitrary and should not cause confusion), we have:

$$y_1[k] = \begin{bmatrix} -y[-1] & \ldots & y[-D] & u[0] & \ldots & u[-N] \end{bmatrix} \begin{bmatrix} a_1 \\ . \\ . \\ . \\ b_0 \\ . \\ . \\ . \\ b_N \end{bmatrix} \tag{4}$$

One can continue this process with a different input frequency ω_2 and therefore different output amplitude A_2 and phase ϕ_2. Clearly we don't have to stop here. Continuing this process we get a matrix equation that we denote as:

$$\mathbf{y} = \mathbf{X}\theta. \tag{5}$$

Now here is the interesting thing. Equation (5) is a matrix vector equation. Provided we know the inputs $u_i[.]$ and the outputs $y_i[.]$, we can solve for the unknown vector θ. Now, there are $D+N+1$ unknowns in the θ vector. To get θ we need at least these many equations in (5). However, from our statistics courses, we would like to have more points than this in order to reduce error. This lead to a least squares problem. At this point students should do a little research into a matrix vector solution to the least squares problem as they will need it in their programs. Remember, $u_i[.]$ and $y_i[.]$ are themselves written in terms of the sampling period t_s, the input frequency ω_i. $y_i[.]$ is additionally written in terms of A_i, the desired magnitude response and ϕ_i, the desired phase response. So essentially, all entries in $\mathbf{y}$ and θ can be filled in.

2 An example

We are going to provide inputs to an FIR filter design. You are expected to use equation (5) to get the filter coefficients. The system sample rate is 1000Hz and let $N = D = 2$. Assume we are going to sample the magnitude and phase responses at 8 frequencies. These are shown in a table below.

3 What to do?

This project has two parts. The first part deals with performing the relatively simple derivations needed to understand the basic theory while the latter one deals with programming in MATLAB.

Derivations - 15%.

1. We have only shown a matrix vector equation (5), i.e., no entries are shown. For $N = 10$ and $D = 9$ obtain all the entries in the $\mathbf{y}$ vector and the $\mathbf{X}$ matrix.

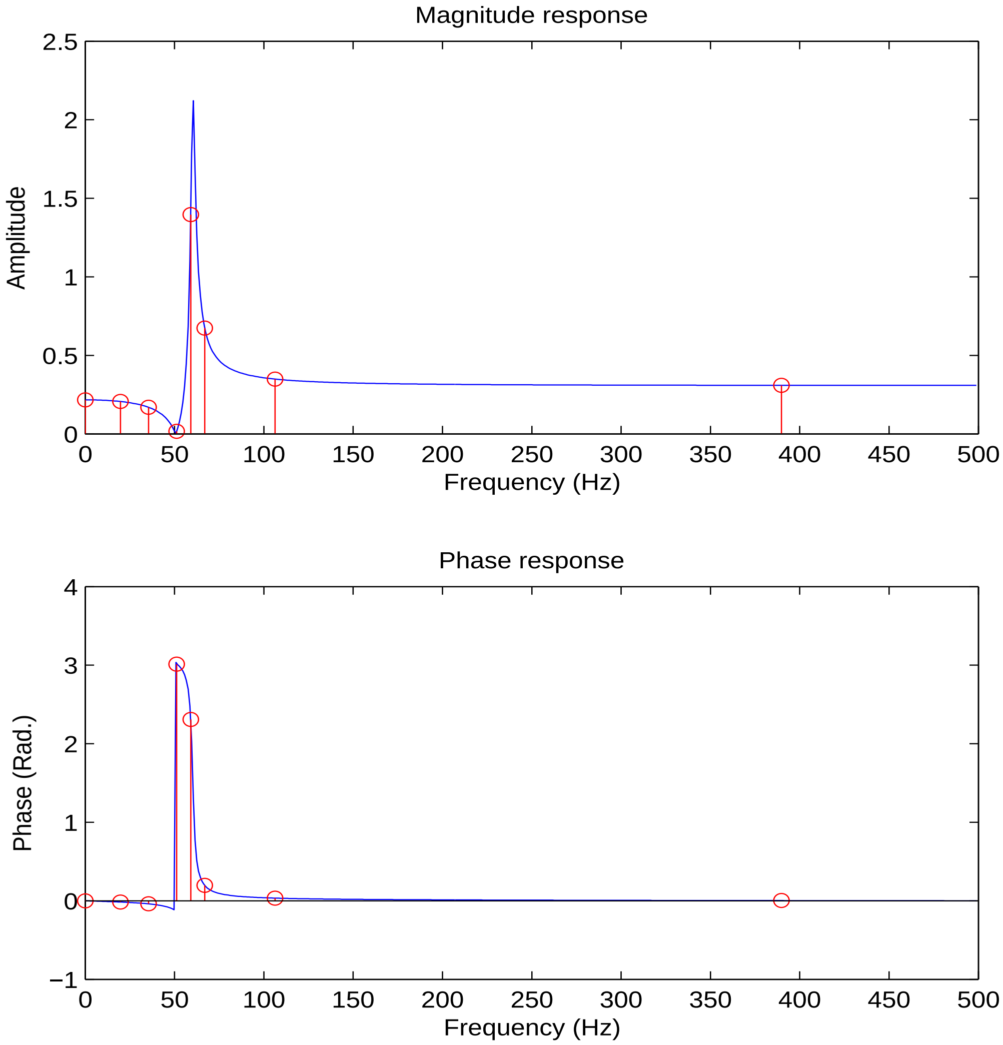

Figure 1: Results for the problem described in Table 1. The desired responses are shown in stem–plot form and are from Table 1; while the continuous lines show the results of our least–squares program.

Table 1: Table of input samples of magnitude responses and phases at various frequencies. These will be used in a least–squares program to determine the filter parameters with $N = D = 2$

f_m (Hz)	A_m	ϕ_m (Rad.)
000.0000	0.2172	+0.0000
019.6850	0.2065	-0.0156
035.4331	0.1696	-0.0383
051.1811	0.0164	+3.0125
059.0551	1.3959	+2.3087
066.9291	0.6734	+0.1961
106.2990	0.3490	+0.0343
389.7640	0.3095	+0.0031

2. Provide a comparison of this method with the bilinear–transform method described on page 434 of Ambardar's book.

3. **Due date**: `xxx` May 2008 in class. Your submissions should be neatly written (or typed). Make all assumptions clear to the reader and show all steps in derivations.

MATLAB - 85%

1. Write a MATLAB function `function [a, b] = LSQFilterDesign(f, A, phi, NEq, ts, N, D)` that solves the least–squares problem. The input parameters are `f` = frequency array, `A` = amplitude array, `phi` = phase array, `ts` = sampling rate, `NEq` = length of `f, A` or `phi`, `N` = number of coefficients b_i in the numerator polynomial, `D` = number of coefficients a_i in the denominator polynomial. The outputs are `a` and `b` – the coefficients of denominator and numerator polynomials respectively of the transfer function. Your file name should be the same as the function name, i.e., `LSQFilterDesign.m`.

2. Write a calling program `Project03Spring2008Main.m` that calls `LSQFilterDesign.m` with the appropriate inputs, receives the outputs and plots a graph like that shown in Figure 1.

3. **Due date**: xxx May 2008 by email only.

Remember, points will be taken off for not following the format described above, regardless of whether you get the right results or not. Late submissions will not be accepted and will be given a grade of zero points.