**Lab III Digital Filters**

Fall 2019

## Introduction

This lab concentrates on the design and realization of typical digital filters, e.g. FIR filters, IIR filters, by using the classical methods and the MATLAB tools. Then these filters are applied to filtering signals in some ways. After that, an application of filtering sound is provided as a practice.

## Design an IIR filter with bilinear transform

IIR filter is normally designed from common analog filters, e.g. Butterworth, Chebyshev, Elliptic, which are named prototype filters. The principle of the IIR filter design is to keep the frequency response as similar as that of the corresponding prototype analog filter.

A simple way to design the IIR filter is the bilinear transform (known as Tustin's method), which is used in digital signal processing and discrete-time control theory to transform a continuous-time system to the corresponding discrete-time, and vice versa, referring to reference book.

* 1. Develop an analog 4-order Chebyshev I style low-pass filter, with the cutoff frequency at *fc* = 10 using **cheby1**. Show **as** and **bs** as the denominator and numerator coefficients of the analog filter.
  2. Given the sampling frequency *fs* = 1000, using **cheby1** (bilinear transform) to convert the analog prototype filter into an IIR filter. Show **az** and **bz** as the denominator and numerator coefficients of the digital filter.
  3. Use **freqs** and **freqz** to obtain and compare the FRFs of these two analog and digital filters.

## Design an FIR filter with the Window method

A simple way to design FIR filter is to use window function truncate the impulse response function of an ideal filter, named the window method.

You may use the following parameters, the normalized cut-off frequency *fc* = 0.1.

* 1. Design the FIR filters with the rectangle window, where the window length *N* = -20~20 and -60~60, respectively. Calculate the corresponding FRFs of the FIR filters. And then compare their impulse response functions and FRFs in terms of different window length.
  2. Design the FIR filters with rectangle window and the Kaiser window with the window length *N* = 0 ~ 60. And then compare their impulse response functions and FRFs in terms of the window type.
  3. Transform the low-pass filter to high-pass filter with Eq.1



## Pole/Zero Designs

Pole/zero placement can be used to design simple filters, such as first-order smoothers, notch filters, and resonators. To illustrate the technique, we design the transfer function for notch filter as



In the equation, zeros are at  and there are corresponding poles .

A biomedical signal, sampled at a rate of 250 Hz, is plagued by 50 Hz power frequency interference and its harmonics 100 Hz.

* 1. Design a digital notch filter *H*(*z*) that removes all these harmonics and remains essentially flat at other frequency with bandwidth 10Hz. Plot the poles and zeros with the unit circle for reference. Then show its FRFs.
  2. Design a comb filter to keep the 50 Hz with 20 Hz bandwidth. Plot the poles and zeros with the unit circle for reference. Then show its FRFs.

## Filter Design and Analysis (FDA) tool

We can use FDA tool to design a filter in MATLAB. This tool provides us many ways to design the filter we want. In addition, we can also observe the parameters of the filter by this tool.

A steady-state time sequence *p*[*n*] can be expressed as the summation of harmonics *ui*:



where the magnitude, frequency and phaseare *Ai*, , and , respectively，as listed in Table 2.

Table 2 Parameters

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Parameters | *i* = 1 | | | *i* = 2 | | | *i* = 3 | | |
|  | given | calculated | error | given | calculated | error | given | calculated | error |
|  | 16 |  |  | 5 |  |  | 8 |  |  |
|  | 2π/8 |  |  | 2π/7 |  |  | 2π/4 |  |  |
|  | π/8 | —— | —— | π/2 | —— | —— | 0 | —— | —— |

* 1. Design an FIR filter H1 with Kaiser window, with the following requirements: keep *u*2 unchanged as much as possible and eliminate *u*1 and*u*3 at least 40 Db in magnitude.

1. Show impulse response function and FRF of H1.
2. Indicate the magnitudes (dB) and phases (radians) of H1 at , and .
3. Filter the limit-duration signal ***p*** by H1. Compare before and after filtering signal in both the time-domain and frequency-domain.
   1. Design an IIR filter H2 with Butterworth proto, with the following requirements: keep *u*1 and *u*3 unchanged as much as possible and eliminate *u*2 at least 40 dB in magnitude.
4. Show step response function and FRF of H2.
5. Indicate the magnitudes (dB) and phases (radians) of H2 at , and .
6. Filter the limit-duration signal ***p*** by H2. Compare before and after filtering signal in both the time-domain and frequency-domain.

4.3 Compare the characteristics of FIR filters and IIR filters