

GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL of ELECTRICAL and COMPUTER ENGINEERING
EE 2200 Winter 1999
Lab #7: Frequency Response & Bandpass Filters

Date: week of 22 Feb 1999

This is *the official* Lab #7 description; it is based on Lab A of Lab C.7 in Appendix C of the text, but the warm-up has been changed quite a bit.

The Warm-up section of each lab must be completed in Lab and the steps marked *Instructor Verification* must also be signed off **during the lab time**.

The lab report for this lab will be informal: discuss your results from section 4. Staple the **Instructor Verification** sheet to the end of your lab report.

The report will due during the week of **1-March** at the start of your lab.

1 Introduction

The goal of this lab is to study the response of FIR filters to inputs such as complex exponentials and sinusoids. In addition, we will use FIR filters to study properties such as linearity and time-invariance. In the experiments of this lab, you will use `firfilt()`, or `conv()`, to implement filters and `freqz()` to obtain the filter's frequency response.¹ As a result, you should learn how to characterize a filter by knowing how it reacts to different frequency components in the input.

This lab also introduces a practical application where sinusoidal signals are used to transmit information: a touch-tone dialer. Bandpass FIR filters can be used to extract the information encoded in the waveforms.

1.1 Frequency Response of FIR Filters

The output or *response* of a filter for a complex sinusoid input, $e^{j\hat{\omega}n}$, depends on the frequency, $\hat{\omega}$. Often a filter is described solely by how it affects different frequencies—this is called the *frequency response*.

For example, the frequency response of the two-point averaging filter

$$y[n] = \frac{1}{2}x[n] + \frac{1}{2}x[n-1]$$

can be found by using a general complex exponential as an input and observing the output or response.

$$x[n] = Ae^{j\hat{\omega}n + \phi} \tag{1}$$

$$y[n] = \frac{1}{2}Ae^{j\hat{\omega}n + \phi} + \frac{1}{2}Ae^{j\hat{\omega}(n-1) + \phi} \tag{2}$$

$$= Ae^{j\hat{\omega}n + \phi} \frac{1}{2} \left\{ 1 + e^{-j\hat{\omega}} \right\} \tag{3}$$

In (3) there are two terms, the original input, and a term that is a function of $\hat{\omega}$. This second term is the frequency response and it is commonly denoted by $H(e^{j\hat{\omega}})$.²

$$H(e^{j\hat{\omega}}) = \mathcal{H}(\hat{\omega}) = \frac{1}{2} \left\{ 1 + e^{-j\hat{\omega}} \right\} \tag{4}$$

¹If you are working at home and do not have the function `freqz.m`, there is a substitute available called `freqz.m`. You can get it from the EE-2200 WebCT page.

²The notation $H(e^{j\hat{\omega}})$ is used in place of $\mathcal{H}(\hat{\omega})$ for the frequency response because we will eventually connect this notation with the z -transform, $H(z)$, in Chapter 7.

Once the frequency response, $H(e^{j\hat{\omega}})$, has been determined, the effect of the filter on any complex exponential may be determined by evaluating $H(e^{j\hat{\omega}})$ at the corresponding frequency. The output signal $y[n]$, will be a complex exponential whose complex amplitude has a constant magnitude and phase. The phase describes the phase change of the complex sinusoid and the magnitude describes the gain applied to the complex sinusoid.

The frequency response of a general FIR linear time-invariant system is

$$H(e^{j\hat{\omega}}) = \mathcal{H}(\hat{\omega}) = \sum_{k=0}^M b_k e^{-j\hat{\omega}k} \quad (5)$$

MATLAB has a built-in function for computing the frequency response of a discrete-time LTI system. The following MATLAB statements show how to use `freqz` to compute and plot the magnitude (absolute value) of the frequency response of a two-point averaging system as a function of $\hat{\omega}$ in the range $-\pi \leq \hat{\omega} \leq \pi$:

```
bb = [0.5, -0.5];           %-- Filter Coefficients
ww = -pi:(pi/100):pi;      %-- omega hat
H = freqz(bb, 1, ww);      %--freqz.m is an alternative
subplot(2,1,1);
plot(ww, abs(H))
subplot(2,1,2);
plot(ww, angle(H))
xlabel('Normalized Radian Frequency')
```

For FIR filters, the second argument of `freqz(-, 1, -)` must always be equal to 1. The frequency vector `ww` should cover an interval of length 2π for $\hat{\omega}$, and its spacing must be fine enough to give a smooth curve for $H(e^{j\hat{\omega}})$. Note: we will always use capital H for the frequency response.

1.2 Periodicity of the Frequency Response

The frequency responses of discrete-time filters are *always* periodic with period equal to 2π . Explain why this is the case by stating a definition of the frequency response and then considering two input sinusoids whose frequencies are $\hat{\omega}$ and $\hat{\omega} + 2\pi$.

$$x_1[n] = e^{j\hat{\omega}n} \quad \text{versus} \quad x_2[n] = e^{j(\hat{\omega} + 2\pi)n}$$

Consult Chapter 6 for a mathematical proof that the outputs from each of these signals will be identical (basically because $x_1[n]$ is equal to $x_2[n]$.)

The implication of periodicity is that a plot of $\mathcal{H}(\hat{\omega})$ only needs to extend over the interval $-\pi \leq \hat{\omega} \leq \pi$.

2 Warm-up

2.1 Frequency Response of the Three-Point Averager

In Chapter 6 we examined filters that average input samples over a certain interval. These filters are called “running average” filters or “averagers” and they have the following form:

$$y[n] = \frac{1}{M+1} \sum_{k=0}^M x[n-k] \quad (6)$$

- (a) Use Euler’s formula and complex number manipulations to show that the frequency response for the 3-point running average operator is given by:

$$H(e^{j\hat{\omega}}) = \mathcal{H}(\hat{\omega}) = \frac{2 \cos \hat{\omega} + 1}{3} e^{-j\hat{\omega}} \quad (7)$$

- (b) Implement (7) directly in MATLAB. Use a vector that includes 400 samples between $-\pi$ and π for $\hat{\omega}$. Since the frequency response is a complex-valued quantity, use `abs()` and `angle()` to extract the magnitude and phase of the frequency response for plotting. Plotting the real and imaginary parts of $H(e^{j\hat{\omega}})$ is not very informative.
- (c) In this part, use `freqz.m` in MATLAB to compute $H(e^{j\hat{\omega}})$ numerically (from the filter coefficients) and plot its magnitude and phase versus $\hat{\omega}$. Write the appropriate MATLAB code to plot both the magnitude and phase of $H(e^{j\hat{\omega}})$. The filter coefficient vector for the 3-point averager is defined via:

```
bb = 1/3*ones(1,3);
```

Note: the function `freqz(bb,1,ww)` evaluates the frequency response for all frequencies in the vector `ww`. It uses the summation in (5), not the formula in (7). The filter coefficients are defined in the assignment to vector `bb`. How do your results compare with part (b)?

Instructor Verification (separate page)

2.2 The MATLAB FIND Function

Often signal processing functions are performed in order to extract information that can be used to make a decision. The decision process inevitably requires logical tests, which might be done with `if`, `then` constructs in MATLAB. However, MATLAB permits vectorization of such tests, and the `find` function is one way to do lots of tests at once. Run the following example to see how `find` works:

```
xx = 1.4:0.33:5, jkl = find(round(xx)==3), xx(jkl)
```

The argument of the `find` function can be any logical expression. See `help` on `relop` for information. Now, suppose that you have a frequency response:

```
ww = -pi:(pi/100):pi; H = freqz(ones(1,10), 1, ww);
```

Use the `find` command to determine the indices where `H` is zero, and then use those indices to display the list of frequencies where `H` is zero. Since there might be round-off error in calculating `H`, the logical test should probably be a test for those indices where the magnitude of `H` is less than some rather small number, e.g., 1×10^{-6} . Explain your answer to the TA by also plotting the magnitude and phase of `H`.

Instructor Verification (separate page)

3 Lab Exercises

3.1 Cascading Two Systems

More complicated systems are often made up from simple building blocks. In the system below two FIR filters are connected “in cascade.”

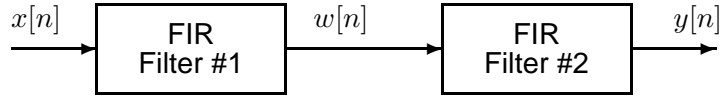


Figure 1: Cascade of two FIR filters.

- (a) First, assume that the above system is described by the two equations

$$w[n] = \sum_{\ell=0}^M \alpha^\ell x[n - \ell] \quad (\text{FIR FILTER \#1})$$

$$y[n] = w[n] - \alpha w[n - 1] \quad (\text{FIR FILTER \#2})$$

Use MATLAB to get the frequency responses for the case where $\alpha = 0.8$ and $M = 9$. Plot the magnitude and phase of the frequency responses for Filter #1 and also for Filter #2. Which one of these filters is a *lowpass filter*.

- (b) Plot the magnitude and phase of the frequency response of the overall cascaded system.
- (c) Explain how the individual frequency responses in part(a) are combined to get the overall frequency response in part(b). Comment on the magnitude combinations as well as the phase combinations.

3.2 Deconvolution

In the previous lab, the two filters from Section 3.1 were used in an image deblurring experiment. You should now re-interpret how that experiment worked by explaining what happens in the frequency domain

- (a) If a single filter has a frequency response $\mathcal{H}(\hat{\omega}) = 1$, how is the output of the filter $y[n]$ related to the input $x[n]$?
- (b) If a single filter has a frequency response $\mathcal{H}(\hat{\omega}) = e^{-jr\hat{\omega}}$, where r is an integer, how is the output of the filter $y[n]$ related to the input $x[n]$?
- (c) A “deconvolved” output should look just like the input prior to blurring. If Filter #1 (in Fig. 1) has a frequency response $\mathcal{H}_1(\hat{\omega})$ and Filter #2 is $\mathcal{H}_2(\hat{\omega})$, state a general condition on the frequency responses so that Filter #2 will “deconvolve” the effect of Filter #1. Use the ideas in Parts (a) and (b).
- (d) Explain how well the filters for the case $\alpha = 0.8$ and $M = 9$ work for deconvolution.

3.3 Nulling Filters for Rejection

Nulling filters are filters that completely eliminate some frequency other than $\hat{\omega} = 0$ or $\hat{\omega} = \pi$. It is possible to make a nulling filter with as few as three coefficients. If $\hat{\omega}_n$ is the desired nulling frequency, then the following length-3 FIR filter

$$y[n] = x[n] - 2 \cos(\hat{\omega}_n)x[n - 1] + x[n - 2] \quad (8)$$

will have a zero at $\hat{\omega} = \hat{\omega}_n$. For example, a filter designed to completely eliminate signals of the form $Ae^{j0.5\pi n}$ would have coefficients

$$b_0 = 1, \quad b_1 = -2 \cos(0.5\pi) = 0, \quad b_2 = 1.$$

- (a) Design a filtering system that consists of the cascade of two nulling filters that will eliminate the following frequencies: $\hat{\omega} = 0.1\pi$, and $\hat{\omega} = 0.4\pi$. For this part, derive the filter coefficients.
- (b) Generate an input signal $x[n]$ that is the sum of three sinusoids:

$$x[n] = 3 \cos(0.1\pi n) + 5 \cos(0.25\pi n + \pi/4) + 2 \cos(0.4\pi n - \pi/3)$$

Make the input signal 150 samples long.

- (c) Filter the sum of three sinusoids signal through the filters designed in part (a). Show the MATLAB code that you wrote to implement the cascade of two FIR filters.
- (d) Make a plot of the output signal—show the first 40 points. Determine the exact mathematical formula (magnitude, phase and frequency) for the output signal for $n \geq 5$. Show that the plot of the output signal matches this formula for most of the first 40 points.
- (e) Explain why the output signal is different for the first few points. How many “start-up” points are found, and how is this number related to the lengths of the filters designed in part (a)? Hint: consider the length of a single FIR filter that is equivalent to the cascade of two length-3 FIRs.

3.4 Simple Bandpass Filter Design

The L -point averaging filter is a lowpass filter. Its passband width is controlled by L , because it is inversely proportional to L . It is also possible to create a filter whose passband is centered around some frequency other than zero. One simple way to do this is to define the impulse response as:

$$h[n] = \frac{2}{L} \cos(\hat{\omega}_c n), \quad 0 \leq n < L$$

where L is the filter length, and $\hat{\omega}_c$ is the center frequency. that defines the frequency location of the passband, e.g., we pick $\hat{\omega}_c = 0.1\pi$ if we want the peak of the filter’s passband to be at 0.1π . The bandwidth of the bandpass filter is controlled by L ; the larger the value of L , the narrower the bandwidth. This particular filter is also discussed in the section on useful filters in Chapter 7.

- (a) Generate a bandpass filter that will pass a frequency component at $\hat{\omega} = 0.25\pi$. Make the filter length (L) long enough so that it will also *greatly* reduce frequency components at $\hat{\omega} = 0.1\pi$ and $\hat{\omega} = 0.4\pi$.
- (b) Filter the “sum of 3 sinusoids” signal from the previous section. Make a plot of 40 points of the input and output signals, and describe how the filter has reduced or removed two of the three sinusoidal components.
- (c) Make a plot of the frequency response (magnitude only) for the filter from part (a). In addition, plot the filter coefficients in the in the same figure, using subplot and the `stem()` function.
- (d) The *passband* of the BPF filter is defined by the region of the frequency response where $|\mathcal{H}(\hat{\omega})|$ is close to one. Typically, the passband width is defined as the length of the frequency region where $|\mathcal{H}(\hat{\omega})|$ is greater than $1/\sqrt{2} = 0.707$.

The *stopband* of the BPF filter is defined by the region of the frequency response where $|\mathcal{H}(\hat{\omega})|$ is close to zero. In this case, it is reasonable to define the stopband as the region where $|\mathcal{H}(\hat{\omega})|$ is less than 0.2.

Explain how the width of the passband is related to filter length L . Comment on the selectivity of the bandpass filter, i.e., use the frequency response to explain how the filter can pass one component at $\hat{\omega} = 0.25\pi$, while rejecting the others at $\hat{\omega} = 0.1\pi$ and $\hat{\omega} = 0.4\pi$.

Lab #7

EE-2200

Winter-1999

INSTRUCTOR VERIFICATION PAGE

Staple this page to the end of your Lab Report.

Name: _____

Date of Lab: _____

Part 2.1 Frequency response of a 3-point averager:

Verified: _____

Date/Time: _____

Part 2.2: Using the MATLAB `find` function to locate frequency response features automatically:

Verified: _____

Date/Time: _____