GEORGIA INSTITUTE OF TECHNOLOGY

SCHOOL of ELECTRICAL and COMPUTER ENGINEERING

ECE 2026 Fall 2013

Lab #9: Frequency Response

Date: 3 - 6 November 2013

Each Lab assignment in ECE2026 consists of three parts: Pre-Lab, In-lab Tasks, and Take-home Questions. It requires you to come into lab prepared. Be sure to read the entire lab carefully before arriving.

Pre-Lab: You should read the Pre-Lab section of the lab and go over all exercises in this section before going to your assigned lab session. Although you do not need to turn in results from the Pre-Lab, doing the exercises therein will help make your in-lab experience more rewarding and go more smoothly with less frustration and panicking.

In-lab Tasks and Verification: There are a number of designated tasks for each student to accomplish during the lab session. Students are encouraged to read, prepare for and even attempt at these tasks beforehand. These tasks must be completed **during your assigned lab time** and the steps marked *Instructor Verification* must also be signed off **during the lab time**. One of the laboratory instructors must verify the appropriate steps by signing on the **Instructor Verification** line. When you have completed a step that requires verification, simply put a plastic cup on top of your PC and demonstrate the step to one of the TAs or the professor. (You can also use the plastic cups to indicate if you have a more general question, i.e. you can use it to get our attention even if you don't have an Instructor Verification ready.)

Take-home Questions: At the end of each lab sheet below all the verification steps, several questions are to be answered by the student, who can choose to complete the answers while in the lab or after leaving the lab.

The lab sheet with all verification signatures and answers to the questions needs to be turned in to the Lab-grading TA at the beginning of the next lab session.

---

1 Introduction

The goal of this lab is to study the frequency response. For FIR filter, this is the response to inputs such as complex exponentials and sinusoids. You can use **`firfilt()`**, or **`conv()`**, to implement filtering and **`freqz()`** to obtain the filter's frequency response[1]. You will learn how to characterize a filter by the way it reacts to sinusoidal components of different frequencies in the input.

2 Pre-Lab

2.1 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 input frequencies—this is called the *frequency response*. For example, the frequency response of the two-point averaging filter

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

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

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

$$y[n] = \tfrac{1}{2}e^{j(\hat{\omega}n+\varphi)} + \tfrac{1}{2}e^{j(\hat{\omega}(n-1)+\varphi)} \tag{2}$$

$$= Ae^{j(\hat{\omega}n+\varphi)} \bullet \tfrac{1}{2}\left\{1+e^{-j\hat{\omega}}\right\} \tag{3}$$

---

[1] If you are working at home and do not have the function **`freqz.m`**, there is a substitute available called **`freekz.m`**. You can find it in the *SP-First Toolbox*.

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[2] by $H(e^{j\hat{\omega}})$:

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

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 of $H(e^{j\hat{\omega}})$ describes the phase change of the complex sinusoid and the magnitude of $H(e^{j\hat{\omega}})$ describes the gain applied to the complex sinusoid.

The general form of the frequency response for an M^{th} order FIR linear time-invariant system with filter coefficients $\{b_k\}$ is

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

2.1.1 MATLAB Function for Frequency Response

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 both the magnitude (absolute value) and the phase of the frequency response of a two-point averaging system as a function of $\hat{\omega}$ in the range $-\pi \le \hat{\omega} \le \pi$:

```
bb = [0.5, 0.5];              %--Filter Coefficients
ww = -pi:(pi/100):pi;         %--omega hat frequency vector
H = freqz(bb, 1, ww);         %<--freekz(bb,1,ww) 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[3].

2.2 Periodicity of the Frequency Response

The frequency responses of discrete-time filters are *always* periodic with period equal to 2π. This can be easily verified by using the definition of the frequency response (5) and then considering two input sinusoids whose frequencies are $\hat{\omega}$ and $\hat{\omega}+2\pi$. That is, use

$$H(e^{j\hat{\omega}}) = \sum_{k=0}^{M} b_k e^{-j\hat{\omega}k} = \sum_{k=0}^{M} b_k e^{-j\hat{\omega}k} e^{-j2\pi k} = H(e^{j(\hat{\omega}+2\pi)})$$

Consult Chapter 6 for more detailed mathematical proof. **The implication of periodicity is that a plot of $H(e^{j\hat{\omega}})$ only has to be made over the interval $-\pi \le \hat{\omega} \le \pi$.**

2.3 Frequency Response of the Four-Point Averager

---

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

[3] If the output of the **freqz** function is not assigned, then plots are generated automatically; however, the magnitude is given in decibels which is a logarithmic scale. For linear magnitude plots a separate call to plot is necessary.

In Chapter 6 we examined filters that compute the average of input samples over an interval. These filters are called "running average" filters or "averagers" and they have the following form for the L-point averager:

$$y[n] = \frac{1}{L}\sum_{k=0}^{L-1} x[n-k] \tag{6}$$

(a) Use Euler's formula and complex number manipulations to show that the frequency response for the 4-point running average operator is given by:

$$H(e^{j\hat{\omega}}) = \left(\frac{2\cos(0.5\hat{\omega}) + 2\cos(1.5\hat{\omega})}{4}\right)e^{-j1.5\hat{\omega}} = C(\hat{\omega})e^{j\psi(\hat{\omega})} \tag{7}$$

(b) Implement (7) directly in MATLAB. Use a vector that includes 400 samples covering the interval $[-\pi, \pi)$ for $\hat{\omega}$. Make plots of $C(\hat{\omega})$ and $\psi(\hat{\omega})$ versus $\hat{\omega}$, but keep in mind that these are not necessarily plots of the magnitude and phase. You would have to use **`abs()`** and **`angle()`** to extract the magnitude and phase of the frequency response for plotting.

(c) In this part, use **`freqz.m`** or **`freekz.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 $H(e^{j\hat{\omega}})$. Follow the example in Section 2.1.1. The filter coefficient vector for the 4-point averager is defined via:

```
bb = 1/4*ones(1,4);
```

Recall that 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)?

Note: the plots should not be identical, but you should be able to explain why they are equivalent by converting the minus sign in the negative values of $C(\hat{\omega})$ to a phase of π radians.

2.4 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 determine which elements of a vector meet a certain logical criterion. In the following example, **`find`** extracts all the numbers that "round" to 3:

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

The argument of the **`find`** function can be any logical expression, and **`find`** returns a list of indices where that logical expression is true, and **`xx(jkl)`** gives the values that round to 3. See **`help`** on **`relop`** for more information. Now, suppose that you have a frequency response:

```
ww = -pi:(pi/500):pi; HH = freqz( 1/4*ones(1,4), 1, ww );
```

Use the **`find`** command to determine the indices where **`HH`** is zero, or very small, e.g., $|H(e^{j\hat{\omega}})| \le 1\times10^{-8}$. Then use those indices to display the list of frequencies where **`HH`** is "zero." Since there might be round-off error in calculating **`HH`**, the logical test should be a test for those indices where the magnitude (absolute value in MATLAB) of HH is less than 1×10^{-8}. Compare your answer to the frequency response that you plotted for the four-point averager in Section 2.3.

3.0 Lab Exercise

The first objective of this warm-up is to use a MATLAB GUI to demonstrate the frequency response. If you are working in the ECE lab it is NOT necessary to install the GUI; otherwise, you should install the *SP-First* toolbox. The frequency response demo, **`dltidemo`**, is part of the *SP-First Toolbox*.

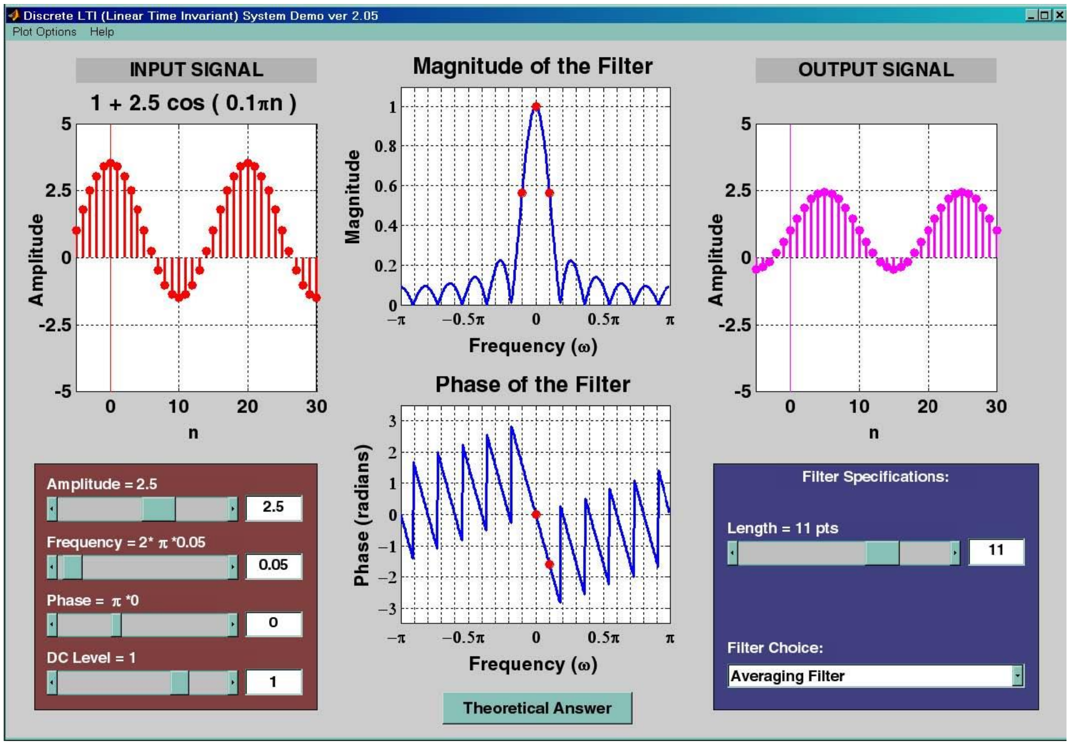

Figure 1: DLTI demo interface. The frequency label is ω because MATLAB won't display the hat in $\hat{\omega}$.

3.1 LTI Frequency Response Demo

The **`dltidemo`** GUI illustrates the "sinusoid-IN gives sinusoid-OUT" property of an LTI system. In this demo, you can change the amplitude, phase and frequency of an input sinusoid, $x[n]$, and you can change the digital filter that processes the signal. Then the GUI will show the output signal, $y[n]$, which is also a sinusoid (at the same frequency). Figure 1 shows the interface for the **`dltidemo`** GUI. It is possible to see the formula for the output signal, if you click on the **`Theoretical Answer`** button located at the bottom-middle part of the window. The digital filter can be changed by choosing different options in the **`Filter Specifications`** box in the lower right-hand corner. In this exercise, you should perform the following steps with the **`dltidemo`** GUI:

(a) Set the input to $x[n] = 4\cos(0.3\pi(n-4))$; note that this sinusoid has a peak at $n = 4$. (Also note: The GUI only allows positive phase values between 0 and 2π; you need to make proper conversion when the phase is specified as a negative value.)

(b) Set the digital filter to be a 5-point averager.

(c) Determine the formula for the output signal and write it in the form: $y[n] = A\cos(\hat{\omega}(n-n_d))$.

(d) Using n_d from $y[n]$ and the fact that the input signal had a peak at $n = 4$, determine how much the peak of the cosine wave has shifted. This is called the *delay* through the filter. Give an equation that explains how the delay is related to the phase of $H(e^{j\hat{\omega}})$ at $\hat{\omega} = 0.3\pi$.

Instructor Verification (separate page)

(e) Now, change the length of the averaging filter so that the output will be zero, i.e., $y[n] = 0$. Find such a filter with the shortest impulse response. Use the GUI to show that you have the correct filter to zero the output. If the filter length is more than 15, you will have to enter the "**`Filter Specifications`**" with the **`user Input`** option.

Hint: Recall the Dirichlet form for the frequency response of the averaging filter, and where it is zero.

(f) When the output is zero, the filter acts as a *Nulling Filter*, because it eliminates the input at a particular frequency, here $\hat{\omega} = 0.3\pi$. Sinusoidal components at other frequencies $\hat{\omega}$ will also be nulled – write an expression for these frequencies for the filter you just specified.

Instructor Verification (separate page)

3.2 Ideal Filters and Practical Filters

The running average FIR filter is a lowpass filter, but it is not a very good one. Better filters can be designed with computer-aided optimization algorithms. In **`dltidemo`**, it is possible to choose:

Ideal Filters which are given by their frequency response, but are not actually FIR filters. In other words, there are no filter coefficients that will produce the ideal frequency response(s). In the GUI, you can find ideal lowpass filters (LPF), highpass filters (HPF) and bandpass filters (BPF). The ideal LPFs and HPFs have one parameter for the cutoff frequency which determines the boundary between the passband and stopband. The ideal BPF has a parameter for center frequency which determines where the band is located. All the ideal filters have an additional parameter for the slope of the phase of $H(e^{j\hat{\omega}})$.

Practical Filters which are approximations to the ideal filters by a length-L FIR filter. The ones shown in **`dltidemo`** were designed using MATLAB's **`fir1`** function for filter design. The GUI has length-15 LPFs and HPFs, and length-21 BPFs. The LPF and HPF designs have one parameter for the cutoff frequency which determines the boundary between the passband and stopband. The BPF has a parameter for center frequency which determines where the band is located.

Notation: the cutoff frequency for HPFs and LPFs will be called $\hat{\omega}_c$.

(a) Define the input to be $x[n] = 3.3\cos(0.5\pi n)$.

(b) Set the filter to be an ideal HPF with a cutoff frequency of $\hat{\omega}_c = 2\pi(0.2)$. Determine a value for the phase slope so that the output will be $y[n] = 3.3\cos(0.5\pi n - \pi)$.

(c) Still using the ideal HPF, determine a cutoff frequency ($\hat{\omega}_c$) so that the output will be zero.

Instructor Verification (separate page)

(d) Now switch to the length-15 HPF, using the same input signal. Determine the output when the cutoff frequency is $2\pi(0.2)$. Is the signal still *passed* by the HPF?

(e) Again with the length-15 HPF, use the cutoff frequency ($\hat{\omega}_c$) from part (c) and determine the formula for the output signal. Why is the output not exactly zero?

(f) Determine a cutoff frequency for the length-15 HPF so that the output signal is nearly zero. Give the formula for the output signal, and compare the cutoff frequency ($\hat{\omega}_c$) to part (c).

Instructor Verification (separate page)

Lab #9
ECE-2026 Fall-2013
Lab Sheet

For each verification, be prepared to explain your answer and respond to other related questions that the lab TA's or professors might ask. Turn this page in at the beginning of your next lab period.

Name: ________________________________ Date of Lab: _________________________

Part	Observations
3.1(c)	Formula for output from length-5 averager as $y[n] = A\cos(\hat{\omega}_0(n - n_d))$
3.1(d)	Give an equation that explains how the filter's *delay* is related to the phase of $H(e^{j\hat{\omega}})$ at $\hat{\omega}$ = 0.3π.

Verified: ______________________________ Date/Time: ___________________________________

Part	Observations
3.1(e)	Length L of running average FIR filter that nulls $x[n] = 4\cos(0.3\pi(n-4))$
3.1(f)	Expression of frequencies nulled by the running-average FIR filter in part (e).

Verified: ______________________________ Date/Time: ___________________________________

Part	Observations
3.2(b)	Phase slope for Ideal HPF to get $y[n] = 3.3\cos(0.5\pi n - \pi)$
3.2(c)	Cutoff frequency for Ideal HPF to get $y[n]$ = 0.

Verified: ______________________________ Date/Time: ___________________________________

Part	Observations
3.2(d)	Output of length-15 HPF with $\hat{\omega}_c$ = 0.4π. Give a formula, i.e., $A\cos(\hat{\omega}_0 n + \varphi)$
3.2(e)	Output formula when using cutoff frequency from part (c). Explain why $y[n]$ is not exactly zero.
3.2(f)	Cutoff frequency for length-15 HPF to make output nearly zero. Give formula for $y[n]$.

Verified: ______________________________ Date/Time: ___________________________________