University of Washington
Department of Electrical Engineering
EE 235 Lab 3:
Echo Convolution for LTI Systems

In this lab, we will investigate convolution, starting with using unit impulses and convolution to implement time delay, and then learn how convolution can be used to decode transmitted messages in digital communications. This lab will also be used to review functions.

Important Concepts From Lecture You Will Use In Lab 3

- Describing LTI systems using the impulse response **h(t)**
- Computing the output **y(t)** of an LTI system using the echo property of convolution

What is expected from you in Lab 3

- Completion of 3 pre-lab exercises (3 points)
- Completion of 2 in-lab check offs with TA (4 points)
- Completion of a lab report (3 points)

Note: all pre-lab exercises must be completed by each student individually and turned in at the beginning of the lab. Each student will be checked off by the TA during the lab section.

PRE-LAB:

Read the Matlab Lab 3 tutorial, which reviews and introduces concepts as indicate below. This material will be needed to answer the prelab questions and in the lab.

Matlab Concepts/Functions To Review	New Matlab Concepts/Functions You Will Learn
• Using the zeros and ones functions • Changing value(s) in a vector • Determining number of samples needed • Creating time samples vector • Relationship between index and time • Basic plotting functions & using subplot	• Graphing signals on one plot • Changing the color of a graph • Displaying a legend on the figure • Using the conv function

1) Consider the impulse response: $h_1(t)=\delta(t)-\delta(t-3)+2\delta(t-4)$. This will create a series of echoes. We will create a discrete-time version over the range $0 \le t \le 6$ seconds.

   a) Suppose we wish to create a vector of zeros using a variable sampling rate **Fs**. If we want the vector to have a duration of 6 seconds (starting at time t = 0), what is the formula for the number of samples **nz** we need in terms of **Fs**?

   b) At what index in that vector should we insert a 1 in order to implement $\delta(t)$? What and where would you insert values for the other two impulse functions? Give the formula for the index in terms of **Fs**.

2) Consider the signal $x_1(t)=u(t)-u(t-1.2)$. We will create a discrete-time version **x1** over the range $0 \le t \le 6$ seconds.

a) Find the number of ones **n1** that you need for a pulse that lasts from [0,1.2] using sampling rate **Fs**. How many additional zeroes do you need in order for **x1** to have a total length of 6 seconds? Write the Matlab code that you would need to implement this, using the zeros and ones functions. (There are multiple ways you can solve this.)

b) Sketch the output of the LTI system described by $h_1(t)$ if the input is $x_1(t)$.

3) When you use the **y=conv(x,h)** command in Matlab, the length of the output vector **y** will be longer than the input vectors. You may want to plot **x, y** and **h** together on the same time axis. Specify the Matlab command you would use to extract the portion of **y** that corresponds to the same time region as the inputs assuming that **x** and **h** are the same length and both start at time 0.

<u>EXERCISE #1: Impulse response with time shifts</u>

In this exercise, we will practice create $h(t)$ from the Prelab part 1 in Matlab. If you recall, in class, a unit impulse **δ(t)** is an ideal signal with very small duration and infinite height. In reality, any spiky signal of short duration and finite amplitude can be used to approximate a unit impulse. For this lab, we will approximate $\mathbf{K\delta(t - t_o)}$ as a vector that has the value **K** at the nearest index where $\mathbf{t = t_o}$, and the values 0 at all other indices in the vector.

<u>Lab Exercise</u>:

a) First, create a new working folder called in **<u>Lab 3</u>** in your U Drive. In your working folder, open up a script file and call it **<u>Ex1.m</u>**. Clear all variables and close all figures.
b) Create 6-second time function **h1** using the **zeros** function, assuming a sampling rate of **Fs** = 8000. Your result should be a <u>column</u> vector.
c) Using your formula from pre-lab, compute the index that corresponds to the location in **h1** of **δ(t),** and assign the value 1 to **h1** at that time index. Repeat for the other two impulses, using for **K** the value that corresponds to the area of the impulses.

<u>EXERCISE #2: Plotting multiple signals on one plot</u>

In this exercise, we will practice plotting different signals together, using the individual time-shifted pulses that you expect to see in $y_1(t)$.

<u>Lab Exercise</u>:

Let us now create and plot the component signals from pre-lab.

a) Create vector **x1** to implement the signal $x_1(t)=u(t)-u(t-1.2)$ for time [0,6] seconds**,** as you proposed in the prelab, part 2.

b) Create the corresponding time vector **t** using the length of **x1** and the sampling rate **Fs**. We will use this time vector for the other two signals as well.

c) Create the signal **x2** that corresponds to the time-shifted signal $x_2(t)= -x_1(t-3)$. You may make use of your code from Lab 2, but make sure that **x2** has the same length and time range as **x1**, since you will plot all signals with the same time vector.

d) Repeat for **x3** where $x_3(t)= 2x_1(t-4)$.

e) We will now graph all three signals on one plot, each with a different color:
   i. Load a new figure window.
   ii. Type in the command **hold on** to allow multiple graphs on one plot.
   iii. Plot **x1 vs. t**, and assign it the color <u>blue</u>. Plot **x2 vs. t**, and assign it the color <u>red</u>. Then, plot **x3 vs. t**, and assign it <u>magenta</u>.
   iv. Adjust the axes so that the time-axis is between 0 and 6, and the amplitude-axis is between -2 and 3. No need to turn on the grid, but title and label your plot.
   v. Using the function **legend**, display a legend on the figure to label the signals as **x1(t)**, **-x1(t - 3)**, and **2x1(t - 4)**.

f) Make sure to comment your script, and then run it. Verify that the plots Matlab produces matches your plots from pre-lab.

<u>Lab Check-Off #1 of 2</u>: Demonstrate your script **<u>Ex1.m</u>** to your lab TA and show that you know how to plot multiple signals in Matlab.

<u>Lab Report Question #1 of 2</u>: A student in class forgets to use the **hold on** command, as well as the line of code to produce a **legend**. Describe the plot that you expect Matlab to produce. Explain your answers.

<u>EXERCISE #3: Convolving signals in Matlab</u>
In this exercise, we will use get the output signal $y_1(t)$ by using addition of the individual components and by convolution.

<u>Lab Exercise</u>:
a) Compute $y_1(t)$ by adding the three signals from exercise 2: **y1a = x1+x2+x3**.

b) Compute $y_1(t)$ using Matlab's conv function: **y1=conv(x1,h1).** Since **y1** will be longer than **x1**, extract out the portion of y1 that corresponds to time [0,6] seconds and set that to be **y1b** for plotting purposes.

c) To see the relationship between the input, impulse response, and output signals, we will use subplots. On a <u>4 x 1</u> subplot, plot the following:
   - **x1 vs. t**
   - **h1 vs. t**
   - **y1a vs. t**
   - **y1b vs. t**

   On all plots, adjust the time-axis to be between 0 and 6 and the y-axis to be between -2 and 3. Do not forget to title and label all three of your plots.

d) Comment your script, and run what you have so far to verify your results from pre-lab.

<u>Lab Check-Off #2 of 2</u>: Demonstrate your script **<u>Ex2.m</u>** to your lab TA and show that you know how to implement an LTI system and can properly use the **conv** function.

<u>Lab Report Question #2 of 2</u>: Explain how the system output would sound if the input was the **train.mat** signal used in Lab 1.