

Laboratory report

**Lab 3: Convolution**



April 25, 2022

**Logo

Description automatically generated**

Instructor: Thai Vu Hien

Class : 19ES

Group 5 : Tran Ngoc Nguyen

Luu Dinh Dai Duc

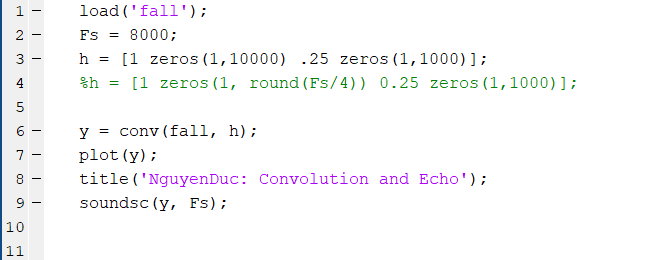
# Introduction

# Some useful MATLAB commands

# Convolution

## Exercise 1: Convolution and Echo

1. **Create a new script for this problem. Download the trumpet jazz lick "fall" here1 , and then load it into MATLAB using load('fall') and plot it. Use whos to see that the variables fall and Fs are created for you. (The sampling rate (Fs) for this signal should be 8000 Hz.)**

****

1. **Use the following commands to convolve the following impulse response h, with the trumpet sound.**

**Text

Description automatically generated**

**Chart, histogram

Description automatically generated**

1. **What if the second echo (in h) is a negative coe‑cient? When you play it, it should not sound different since your ear is not sensitive to that sort of modification (simple phase change).**

**Chart, histogram

Description automatically generated**

* When th sencond echo is a negative coefficient (-0.25), it will not create a difference sound. Because our ear is not sensitive to that sort of modification.

1. **Now let's build a system that delays the echo for a quarter second by inserting Fs/4 zeros before the second impulse:**

Text

Description automatically generated

Chart

Description automatically generated

How do the input and output signals compare in the above step? (Look and listen). Experiment with different numbers of zeros, and try repeating this with some of the built-in MATLAB sounds.

* When we delay the echo for a quarter of sencond (Fs/4), the time this sound running is decrease by 3(from 0.5 s to 1.5s)

## Exercise 2: Convolution and Smoothing

1. **Build a box impulse response**

**Text

Description automatically generated**

**A picture containing chart

Description automatically generated**

1. **How does the output sound different from the input signal?**

* The magnitude of the output signal is clearly smaller than the input signal.

1. **Visually, a difference is that the input signal fall looks like it's centered around value 0, and the system output y2 looks like it's more positive. Let's look more closely. Find the average value of the signal fall (use sum(fall)/length(fall)), and you should see that in fact the fall signal isn't really centered around 0.**

**Text

Description automatically generated with medium confidence**

* The input signal looks like it’s centered around the value 0 and the output signal y2 looks like more positive 

1. **Next, to see what this system does to the input signal, zoom in on part of the signal**

**Text

Description automatically generated**

**A picture containing histogram

Description automatically generated**

* The convolved signal should look a little smoother. This is because this impulse response applies a low-pass filter to the signal. The original signal is made up of sounds at many different frequencies, and the lower frequencies pass through the system, but the higher frequencies are attenuated. This affects how it sounds as well as how it looks.

## Exercise 3: Box Function

1. **Parameters, a time vector that specifies the**

**Text, letter

Description automatically generated**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

1. Use the unitstep function to create a box-shaped time signal. Write a new function called boxt.m that creates a box with specified start and end times t1 and t2. In other words, your function should take three inputs: scalars t1 and t2, and a time vector t, and should output a vector of the same size as t, which contains the values of u(t-t1)-u(t-t2) evaluated at each point in t

**Text

Description automatically generated**

1. Create a script file called boxtscript.m that uses the function to create a box that starts at time t = -1 and ends at time t = 1, where the signal lasts from time t = -3 to t = 3. Generate three different versions of this box using three different time granularities, where the nest granularity has very sharp edges similar to the ideal box and the coarsest granularity has a step size of 0.5. **Text

   Description automatically generated**

**Graphical user interface, table

Description automatically generated**

1. Plot all three versions in one figure using subplot and save it as boxtscript.tif.

**A picture containing line chart

Description automatically generatedGraphical user interface, text, application

Description automatically generated**

1. Time: If u is a vector of length n with time span tu = t1:del:t2, and v is a vector of length m with time span tv = t3:del:t4, and both have the same time step del, then the result of conv(u,v) will be a vector of length n + m - 1 with a time span tc = (t1+t3):del:(t2+t4).
2. Using the box function that you wrote in step 2 with a su‑ciently ne grained step size (for example, del = 0.01), create box signals from (0,4) and (-1,1), with time span of (-5,10). Find and plot the result of the convolution of the two boxes and save it as convplot.tif. Use the above discussion of Time to create the appropriate time vector in your plot. Verify that the timing of signal rising and falling matches what you expect in theory

**A picture containing graphical user interface

Description automatically generatedText

Description automatically generated**

1. Amplitude: In the resulting plot from the previous step, you should notice that the amplitude is much higher than the max of 2 that you would expect from analytically computing the convolution. This is because it is thinking that the length of the box is n rather n del, which impacts the area computation in convolution. To get the correct height, you need to scale by del. Scale and plot the resulting function, and verify that the height is now 2. Save the gure as scaled.tif

A picture containing graphical user interface

Description automatically generated Text

Description automatically generated

1. Triangle: Design the impulse response for a system h and a system input x such that you get a perfectly symmetric triangle of length 100 as the system output y. Use subplot to plot x, h, and y, and save the plot as tri.tif

**Chart

Description automatically generated with medium confidenceText

Description automatically generated**

* The output has the form of triangle pulse when the input signal and impulse response of the system have the form of box-shaped time signal with the same length.