ECE 3300
MATLAB Assignment 3

This assignment explains how to use MATLAB to perform the convolution of two signals in discrete time and in continuous time.

Convolution in MATLAB

Given two vectors `x` and `y`, the command `z=conv(x,y)` assigns to `z` a vector consisting of the convolution of `x` and `y`. For example if, $x[n] = \delta[n]+\delta[n-1]$ and $y[n] = \delta[n]+2\delta[n-1]$, the convolution is $(x*h)[n] = y[n]+y[n-1] = \delta[n]+2\delta[n-1]+\delta[n-1]+2\delta[n-2] = \delta[n]+3\delta[n-1]+2\delta[n-2]$. Using MATLAB, If `x=[1 1]` and `y=[1 2]`, then `z=conv(x,y)` gives output `[1 3 2]`.

Convolution of Discrete-Time Signals

The command `conv` does not keep track of time values, so we need to do this separately. We have learned that if $x[n]$ is "on" from $n = a_1$ to $n = c_1$ and $y[n]$ is "on" from $n = a_2$ to $n = c_2$, then $(x * h)[n]$ is "on" from $n = a_1 + a_2$ to $n = c_1 + c_2$. Given that $x[n]$ is represented in MATLAB by `n_1=a_1:c_1` and `x` and that $y[n]$ is represented by `n_2=a_2:c_2` and `y`, $z[n] = (x * h)[n]$ is obtained via `n=n_1(1)+n_2(1):n_1(end)+n_2(end)` and `z=conv(x,y)`.

PROBLEM STATEMENT: PART ONE

In this problem we seek to verify that the convolution of two rectangularly shaped signals is trapezoidally shaped.

1. Let $x[n] = (u[n] - u[n - 20])$ and $h[n] = (u[n] - u[n - 30])$. Plot $x[n]$ and $h[n]$ on separate graphs using stemplot as per the instructions for discrete-time plots in the first Matlab assignment.

2. Plot $(x * h)[n]$ in a separate graph (as a stemplot).

3. What is the peak height? For how many samples does the convolution have this peak height? Hypothesize and write down what the answers to these two questions are for the more general case $x[n] = (u[n]-u[n-m_1])$ and $h[n] = (u[n]-u[n-m_2])$. (Answers should be functions of m_1 and m_2. Feel free to run different cases to verify your answer.)

Convolution of Continuous-Time Signals

For continuous time signals $x(t)$ and $y(t)$ we can approximate the convolution integral as a sum of rectangles with width b, where b is the time between samples. Let $x[m] = x(mb)$ and $y[m] = y(mb)$, and set $t = nb$. Then

$$
\begin{aligned}
[x * y](t) &= \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau \\
&= \int_{-\infty}^{\infty} x(\tau)y(nb-\tau)d\tau \\
&\approx \sum_{m=-\infty}^{\infty} x(mb)y(nb-mb)b \\
&= \sum_{m=-\infty}^{\infty} x[m]y[n-m]b
\end{aligned}
$$

Apart from the final multiplication by b, the sum is readily implemented with the command `conv`. Thus, if $x(t)$ is implemented in MATLAB via `t_1=a_1:b:c_1` and `x` and $y(t)$ is implemented via `t_2=a_2:b:c_2` and `y`, then $z(t) = [x * y](t)$ can be implemented (approximately) via the following commands: `t=t_1(1)+t_2(1):b:t_1(end)+t_2(end)` and `z=conv(x,y).*b`.

PROBLEM STATEMENT: PART TWO

In this problem we seek to verify that the convolution of $\sqrt{t}e^{-at}u(t)$ with itself is $ct^2e^{-at}u(t)$, where c is a constant. In all parts of this problem use a sampling time of 0.01.

1. Let $x(t) = \sqrt{t}e^{-t}(u(t) - u(t-10))$. (Note that we are truncating the signal because convolution in Matlab requires a finite number of samples.) Let $y(t) = [x * x](t)$. Plot $x(t)$ and $y(t)$ on separate graphs.

2. Let $z(t) = ct^2e^{-t}(u(t) - u(t-20))$, where c is a value of your choosing. (Make it so that $y(t)$ and $z(t)$ have the same number of samples.) Calculate and have the program print out the mean squared error between $y(t)$ and $z(t)$. Also divide the mean squared error by the energy in $y(t)$ and have the program print out this value as a percentage. Run the code over and over, adjusting the value of c so as to make the mean squared error as small as possible. (You can automate this process in Matlab if you wish.) Have the code print out the optimum value of c (to three signficant figures) and the values of the mean squared error and the percentage when dividing MSE by the energy of $y(t)$ in this case.