

ELEN3801 - Fall 2009

Homework 4

Due Thursday October 8th at the **beginning** of class

(Mudd 227 9:10am)¹

Carefully justify ALL your answers

4.1 - Let H be a time-invariant (but not necessarily linear) system. Show that the system's response to a constant input is also a constant, that is, if the input is $f(t) = a$ for all $t \in \mathbb{R}$, then $H\{f(t)\} = b$, where $b \in \mathbb{R}$.

4.2 - Consider an LTI system that has impulse response $h(t) = e^{-t}u(t)$. Find the system response $y(t)$ if the input is

$$(a) u(t) \quad (b) e^{-t}u(t) \quad (c) \sin(3t)u(t)$$

4.3 - (**CORRECTED**) Let $p(t) = u(t+1) - u(t-1)$. Compute and sketch $(p * p)(t)$ and $(p * p * p)(t)$. Comment on your results.

4.4 - Consider a system that has impulse response $h(t) = -\delta(t) + 2e^{-t}u(t)$. Compute the system's response to input $e^t u(-t)$ and sketch both the input and output signals.

4.5 - (*Optional*) Prove the associativity property of the convolution. That is, prove that

$$(f_1 * (f_2 * f_3))(t) = ((f_1 * f_2) * f_3)(t)$$

for arbitrary signals $f_1(t)$, $f_2(t)$ and $f_3(t)$.

4.6 - (*MATLAB Question*)² Let's use MATLAB to help us understand how to manipulate signals, and even approximate the computation of convolutions.

(a) - Create a function that computes the signal $f(t)$ of question 2.2. You should create an m-file named `f.m` with the following code (try to understand what this function is doing):

```
function y=f(t)
%This function computes the value of -t(u(t+1)-u(t))

y=-t.*((t+1>=0)-(t>=0));
```

¹You can always hand-in the homework earlier if you so desire - just give it to me or to one of the TAs, or leave it in my mailbox (in the EE office on the 13th floor of Mudd).

²MATLAB is available in the computers of the Gussman lab (251 Mudd).

- (b) - Construct a vector of time indexes $\mathbf{t}=-4:0.01:4$ and use it to check the answers of question 2.2. That is, plot $f_1(t) = f(-t)$, $f_2(t) = f(t-1) + f(1-t)$, $f_3(t) = f(t-1) + f(-t-1)$, $f_4(t) = f(t-1/2) + f(-(t+1/2))$ and $f_5(t) = 3/2f(t/2-1)$. For example for $f_2(t)$ you just need to type the command `plot(t,f(t-1)+f(1-t))`. Include your plots and your code in the handout, and comment on the results.
- (c) - Let's use MATLAB to compute the convolution integral. Clearly the signal representation we are using in the computer is in discrete time. We can do it by approximating the integral with a sum (and hence using the convolution sum instead). MATLAB can do this very easily. To approximately convolve $f(t)$ with itself we just need to type `z=0.01*conv(f(t),f(t));`. Notice that the size of \mathbf{z} , given by `length(z)` is larger than the size of $f(t)$ (this has to do with the width property of the convolution, and the fact that we are not representing this signals for the entire real line). Also the extra term 0.01 has to do with the discretization step we used when constructing \mathbf{t} . Let's plot the result by using `plot(-8:0.01:8,z)`. Submit this plot.
- (d) - Use the above approach to check your results for question 4.3. Begin by creating the signal $p(t)$ using the function `f.m` already created, by noticing that $p(t) = f((t-1)/2) + f(-(t+1)/2)$. Use the above methodology to compute and plot $(p*p)(t)$ and $(p*p*p)(t)$. Attach the plots and code in your handout.