**EE 313 Linear Signals & Systems (Fall 2018)**

***Solution Set for Homework #8 on Continuous-Time Signals & Systems***

By: Mr. Houshang Salimian & Prof. Brian L. Evans

Here are several useful properties of the Dirac delta functional (generalized function):

1. *Unit area*:
2. *Sifting property:*
3. *Even symmetry:*
4. *Relationship to the unit step function.*

*Here are several comments about bounded-input bounded-output (BIBO) stability:*

1. *BIBO Stability: If input x(t) is bounded in amplitude, i.e. for a finite value B, then output y(t) is always bounded in amplitude, i.e. for a finite value B1. This definition does not require the system to be LTI.*
2. *BIBO stability for LTI systems: For a continuous-time LTI system with an impulse response h(t), BIBO stability reduces to . A derivation is given in problem 3 below.*
3. *BIBO stability for FIR filters: From f), it immediately follows that FIR filters are always BIBO stable (if |h(t)| < ∞ for all t). This is also reflected in the fact that all the poles of an FIR filter are at z=0 (inside the unit circle), which implies stability.*

Please see Handout I on *Bounded-Input Bounded-Output Stability* at <http://users.ece.utexas.edu/~bevans/courses/signals/handouts/Appendix%20H%20BIBO%20Stability.pdf>

*Convolution: Let c(t) = x(t)\*y(t) =>*

**Problem 1:**

1. In this question we can use the following property:





1. The Dirac delta functional is defined in terms of integration: (a) it has unit area at the origin and (b) has a sifting property. The Dirac delta functional is waiting around to be integrated. Please avoid simplifying expressions involving the Dirac delta that are not being integrated.



1. The Dirac delta functional is defined in terms of integration: (a) it has unit area at the origin and (b) has a sifting property. The Dirac delta functional is waiting around to be integrated. Please avoid simplifying expressions involving the Dirac delta that are not being integrated







**Problem 2:** This is averaging filter (unnormalized). Its output is the average of the previous two seconds of input, the current input value, and the future two seconds of input. If a gain of ¼ had been applied, then we’d have a normalized averaging filter (normalized so that the area of the absolute value of the impulse response is one).





**Alternate Solution**:



1. If , then

So, a bounded input generates a bounded output and hence the system is bounded-input bounded-output (BIBO) stable.

A continuous-time LTI system is stable if and only if:

Here, and the system is BIBO stable.

1. This system is not causal, because current output is dependant to future value of input. For instance at t=1: which shows that output at t=1 is related to input values in future, i.e. t = 1 to 3.

Note: A continuous-time, LTI system is causal if and only if, In this question, , which means this system is not causal.





In order to calculate the convolution we should break time domain into three regions.

1st case (No overlap): for 

In this case, do not face any overlap, so



2nd case (partial overlap): for  there is partial overlap between



3rd case (complete overlap):

for 



Therefore:



MATLAB code for plotting output:

fs = 8000;

t = -5: 1/fs :4;

yy = zeros(size(t));

yy (t>=-3 & t<1) = t(t>=-3 & t<1)+3; %second case -3 =< t < 1 and y(t) = t+3

yy(t >= 1) = 4; % third case t >= 1 and y(t) = 4

plot(t,yy)

ylim ([-0.5 4.5])

xlabel ('t(s)')

ylabel ('y(t)')



**Problem 3:**











See graphical flip-and-slide convolution on page 6.

MATLAB code:

clear all

fs = 8000;

t = -2: 1/fs :4;

unitstep = zeros(size(t));

unitstep (t>= 0) = 1; % define unit step function

x = unitstep; % define input x(t) = u(t)

impulse = dirac(t); % define dirac delta function

idx = impulse == Inf;

impulse (idx) = 4;

h = impulse - 3\*exp(-3\*t).\*unitstep; % h(t) is system response

y= exp(-3\*t).\*unitstep; % y(t) = system's output for x(t) = u(t)

figure

plot(t,x)

ylim([-0.5 1.5])

xlabel ('t(s)')

ylabel ('x(t)')

figure

plot(t,h)

xlabel ('t(s)')

ylabel ('h(t)')

figure

plot(t,y)

ylim([-0.5 1.5])

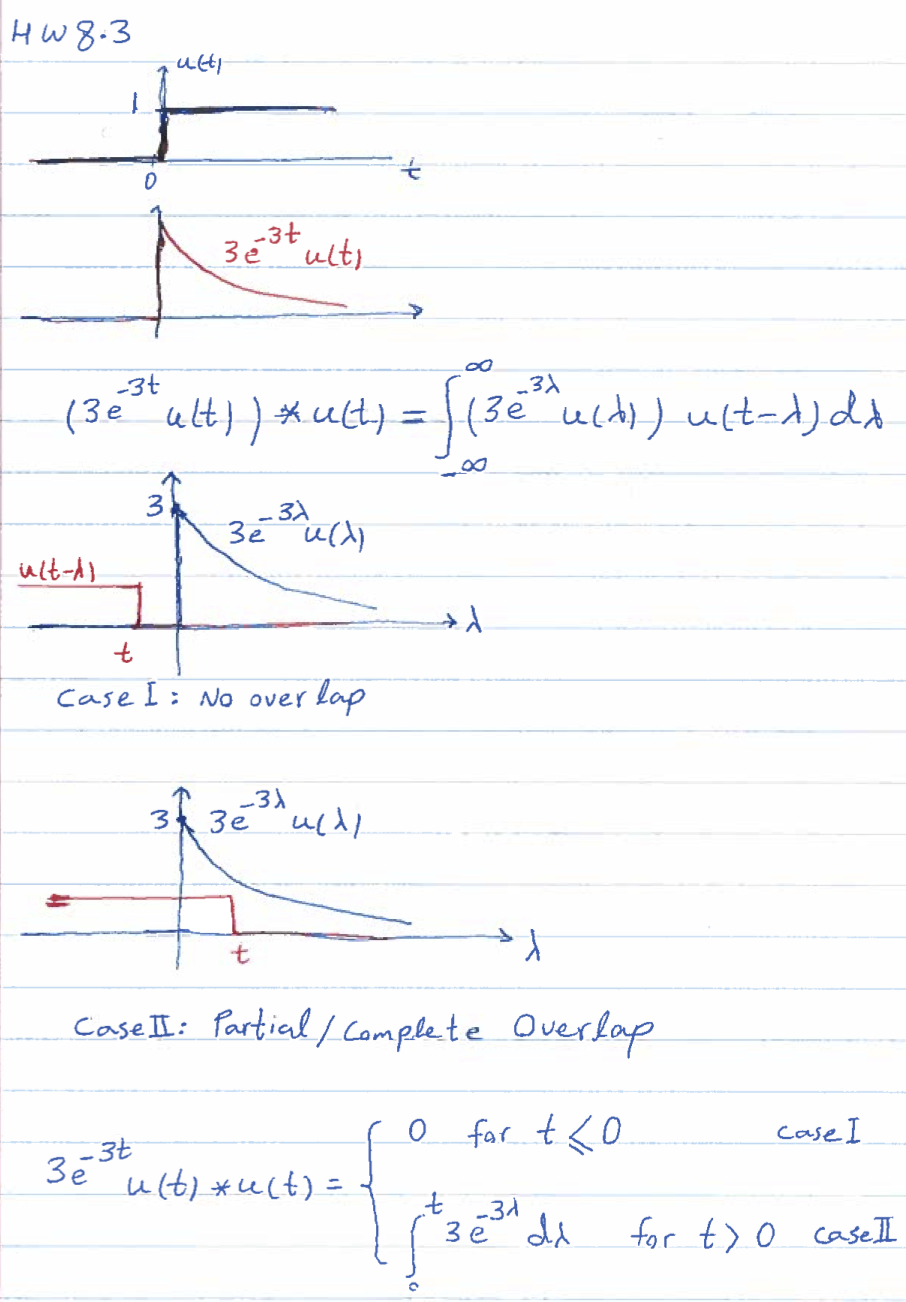
xlabel ('t(s)')

ylabel ('y(t)')







****

**Problem 4:**

The impulse response for the first system will be calculated by placing:





Where the output of first LTI system, w(t), is , and the output of second LTI system is . Here, two systems are connected in cascade:



The impulse response for the cascaded systems is:



MATLAB code:

clear all

fs = 8000;

t = -2: 1/fs :4;

unitstep0 = t>= 0;

unitstep2 = t>= 2;

h = unitstep0 - unitstep2; % h(t) is system response

figure

plot(t,h)

ylim ([-0.5 1.5])

xlabel ('t(s)')

ylabel ('h(t)')

