PROJECT for EE 483

COMMUNICATIONS SYSTEMS I - Fall 2004

Computer Assignment 3: Linear Systems and Filtering

It is known that the response $y(t)$ of a linear filter to an input signal $x(t)$ is given by:

$$y(t) = h(t) * x(t),$$

where $h(t)$ is the impulse response of the filter. In MATLAB you can find the convolution of two signals using the function `conv`. For example, to convolve the signals $x(t) = \cos(5\pi t)$ and $h(t) = \sin(10\pi t)$, where t is in the range -5 to 5 using increments of 0.01, you can do the following:

```
t=-2:0.01:2;
x=cos(5*pi*t);
h=sin(10*pi*t);
y=conv(x,h);
plot(y);
```

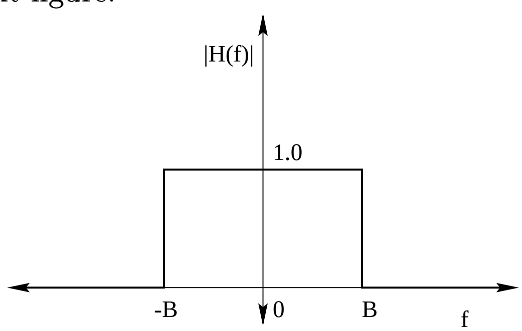
It is recommended to evaluate the output of a linear filter by using the function `linfilt` that is included in the Appendix. In this case the previous example is given by the following commands:

```
t=-2:0.01:2;
x=cos(5*pi*t);
h=sin(10*pi*t);
[y,ty]=linfilt(h,t,x,t);
plot(ty,y);
```

The convolution function conv($\cdot$) of a signal $x(t)$ with another signal $h(t)$ does not require that the signals have the same length (e.g. defined for the same time range). However, note that the convolution product $y(t)$ may be defined for a different time range than the original signals. The function `linfilt` returns both the product of the convolution and the time range of $y(t)$.

Exercise 1 Convolution in time: its what linear systems do
Consider an input signal $x(t)$ that consists of two delta functions at t=1 and t=3 with amplitudes 3 and 2, respectively, to a linear system with impulse response h that is an exponential pulse ($h(t) = e^{-t}$). Plot $x(t)$, $h(t)$ and the output of the linear system $y(t)$ for t in the range of 0 to 10 using increments of 0.01. Use the matlab function `conv`.

Exercise 2
In this exercise we will determine the output of the ideal lowpass filter with bandwidth B and delay $t_0 = 0$, assuming that its input $x(t)$ is a rectangular pulse of unit amplitude and duration T. The frequency response of this filter is shown in the next figure.



We recall that the impulse response of an ideal lowpass filter with bandwidth B is

$$h(t) = 2B\text{sinc}(2Bt).$$

Create an M-file to:

(a) Plot the input to the filter, $x(t)$, where t ranges from -10 to 10 using 0.01 increments. Assume that $T = 5$.

(b) Plot the impulse response $h(t)$, where t ranges from -10 to 10 using 0.01 increments. Assume that $B = 2$.

(c) Plot the output of the filter, $y(t)$. Assume that $T = 5$ and $B = 2$.

(d) Repeat (b), (c) for $T = 5$ and $B = 1$ and 4. Comment on the oscillatory behavior around the discontinuities of the output pulse.

(e) Plot the filter frequency response, and the amplitude spectra of the input to the filter $x(t)$ and the filter output $y(t)$ for $T = 5$ and $B = 1$. In your plots show all spectra for frequencies between -10 to 10 (you can select this frequency range in your program or manually on the matlab plots).

To evaluate the amplitude spectra you may use the function `fouriert` given in Assignment 2.

Appendix

You can evaluate the output of a linear filter (in exercise 2) by using the following function:

```
function [y,ty]=linfilt(h,th,x,tx);

Dtx=tx(2)-tx(1); %We find the time spacing for the input x.
Dth=th(2)-th(1); %We find the time spacing for the impulse response h.
if abs(Dtx-Dth)>1e-5;
   error('ERROR: Time spacings are not equal!\n')
else
   ly=length(x)+length(h)-1;   % We get the length of the response vector.
   %We find the lower time limit of x.
   tx0=tx(1);
   %We find the lower time limit of h.
   th0=th(1);
   %We calculate the lower time limit of y,
   ty0=tx0+th0;
   Dty=Dtx;   %We find the time spacing for y.
   ty=[ty0:Dty:ty0+Dty*(ly-1)];  %We set the time axis for y.
   y=conv(h,x);     %And we calculate y.
end
```

The inputs to this function are the impulse response, `h`, of the linear filter, the time axis, `th`, of the impulse response, the input signal `x`, and the time axis, `tx`, of `x`. The outputs are the vector `y` containing the output of the filter, and the vector `ty` containing the time axis of `y`. So, to plot the output you can type: `plot(ty,y);` . Do not forget to name the M-file that contains the above function `linfilt.m`.

Your report should include all plots and M-files you are asked to create in Exercises 1 and 2.