ECE 146A: Communication Systems
Lab 1: Signals and Systems Computations with Matlab

1 Objective

The goal of this lab is to gain familiarity with computations and plots with Matlab, and to reinforce key concepts in signals and systems. The questions are chosen to illustrate how we can emulate continuous time operations using the discrete time framework provided by Matlab.

Equipment

Matlab is available in the workstations in the software lab.

2 Lab Assignment

Functions and Plots

1 (a) Write a Matlab function *signalx* that evaluates the following signal at an arbitrary set of points:

$$x(t) = \begin{cases} 2e^{t+2}, & -3 \leq t \leq -1 \\ 2e^{-t}\cos 2\pi t, & -1 \leq t \leq 4 \\ 0, & \text{else} \end{cases}$$

That is, given an input vector of time points, the function should give an output vector with the values of x evaluated at those time points. For time points falling outside $[-3, 4]$, the function should return the value zero.
(b) Use the function *signalx* to plot $x(t)$ versus t, for $-6 \leq t \leq 6$. To do this, create a vector of sampling times spaced closely enough to get a smooth plot. Generate a corresponding vector using *signalx*. Then plot one against the other.
(c) Use the function *signalx* to plot $x(t-3)$ versus t.
(d) Use the function *signalx* to plot $x(3-t)$ versus t.
(e) Use the function *signalx* to plot $x(2t)$ versus t.

Convolution

2(a) Write a Matlab function *contconv* that computes an approximation to continuous-time convolution as follows.
Inputs: Vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ representing samples of two signals to be convolved. Scalars t_1, t_2 and dt, representing the starting time for the samples of $\mathbf{x}_1$, the starting time for the samples in $\mathbf{x}_2$,

and the spacing of the samples.
Outputs: Vectors **y** and **t**, corresponding to the samples of the convolution output and the sampling times.
(b) Check that your function works by using it to convolve two boxes, $3I_{[-2,-1]}$ and $4I_{[1,3]}$, to get a trapezoid (e.g., using the following code fragment):

```
dt=0.01;%sample spacing
s1 = -2:dt:-1; %sampling times over the interval [-2,-1]
s2= 1:dt:3; %sampling times over the interval [1,3]
x1=3*ones(length(s1),1); %samples for first box
x2=4*ones(length(s2),1); %samples for second box
[y,t]= contconv(x1,x2,s1(1),s2(1),dt);
figure(1);
plot(t,y);
```

Check that the trapezoid you get spans the correct interval (based on the analytical answer) and has the correct scaling.

Matched filter

3(a) Consider the signal $u(t) = 2I_{[1,3]}(t) - 3I_{[2,4]}(t)$. Plot $u(t)$ and its matched filter $u_{mf}(t) = u(-t)$ on the same plot.
(b) Use the function *contconv* to convolve $u(t)$ and $u_{mf}(t)$. Plot the result of the convolution. Where is the peak of the signal?
(c) Now, consider a complex-valued signal $s(t) = u(t) + jv(t)$, where $v(t) = I_{[-1,2]}(t) + 2I_{[0,1]}(t)$. The matched filter is given by $s_{mf}(t) = s^*(-t)$. Plot the real parts of $s(t)$ and $s_{mf}(t)$ on one plot, and the imaginary parts on another.
(d) Use the function *contconv* to convolve $s(t)$ and $s_{mf}(t)$. Plot the real part, the imaginary part, and the magnitude of the output. Do you see a peak?
(e) Now, use the function *contconv* to convolve $s_1(t) = s(t - t_0)e^{j\theta}$ and $s_{mf}(t)$, for $t_0 = 2$ and $\theta = \frac{\pi}{4}$ Plot the real part, the imaginary part, and the magnitude of the output. Do you see a peak?
(f) If you did not know t_0 and θ, could you estimate it from the output of the convolution in (e)? Try out some ideas and report on the results.

Fourier transform

The following Matlab function is a modification of Code Fragment 2.5.1.

```
function [X,f,df] = contFT(x,tstart,dt,df_desired)
%Use Matlab DFT for approximate computation of continuous time Fourier
%transform
%INPUTS
%x = vector of time domain samples, assumed uniformly spaced
%tstart= time at which first sample is taken
%dt = spacing between samples
%df_desired = desired frequency resolution
```

```
%OUTPUTS
%X=vector of samples of Fourier transform
%f=corresponding vector of frequencies at which samples are obtained
%df=freq resolution attained (redundant--already available from
%difference of consecutive entries of f)
%%%%%%%%
%minimum FFT size determined by desired freq res or length of x
Nmin=max(ceil(1/(df_desired*dt)),length(x));
%choose FFT size to be the next power of 2
Nfft = 2^(nextpow2(Nmin))
%compute Fourier transform, centering around DC
X=dt*fftshift(fft(x,Nfft));
%achieved frequency resolution
df=1/(Nfft*dt)
%range of frequencies covered
f = ((0:Nfft-1)-Nfft/2)*df; %same as f=-1/(2*dt):df:1/(2*dt) - df
%phase shift associated with start time
X=X.*exp(-j*2*pi*f*tstart);
end
```

4(a) Use the function *contFT* to compute the Fourier transform of $s(t) = 3\text{sinc}(2t - 3)$, where the unit of time is a microsecond, the signal is sampled at the rate of 16 MHz, and truncated to the range $[-8, 8]$ microseconds. We wish to attain a frequency resolution of 1 KHz or better. Plot the magnitude of the Fourier transform versus frequency, making sure you specify the units on the frequency axis. Check that the plot conforms to your expectations.
(b) Plot the phase of the Fourier transform obtained in (a) versus frequency (again, make sure the units on the frequency axis are specified). What is the range of frequencies over which the phase plot has meaning?

Matched filter in frequency domain

5(a) Consider the signal $s(t)$ in 3(c). Assuming that the unit of time is a millisecond and the desired frequency resolution is 1 Hz, use the function *contFT* to compute and plot $|S(f)|$.
(b) Use the function *contFT* to compute and plot the magnitude of the Fourier transform of the convolution $s * s_{mf}$ numerically computed in 3(d). Also plot for comparison $|S(f)|^2$, using the output of 5(a). The two plots should match.
(c) Plot the phase of the Fourier transform of $s * s_{mf}$ obtained in 5(b). Comment on whether the plot matches your expectations.

3 Lab Report

- Discuss the results you obtain, answer any specific questions that are asked, and print out the most useful plots to support your answers.
- Append your programs to the report. Make sure you comment them in enough detail so they are easy to understand. In addition to the functions you are asked to write, label the code

fragments used for each assigned segment (1 through 5) separately.

- Write a paragraph about any questions or confusions that you may have experienced with this lab.