Lab 3-Fall 2016-2017

Visualizingand Manipulating Signals

Objective: To proof the convolution, correlation operations and the Fourier series representation of a signal using MATLAB.

Pre-requests: Basics of MATLAB and fundamentals of signals & systems.

Useful References:

- Lecture Notes of the course,
- Signal processing & Linear Systems, (B. P. Lathi, ©2004, ISBN: 978-0-19-568583-1).
- Communication Systems, (Simon S. Haykin, © 2000, ISBN: 978-0-47-117869-9).

Procedure part I: Convolution Sum operation;

1. The Convolution sum of two sequences $x[n]$ and $h[n]$, written as $y[n]=x[n]*h[n]$, is defined by

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

MATLAB has a built-in function, **conv**, to perform convolution on finite-length sequences of values. This function assumes that the two sequences have been defined as vectors and it generates an output sequence that is also a vector. Convolving a sequence $x[n]$ of length N with a sequence $h[n]$ with length M results in a sequence of length **L=N+M-1**. The extent of $x[n]*h[n]$ is equal to the extent of $x[n]$ plus the extent of $h[u]$. Write the following program in your PC and draw the results:

```
clear all; close all; clc;
x=[1 2 2 1 2]; % define sequence x[n]
nx=[-2:2]; % define range of  x[n]
h=[2 2 -1 1 2 2 1]; nh=[-3:3]; % define sequence h[n] and its range
nmin=min(nx)+min(nh); % specify the lower bound of convolved sequences
nmax=max(nx)+max(nh); % specify the upper bound of convolved sequences
y=conv(x,h); % compute convolution
n=[nmin:nmax]; %  spec its range
stem(n,y,'filled');grid % plot the resulting sequence y[n]
title('convolution of two sequence') % add title to the plot
ylabel('y[n]=x[n]*h[n]') % label the y-axis
xlabel('index,[n]') % label the horizontal axis
[n' y'] % print index and sequence y[n] as column vectors
```

Numerical Convolution

The other form of convolution is known as the **convolution integral**. If $x(t)$ and $h(t)$ are two continuous-time signals, then the convolution integral is defined by

$$y(t) = x(t) * h(t) = \int_{0}^{t} x(\tau)h(t-\tau)d\tau$$

Since computers have a hard time integrating, we shall consider evaluation $y(t)$ numerically. Let $t=kT$

$$\therefore y(kT) = \int_{0}^{kT} x(\tau)h(kT-\tau)d\tau \cong T\sum_{i=0}^{k} x(iT)h\big((k-i)T\big) = Tx(kT) * h(kT)$$

where T is the integration step size. This equation is the convolution sum and can be found using the **conv** function.

2. Evaluate the convolution of $x(t) = \sin 2t$ with $h(t) = e^{-0.1t}$ from t=0 to 5 with T=0.1.

```
clear all; close all; clc;
t=0:0.15:5; % define t=KT
x=sin(2*t); % define x(kt)
h=exp(-0.1*t); % define h(kT)
y=0.1*conv(x,h); % evaluate convolution
stem(linspace(0,5,length(y)),y)
```

Procedure part II: Fourier series is often used to model periodic signals.

By truncating the Fourier series, signals can be approximated accurately enough for applications. The computation and study of Fourier series is known as harmonic analysis. The process of expanding a periodic signal in Fourier series is termed **analysis**, while the process of reconstructing a waveform from its Fourier series is termed Fourier **synthesis**. It is well known that around a discontinuity of a function, the partial sum of the Fourier series exhibit oscillatory behavior (ringing) known as the **Gibbs phenomenon**. The **Gibbs** phenomenon is caused by the difficulty to represent sharp discontinuities with smooth trigonometric functions. The Fourier series of a pulse waveform is given by:

$$y(t) = \frac{1}{2\pi} + \sum_{n=1}^{\infty} \frac{2}{n\pi} \sin\left(\frac{1}{2}n\right) \cos(n\pi)$$

Q. Write a script file to synthesize an approximation of the pulse wave. Experiment with the number of terms in the partial sum and describe your observations as the number of terms increase for instance, try as a number of terms to be 10, 50, 100, 500, and 1000. The error caused by using only a finite number of terms (truncated series) is called the truncation error.

use the following MATLAB program to see the results

```
clear all; close all; clc;
t=-3*pi:0.02:3*pi; % specify time span
N=input('Enter the number of terms:') % input the number of terms
x=zeros(size(t)); % initialization
for n=1:N   % time index
    x=x+(2/(n*pi))*sin(0.5*n)*cos(n*t); % form the partial sum
end  % end of loop
y=x+1/(2*pi); % add dc component
plot(t,y);grid % plot the partial sum
s=int2str(N); % convert N into a string
title(['Fourier synthesis of a pulse wave:,'s', terms']) % add title to the
resulting plot
xlabel('time,[s]') % label the horizontal axis
xlabel('Approximation of y(t)') % label the vertical axis
```

Discussion:

1. why in procedure I, the discrete time axis ***n*** was from -5 to +5?
2. perform the convolution operation for another two sequences.
3. list three applications for the Fourier series .

Good luck
Dr. MontadarAbasTaher