# ECE355L Signals and Systems Lab

# Project 4: Fourier Series

# Report due:

**Trigonometric Fourier Series**

In Matlab, we can generate truncated trigonometric Fourier series using the symbolic math toolbox. To calculate the coefficients, $a_0$, $a_n$, and $b_n$, symbolic math commands, *int* and *subs*, can be used.

$$f(t)=\frac{a_0}{2}+\sum_{k=1}^{\infty} ¿¿$$

where

$$a_k=\frac{1}{L}\int_{-L}^{L} f(t)\cos\frac{k\pi t}{L}$$

$$b_k=\frac{1}{L}\int_{-L}^{L} f(t)\sin\frac{k\pi t}{L}$$

The nth partial sum of the Fourier series is

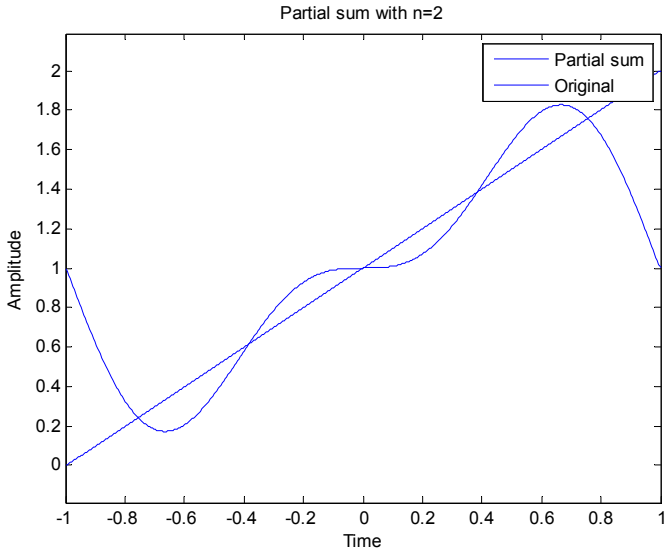
$$x_n(t)=\frac{a_0}{2}+\sum_{k=1}^{n} ¿¿$$

**EX. 1)** To understand Fourier series and Gibb's phenomenon, plot truncated trigonometric, (n = 2) for the periodic wave *f*(*t*) illustrated in the following figure on the interval [-1,1].

```
clc
clear all
close all
syms t k L n                                        % Initialize symbolic variables
evalin(symengine,'assume(k,Type::Integer)');  % Let matlab know that the
variable k is an integer
a = @(f,t,k,L) int(f*cos(k*pi*t)/L,t,-L,L); % create kth cosine coefficient a
b = @(f,t,k,L) int(f*sin(k*pi*t)/L,t,-L,L);   % create kth sine coefficient b
fs = @(f,t,n,L) a(f,t,0,L)/2 + ...
    symsum(a(f,t,k,L)*cos(k*pi*t/L) + b(f,t,k,L)*sin(k*pi*t/L),k,1,n);   %
generate the nth partial sum

f = t;                                                  % Original function
```

```
ezplot(fs(f,t,2,1),-1,1)      %  Plotting the functions and the partial sum
hold on
ezplot(f,-1,1)
hold off
title('Partial sum with n=2'),xlabel('Time'),ylabel('Amplitude')
legend('Partial sum','Original')
```




**Exercise:**

1. Modify the example code to plot the partial sums for n = 10, 20, 50 and 100.
2. Plot the partial sums for n = 2, 10, 20, 50, 100 and 1000 for the functions shown below