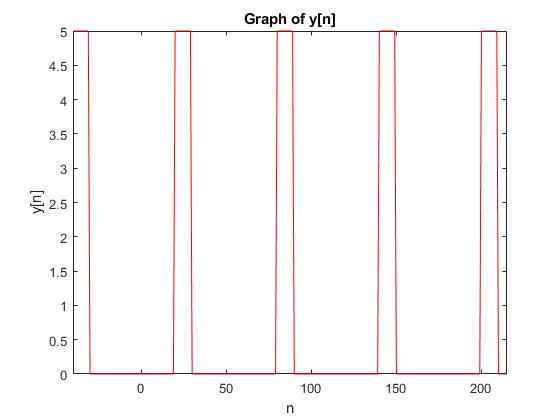
**Lab Work 3**

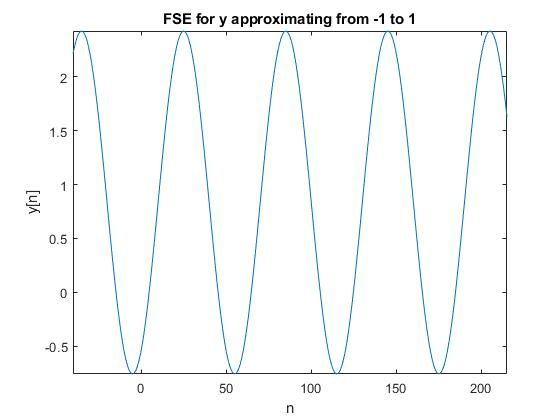
Appendix A - Graphs

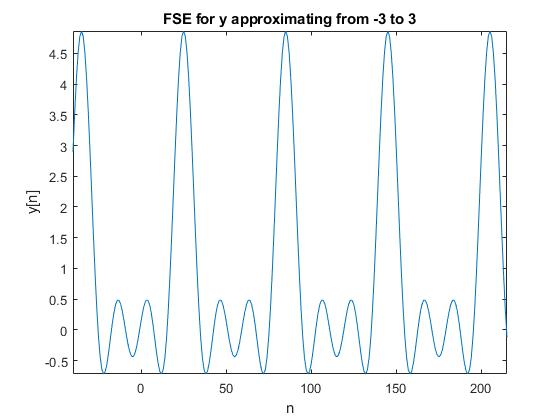
Q1

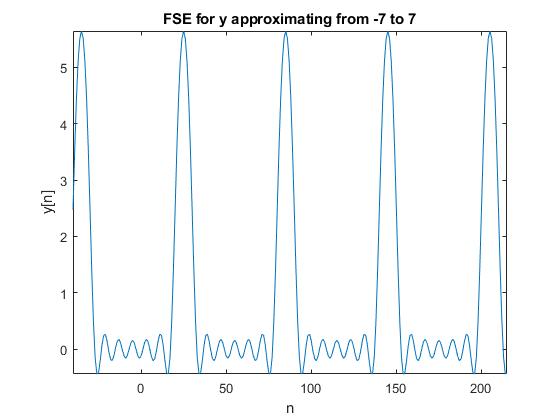
Function

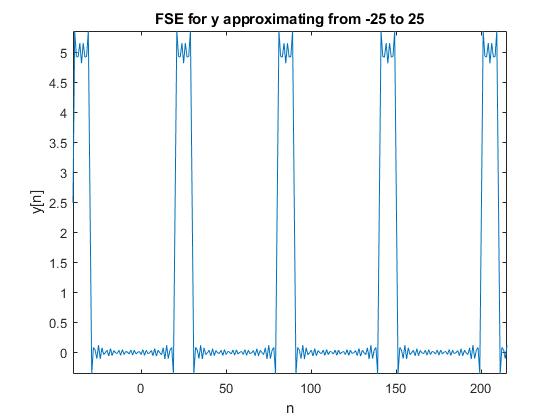


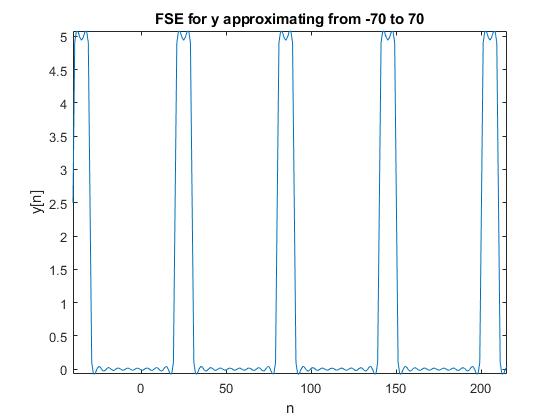
FSE Approximations



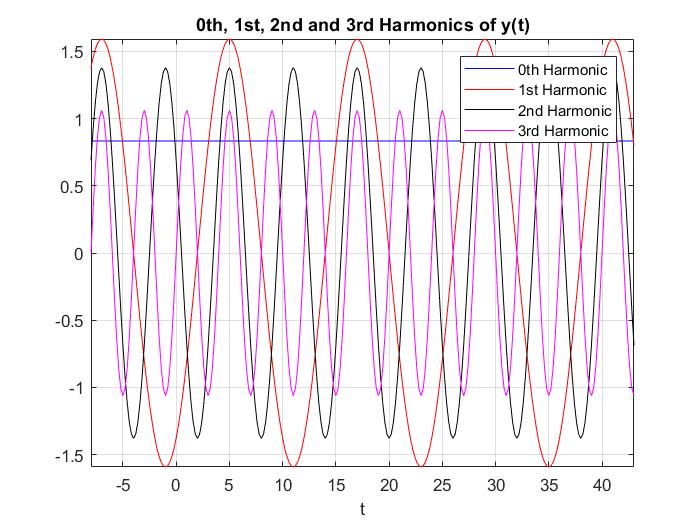






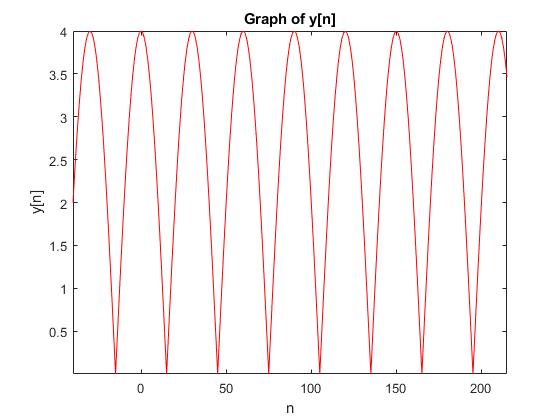


Harmonics

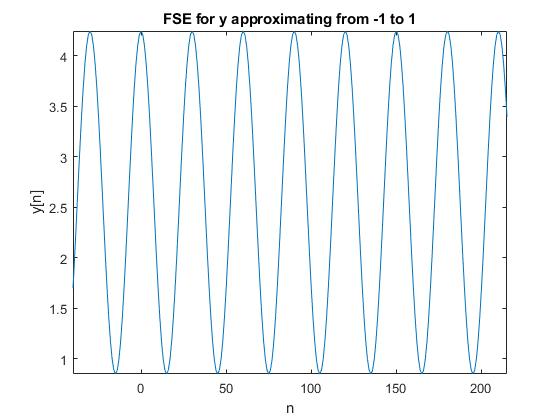


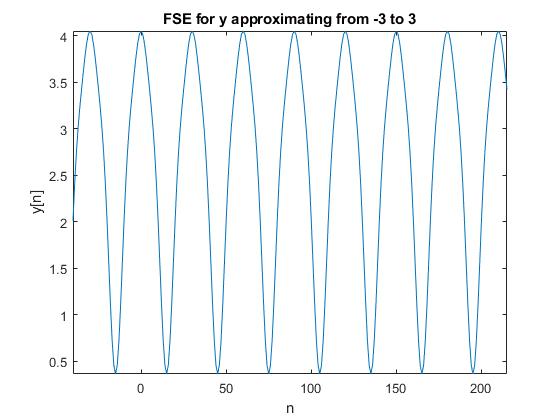
**Question 2**

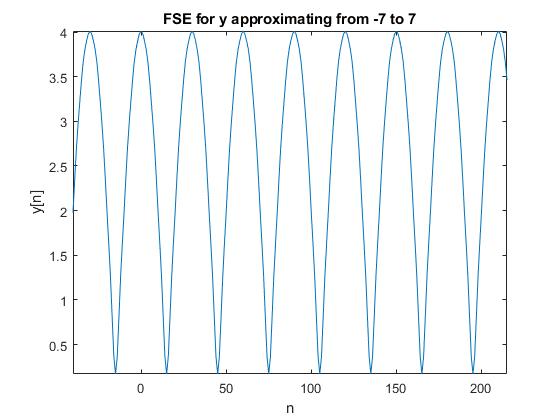
**Function**

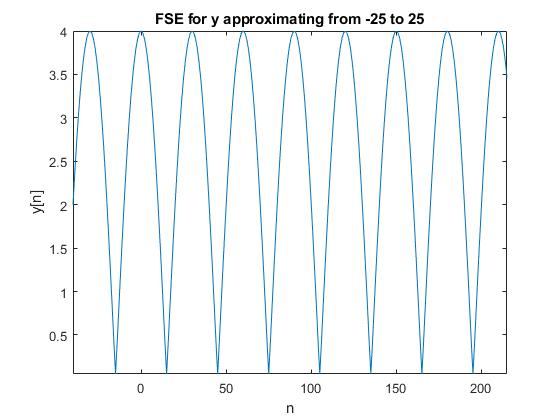
****

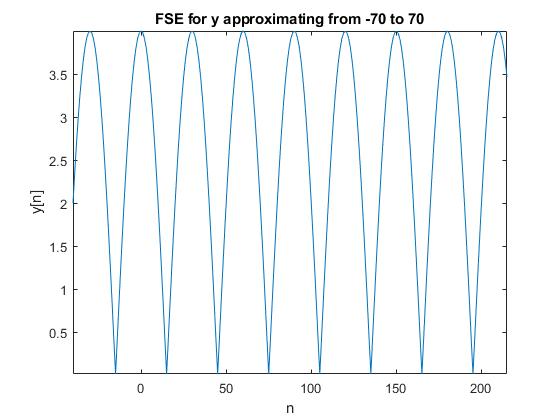
**FSE Approximations**

****

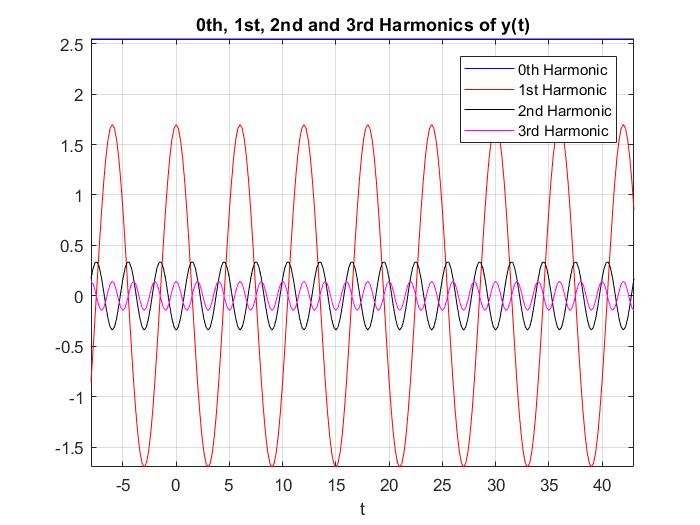
****

****

****

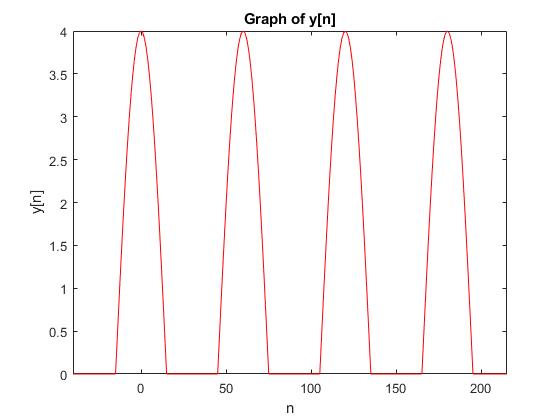
****

**Harmonics**

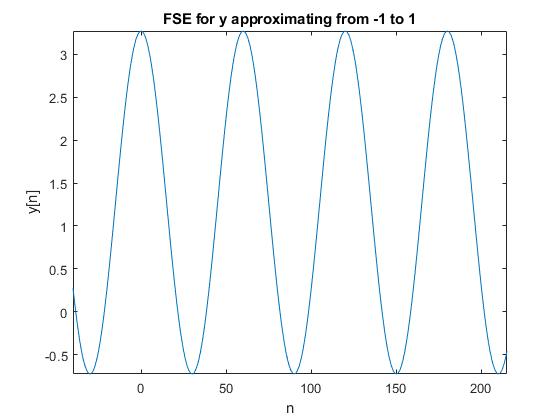
****

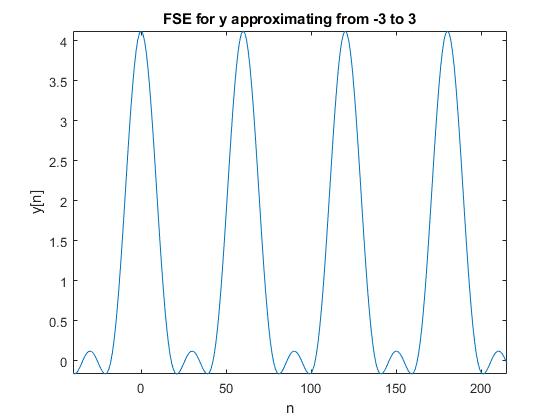
**Question 3**

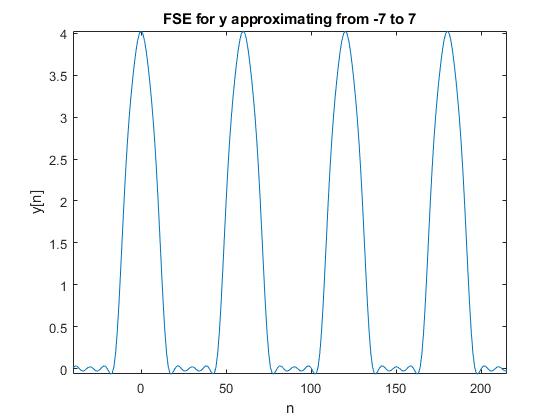
**Function**

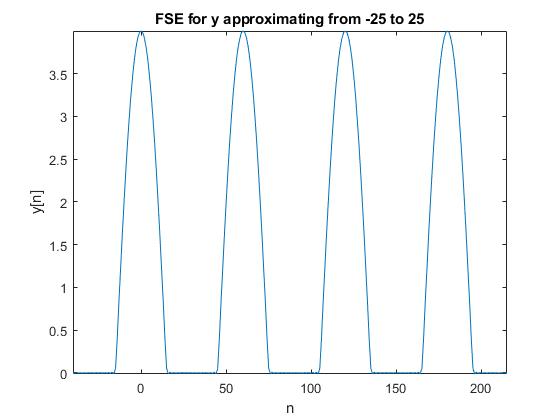
****

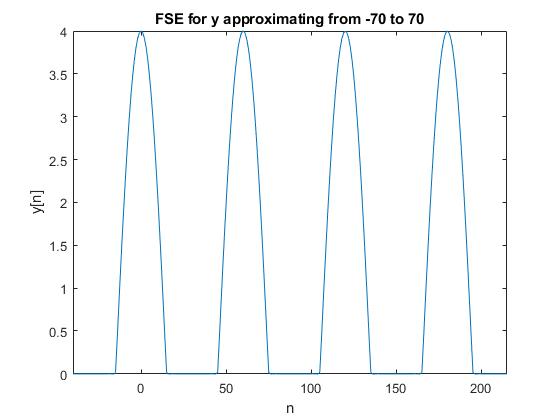
**FSE Approximations**

****

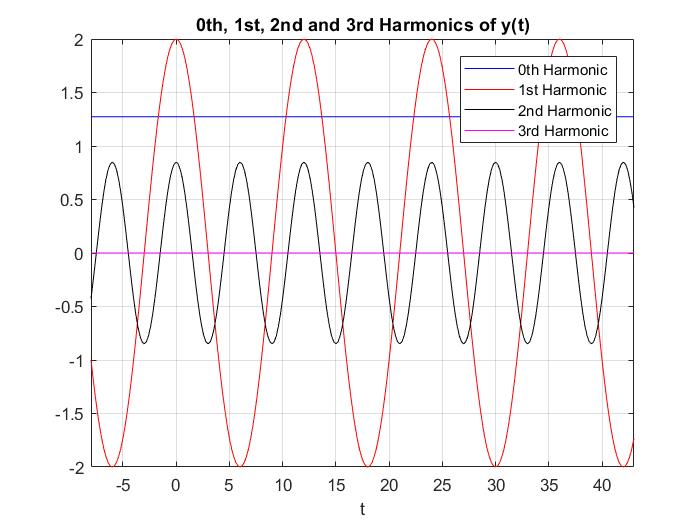
****

****

****

****

**Harmonics**

****

**Appendix B - MATLAB Code**

%% Q1

% Part A

% Defining parameters for discrete function y\_n

prd = 12;

t\_s = 0.2;

n = -40:215;

nt\_s = t\_s .\* n;

% Define anonymous periodic function y\_t

y\_t = @(nt\_s) 0.\*(0<=mod(nt\_s,prd) & mod(nt\_s,prd)<4)...

+ 5.\*(4<=mod(nt\_s,prd) & mod(nt\_s,prd)<6)...

+ 0.\*(6<=mod(nt\_s,prd) & mod(nt\_s,prd)<12);

% Discretizing

y\_n = y\_t(nt\_s);

% Plot a section of y\_n

plot(n,y\_n,'r');

axis tight

title('Graph of y[n]');

ylabel('y[n]');

xlabel('n');

% Part D

% Finding and plotting the ak weights of the expansion

%figure();

k\_lim = 70;

a\_k1\_array = a\_k1(k\_lim);

%stem(-k\_lim:k\_lim, a\_k1\_array);

% Approximating and plotting the FSE

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k1\_array, k\_lim));

title('FSE for y approximating from -70 to 70');

xlabel('n');

ylabel('y[n]');

axis tight

% Part I

% Plot the harmonics

figure();

plot(nt\_s, kth\_harmonic(nt\_s, 0, a\_k1\_array, prd, k\_lim),'b');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 1, a\_k1\_array, prd, k\_lim),'r');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 2, a\_k1\_array, prd, k\_lim),'k');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 3, a\_k1\_array, prd, k\_lim),'m');

title('0th, 1st, 2nd and 3rd Harmonics of y(t)');

xlabel('t');

axis tight

grid on

legend('0th Harmonic','1st Harmonic','2nd Harmonic','3rd Harmonic');

hold off

% Part E,F,G,H

for r = [25, 7, 3, 1]

%figure();

k\_lim = r;

a\_k1\_array = a\_k1(k\_lim);

%stem(-k\_lim:k\_lim, a\_k1\_array);

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k1\_array, k\_lim));

title(['FSE for y approximating from -', num2str(r), ' to ', num2str(r)]);

xlabel('n');

ylabel('y[n]');

axis tight

end

%% Q2

% Part A

% Defining parameters for discrete function y\_n

prd = 6;

t\_s = 0.2;

n = -40:215;

nt\_s = t\_s .\* n;

% Define anonymous periodic function y\_t

y\_t = @(nt\_s) abs(4\*cos(pi\*nt\_s/6));

y\_n = y\_t(nt\_s);

% Plot a section of y\_n

figure();

plot(n,y\_n,'r');

title('Graph of y[n]');

ylabel('y[n]');

xlabel('n');

axis tight

% Part D

% Finding and plotting the ak weights of the expansion

%figure();

k\_lim = 70;

a\_k2\_array = a\_k2(k\_lim);

%stem(-k\_lim:k\_lim, a\_k2\_array);

% Approximating and plotting the FSE

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k2\_array, k\_lim));

title('FSE for y approximating from -70 to 70');

xlabel('n');

ylabel('y[n]');

axis tight

% Part I

% Plot the harmonics

figure();

plot(nt\_s, kth\_harmonic(nt\_s, 0, a\_k2\_array, prd, k\_lim),'b');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 1, a\_k2\_array, prd, k\_lim),'r');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 2, a\_k2\_array, prd, k\_lim),'k');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 3, a\_k2\_array, prd, k\_lim),'m');

title('0th, 1st, 2nd and 3rd Harmonics of y(t)');

xlabel('t');

axis tight

grid on

legend('0th Harmonic','1st Harmonic','2nd Harmonic','3rd Harmonic');

hold off

% Part E,F,G,H

for r = [25, 7, 3, 1]

%figure();

k\_lim = r;

a\_k2\_array = a\_k2(k\_lim);

%stem(-k\_lim:k\_lim, a\_k2\_array);

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k2\_array, k\_lim));

title(['FSE for y approximating from -', num2str(r), ' to ', num2str(r)]);

xlabel('n');

axis tight

ylabel('y[n]');

end

%% Q3

% Part A

% Defining parameters for discrete function y\_n

prd = 12;

t\_s = 0.2;

n = -40:215;

nt\_s = t\_s .\* n;

% Define anonymous periodic function y\_t

y\_t = @(nt\_s) abs(4\*cos(pi\*nt\_s/6)).\*(0<=mod(nt\_s,prd) & mod(nt\_s,prd)<3)...

+ abs(4\*cos(pi\*nt\_s/6)).\*(9<=mod(nt\_s,prd) & mod(nt\_s,prd)<12)...

+ 0.\*(3<=mod(nt\_s,prd) & mod(nt\_s,prd)<9);

% Discretizing

y\_n = y\_t(nt\_s);

% Plot a section of y\_n

figure()

plot(n,y\_n,'r');

ylabel('y[n]');

xlabel('n');

title('Graph of y[n]');

axis tight

% Part D

% Finding and plotting the ak weights of the expansion

%figure();

k\_lim = 70;

a\_k3\_array = a\_k3(k\_lim)

%stem(-k\_lim:k\_lim, a\_k3\_array);

% Approximating and plotting the FSE

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k3\_array, k\_lim));

title('FSE for y approximating from -70 to 70');

xlabel('n');

axis tight

ylabel('y[n]');

% Part I

% Plot the harmonics

figure();

plot(nt\_s, kth\_harmonic(nt\_s, 0, a\_k3\_array, prd, k\_lim),'b');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 1, a\_k3\_array, prd, k\_lim),'r');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 2, a\_k3\_array, prd, k\_lim),'k');

hold on

plot(nt\_s, kth\_harmonic(nt\_s, 3, a\_k3\_array, prd, k\_lim),'m');

title('0th, 1st, 2nd and 3rd Harmonics of y(t)');

xlabel('t');

axis tight

grid on

legend('0th Harmonic','1st Harmonic','2nd Harmonic','3rd Harmonic');

hold off

% Part E,F,G,H

for r = [25, 7, 3, 1]

%figure();

k\_lim = r;

a\_k3\_array = a\_k3(k\_lim);

%stem(-k\_lim:k\_lim, a\_k3\_array);

figure();

plot(n, fourier\_ser(nt\_s,prd,a\_k3\_array, k\_lim));

title(['FSE for y approximating from -', num2str(r), ' to ', num2str(r)]);

xlabel('n');

ylabel('y[n]');

axis tight

end

%% Functions

function out = a\_k1(k\_lim)

out = zeros(0,k\_lim);

k\_rng = -1\*k\_lim:k\_lim;

for k = k\_rng

if(k==0)

temp = 5/6;

else

%temp = (5/6)\*exp(-i\*k\*(pi/6))\*(sin(k\*pi/6)/(k\*pi/6));

temp = ((5\*i)/(2\*pi\*k)) \* (exp(-i\*k\*pi) - exp(-i\*(2/3)\*pi\*k));

end

out(k+k\_lim+1) = temp;

end

end

function out = a\_k2(k\_lim)

out = zeros(0,k\_lim);

k\_rng = -1\*k\_lim:k\_lim;

for k = k\_rng

if(k==1/2)

temp = 8\*i/(3\*pi);

elseif(k==-1/2)

temp = 8\*i/(3\*pi);

else

temp = 8\*cos(pi\*k) / (pi\*(1-4\*k^2));

end

out(k+k\_lim+1) = temp;

end

end

function out = a\_k3(k\_lim)

out = zeros(0,k\_lim);

k\_rng = -1\*k\_lim:k\_lim;

for k = k\_rng

if(k==1)

temp = 1;

elseif(k==-1)

temp = 1;

else

temp = (-4\*cos(pi\*k/ 2)) / (pi\*(k^2-1));

end

out(k+k\_lim+1) = temp;

end

end

function out = fourier\_ser(nt\_s,period, a\_k, rng)

out = 0;

for ind = -rng:rng

out = out + a\_k(ind+rng+1).\* exp(i\*(2\*pi/period).\*ind.\*nt\_s);

end

end

function out = kth\_harmonic(nt\_s, k, a\_k, period, rng)

if k == 0

out = a\_k(rng+k+1).\* exp(i\*(2\*pi/period).\*k.\*nt\_s);

else

out = a\_k(rng+k+1).\* exp(i\*(2\*pi/period).\*k.\*nt\_s)...

+ a\_k(rng-k+1).\* exp(-i\*(2\*pi/period).\*k.\*nt\_s);

end

end