```
%fs1
% fs1.m : FOURIER SERIES - PERIODIC RECTANGULAR PULSE SIGNAL
% Periodic Pulse Signal Convergence (105)
% Longest convergence due to the instantaneous spike characteristics
 of a
%   pulse signal which needs a cos(infinity) for the vertical line
%   resulting in the Gibb's Phenomena => continuously differentiable
%   periodic function has a jump discontinuity resulting in "ears"

T  = 2;                 % T = period
Tp = 1.5;               % Tp = width of pulse
t = -T:0.005:T;         % t = time axis
wo = 2*pi/T;            % fundamental frequency
c0 = Tp/T;              % from the formula for c(k)
x = c0*ones(size(t));   % DC component of x(t)
figure(1)
clf

for k = 1:105
   %figure(1)
   subplot(2,1,1)
   ck = Tp/T*sinc(k*wo*Tp/2/pi);    % the formula for c(k)
   xk = 2*ck*cos(k*wo*t);           % Plotted y value
   plot(t,xk);                      % Aggregate plot of all cos
 functions
   hold on
   x = x + xk;                      % Next Iteration by adding next xk
   subplot(2,1,2)
   hold off
   plot(t,x)                        % Plot of mimicked pulse wave
   xlabel('t')
   title(['N = ',num2str(k)])
   pause(0.2)
end

% fs2
% fs2.m : FOURIER SERIES - |cos(t)| PULSE SIGNAL
% Sinusoidal Wave (30)
% Most rapid convergence since we are getting coefficients for a
 periodic
%   smooth signal which is easily represented using a small number of
%   Fourier Coefficients (or a small sum of cos waves)
%   Almost perfect at 5 iterations and indistinguishable after 10

clear all
clc

T   = 2*pi;                 % T = period
wo  = 2*pi/T;               % fundamental frequency
w0s = 1;

Tpr = 1;
```

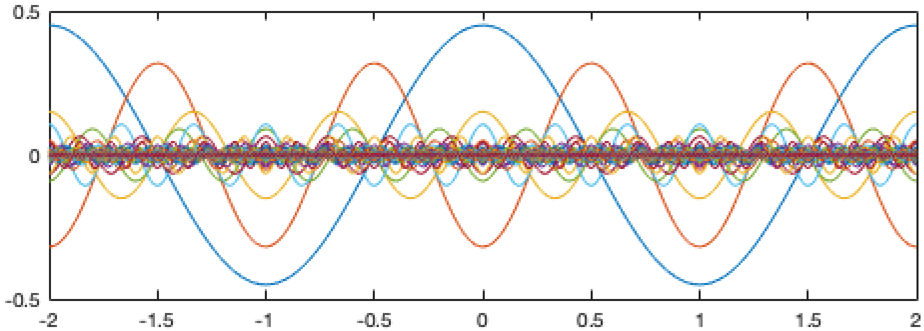
```
t = -Tpr*T:0.005:Tpr*T;     % t = time axis

c0 = 2/pi;
x = c0*ones(size(t));       % DC component of x(t)

Nv = 30;

k = 1:Nv;

% Coefficient formula for pulse wave
ck = 1/2*(sinc(pi/2*(1-k)/pi) + sinc(pi/2*(1+k)/pi));
ck = [0 ck(2:end)];

figure(2)
clf
for k = 1:Nv

    % Loop for xk values
    xk = 2*ck(k)*cos(k*wo*t);

    subplot(2,1,1)
    plot(t,abs(cos(t))); % Original
    ylim([0,1.2])
    % Formatting graph axis
    xticks([-2*pi -1.5*pi -pi -0.5*pi 0 0.5*pi pi 1.5*pi 2*pi])
    xticklabels({'-2 \pi','-3\pi/2','-\pi','-
\pi/2','0','\pi/2','\pi','3\pi/2','2 \pi'})

    % Next iteration, add xk with particular coefficient
    x = x + xk;

    subplot(2,1,2)
    plot(t,x)
    ylim([0,1.2])
    % Formatting graph axis
    xticks([-2*pi -1.5*pi -pi -0.5*pi 0 0.5*pi pi 1.5*pi 2*pi])
    xticklabels({'-2 \pi','-3\pi/2','-\pi','-
\pi/2','0','\pi/2','\pi','3\pi/2','2 \pi'})

    xlabel('t')
    title(['N = ',num2str(k)],...
       'FontSize',20,'Interpreter','latex')
    pause(0.5)
end

%fs3
% fs3.m : FOURIER SERIES - PERIODIC TRIANGULAR PULSE SIGNAL
% Triangular Wave (20)
% Relatively good approximation of a triangle wave with 20 iterations
%   This function is periodic and smooth so we know that a Fourier
%   tarnsform will represent this signal well, but this isn't a
 sinusoidal
%   signal so it will be represented slightly worse
```

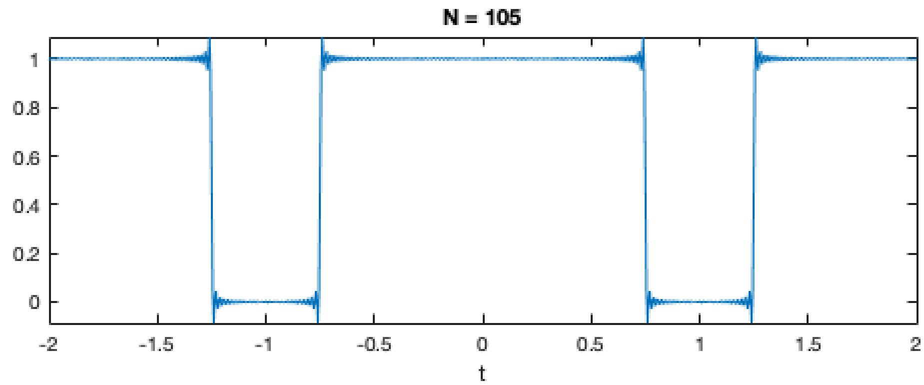
```
T  = 2;                % T = period
t = -2*T:0.005:2*T;    % t = time axis

wo = 2*pi/T;           % fundamental frequency
c0 = 1/2;              % from the formula for c(k)
x = c0*ones(size(t));  % DC component of x(t)

Kv = 20;               % Number of sinusoids

figure(3)
clf
for k = 1:Kv

   ck = 1/T*(sinc(k/T))^2;    % the formula for c(k)
   xk = 2*ck*cos(k*wo*t);     % The kw0 sinusoids

   subplot(2,1,1)
   plot(t,xk);
   title(['$\omega$= ',num2str(k*wo),' Sinusoids'],...
       'FontSize',20,'Interpreter','latex')
   %hold on
   %hold off

   x = x + xk;                % Next iteration

   subplot(2,1,2)

   plot(t,x)
   xlabel('t')
   title(['N = ',num2str(k)],...
       'FontSize',20,'Interpreter','latex')
   pause(0.3)
end

%dfs1
% NUMERICAL COMPUTATION OF FOURIER SERIES COEFFICIENTS
% Sinusoidal wave and Line spectrum of coefficients
% approximate the integral as a summation

% x = abs(cos(pi*t));     % PERIODIC SIGNAL

T = 1;                    % PERIOD
wo = 2*pi/T;

M = 200;                  % DISCRETIZATION OF THE TIME AXIS
delT = T/M;
t = [0:delT:T-delT];      % TIME AXIS
x = abs(cos(pi*t));       % COMPUTE SAMPLES OF x(t) ON GRID

N = 20;                   % COMPUTE FS COEFFICIENTS C(0)...C(N)
J = sqrt(-1);
c = zeros(1,N+1);         % COMPUTE C(k) WITH A SUM
c(1) = 1/T * delT * sum(x);
for k = 1:N
```

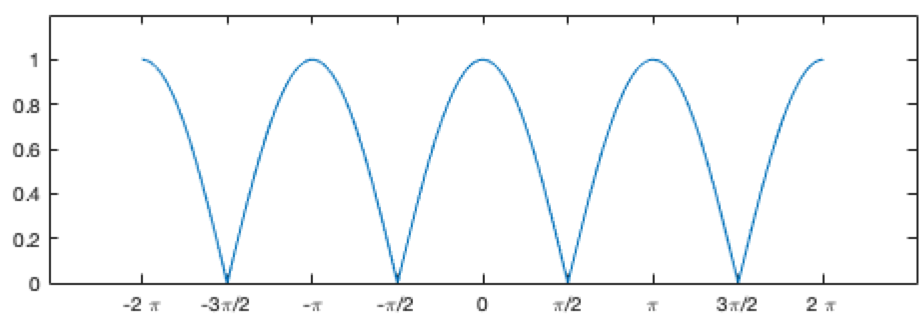
```
    c(k+1) = 1/T * delT * sum(x .* exp(-J*k*wo*[0:M-1]*delT));
    % note: because Matlab indexing begins with 1 instead of 0,
    % it is necessary to add 1 to the index. c(k+1) means 'c(k)'!
end

% NOTE: C(-k) = conj(C(k)) because x(t) is a REAL signal.
% Therefore only compute c(k) for k >= 0.

figure(4)                    % PLOT THE LINE SPECTRUM
stem([0:N]/T,abs(c),'.')  % (THE LINE SPECTRUM IS A PLOT OF C(K))
xlabel('Hertz')
ylabel('|C(k)|')
title('LINE SPECTRUM')

t = [0:500]/500*2*T;        % PLOT TWO PERIODS OF THE SIGNAL
y = c(1) * ones(size(t));   % SYNTHESIZED FROM THE NUMERICALLY
for k = 1:N                 % OBTAINED FS COEFFICIENTS
    y = y + c(k+1)*exp(J*k*wo*t) + conj(c(k+1))*exp(-J*k*wo*t);
end
figure(5)
plot(t,real(y));

%dfs2
% NUMERICAL COMPUTATION OF FOURIER SERIES COEFFICIENTS
% Triangular wave and Line spectrum of coefficients
% approximate the integral as a summation

T = 2;                      % PERIOD
wo = 2*pi/T;

M = 200;                    % DISCRETIZATION OF THE TIME AXIS
delT = T/M;
t = [0:delT:T-delT];        % TIME AXIS


% TWO VERSIONS OF TRIANGLE SIGNALS

% Triangle Wave using Sawtooth function with 50% symmetry
% x = abs( sawtooth ((pi * (t) / 2), 0.5) );

% Triangular Wave Samples using mod (remainder of t / 2)
% Since t is increasing from 0 to T*1.99, remainder follows triangle
 wave
%   pattern since wrap around after t passes multiples of 2
x = abs(mod(t,2)-1);


N = 100;                     % COMPUTE FS COEFFICIENTS C(0)...C(N)
J = sqrt(-1);
c = zeros(1,N+1);           % COMPUTE C(k) WITH A SUM in the for loop
c(1) = 1/T * delT * sum(x);

for k = 1:N
    c(k+1) = 1/T * delT * sum(x .* exp(-J*k*wo*[0:M-1]*delT));
```

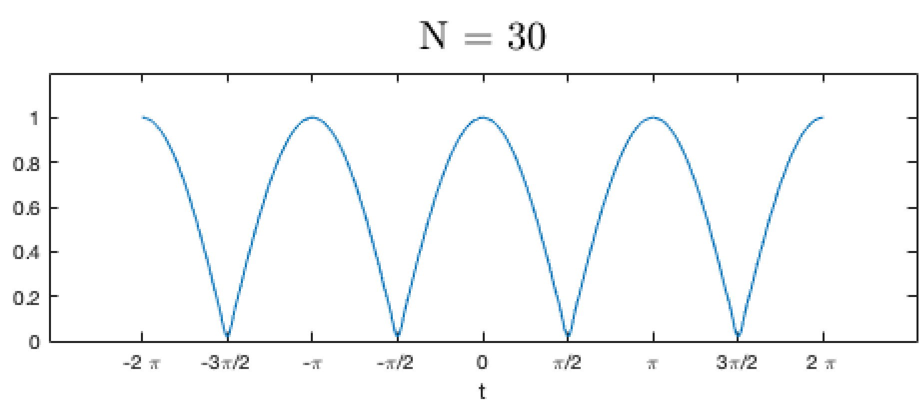
```
    % note: because Matlab indexing begins with 1 instead of 0,
    % it is necessary to add 1 to the index. c(k+1) means 'c(k)'!
end

% NOTE: C(-k) = conj(C(k)) because x(t) is a REAL signal.
% Therefore only compute c(k) for k >= 0.

figure(6)                  % PLOT THE LINE SPECTRUM
stem([0:N]/T,abs(c),'.')  % (THE LINE SPECTRUM IS A PLOT OF C(K))
xlabel('Hertz')
ylabel('|C(k)|')
title('LINE SPECTRUM')

t = [0:500]/500*2*T;       % PLOT TWO PERIODS OF THE SIGNAL
y = c(1) * ones(size(t));  % SYNTHESIZED NUMERICALLY

for k = 1:N                % OBTAINED FS COEFFICIENTS
    y = y + c(k+1)*exp(J*k*wo*t) + conj(c(k+1))*exp(-J*k*wo*t);
end
figure(7)
plot(t,real(y));

%dfs3
% TIME SHIFTING PROPERTY OF FOURIER COEFFICIENTS
% Sinusoidal wave and Line spectrum of coefficients
% As proven in class, shifting a function results in coefficients of
 all
%   same magnitude but shifted phases. know f(t) => Cn
%   f(t-t0) = exp( -(j)(n)(w0)(t0) ) * Cn so same plot as before but
 shift
%   so time shift property is confirmed

T = 2;                          % PERIOD
wo = pi;

M = 200;                        % DISCRETIZATION OF THE TIME AXIS
delT = T/M;
t = [0:delT:T-delT];            % TIME AXIS

% Modified x(t) to x(t-t0) so signal delayed by t0 seconds as seen
x = abs(cos(pi*(t-0.2)));       % COMPUTE SAMPLES OF x(t-0.2) ON GRID

N = 20;                         % COMPUTE FS COEFFICIENTS C(0)...C(N)
J = sqrt(-1);
c = zeros(1,N+1);               % COMPUTE C(k) WITH A SUM
c(1) = 1/T * delT * sum(x);

for k = 1:N
    c(k+1) = 1/T * delT * sum(x .* exp(-J*k*wo*[0:M-1]*delT));
    % note: because Matlab indexing begins with 1 instead of 0,
    % it is necessary to add 1 to the index. c(k+1) means 'c(k)'!
end

% NOTE: C(-k) = conj(C(k)) because x(t) is a REAL signal.
```

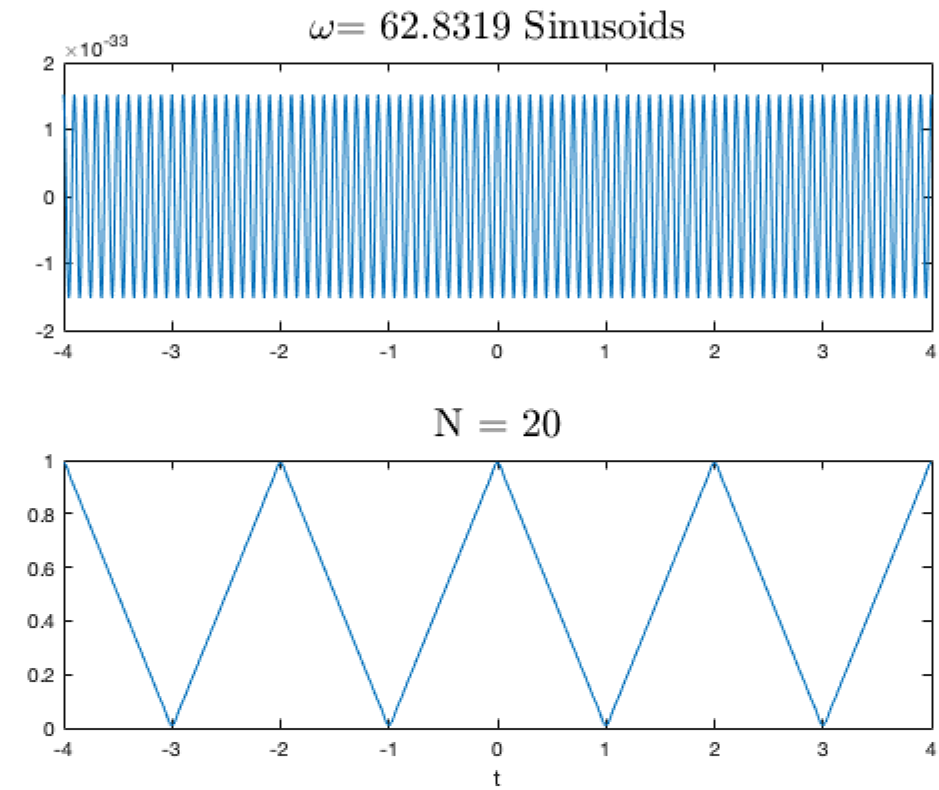
```
% Therefore only compute c(k) for k >= 0.

figure(8)                   % PLOT THE LINE SPECTRUM
stem([0:N]/T,abs(c),'.')  % (THE LINE SPECTRUM IS A PLOT OF C(K))
xlabel('Hertz')
ylabel('|C(k)|')
title('LINE SPECTRUM')

t = [0:500]/500*T;          % PLOT TWO PERIODS OF THE SIGNAL
y = c(1) * ones(size(t));  % SYNTHESIZED FROM THE NUMERICALLY
for k = 1:N                 % OBTAINED FS COEFFICIENTS
    y = y + c(k+1)*exp(J*k*wo*t) + conj(c(k+1))*exp(-J*k*wo*t);
end
figure(9)
plot(t,real(y));

figure(10);
clf;
subplot(2,2,[1 2]);
%Time Shifted graph
plot(t, real(y));
title("Time Shifted");
subplot(2,2,[3 4]);

% Old Unshifted Graph
T = 1;
wo = 2*pi/T;
M = 200;
delT = T/M;
t = [0:delT:T-delT];
x = abs(cos(pi*t));
N = 20;
J = sqrt(-1);
c = zeros(1,N+1);
c(1) = 1/T * delT * sum(x);
for k = 1:N
    c(k+1) = 1/T * delT * sum(x .* exp(-J*k*wo*[0:M-1]*delT));
end
t = [0:500]/500*2*T;
y = c(1) * ones(size(t));
for k = 1:N
    y = y + c(k+1)*exp(J*k*wo*t) + conj(c(k+1))*exp(-J*k*wo*t);
end

plot(t,real(y));
title("Unishfted");
```


0.5
0
-0.5
-2 -1.5 -1 -0.5 0 0.5 1 1.5 2

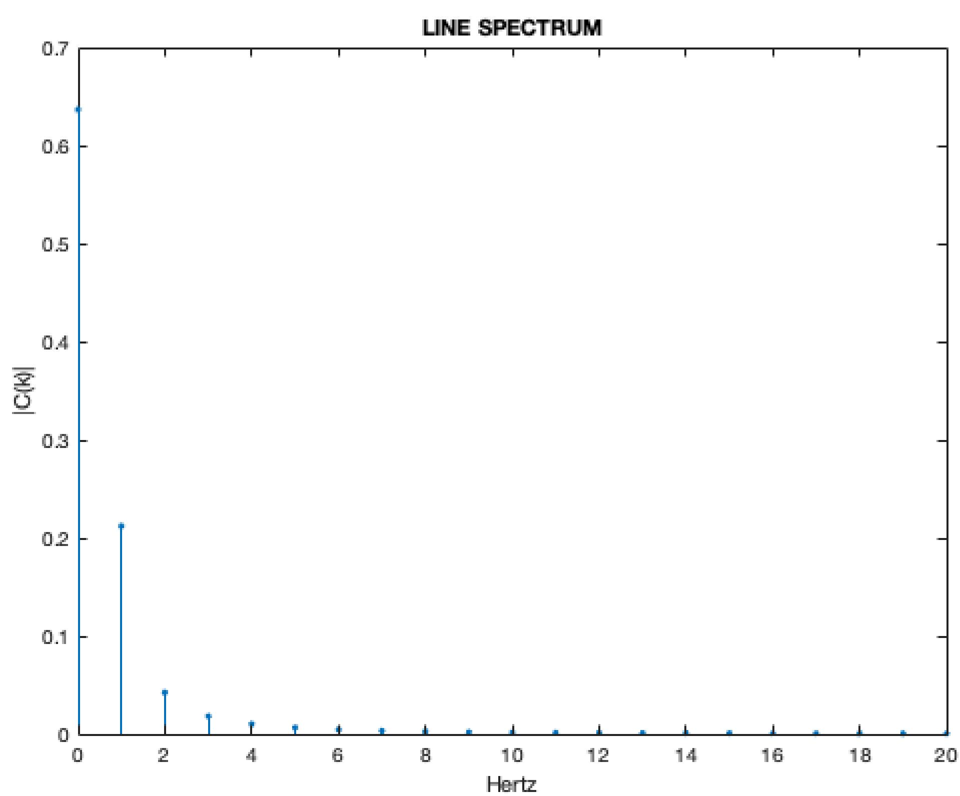

N = 105
1
0.8
0.6
0.4
0.2
0
-2 -1.5 -1 -0.5 0 0.5 1 1.5 2
t


1
0.8
0.6
0.4
0.2
0
-2 π -3π/2 -π -π/2 0 π/2 π 3π/2 2 π

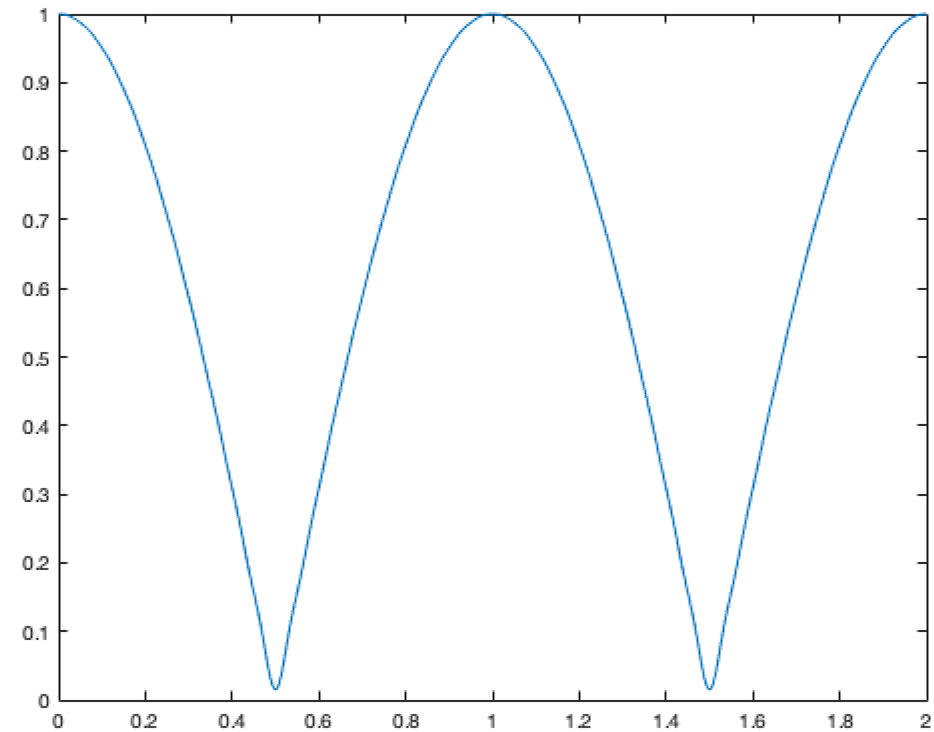

N = 30
1
0.8
0.6
0.4
0.2
0
-2 π -3π/2 -π -π/2 0 π/2 π 3π/2 2 π
t

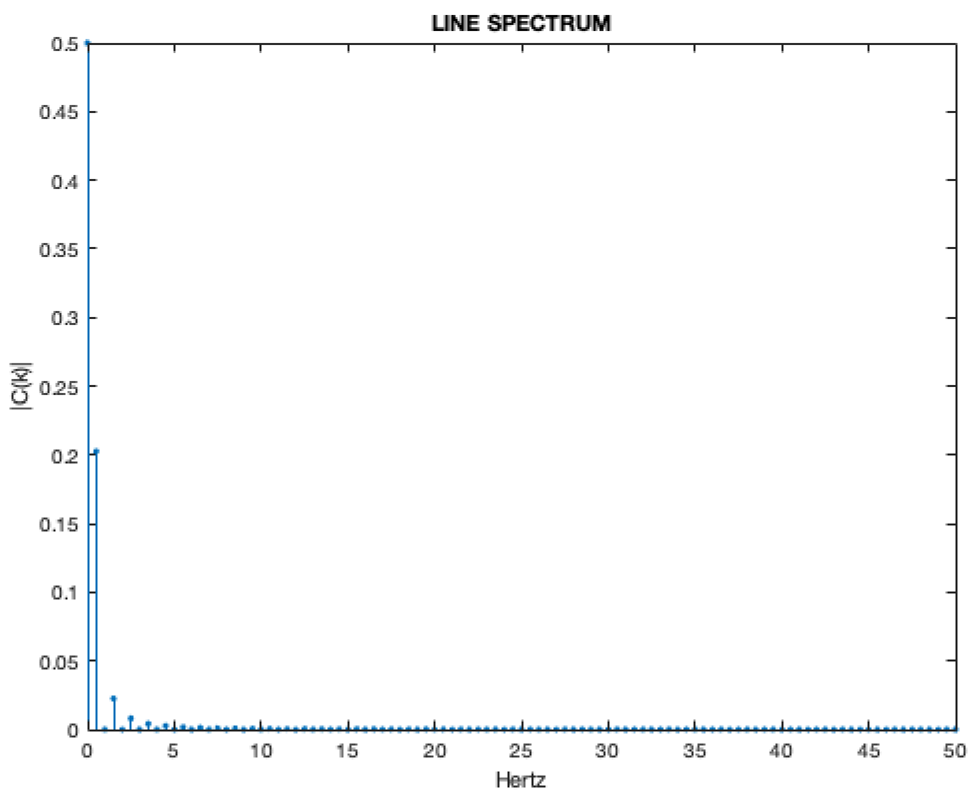


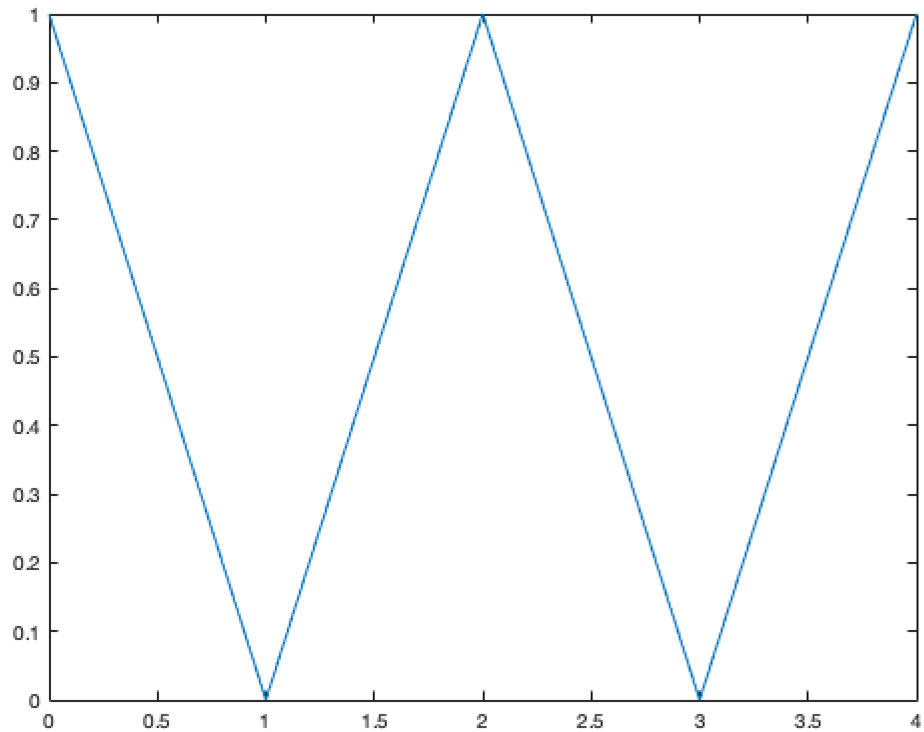
ω= 62.8319 Sinusoids
×10^-33
2
1
0
-1
-2
-4
-3
-2
-1
0
1
2
3
4
N = 20
1
0.8
0.6
0.4
0.2
0
t

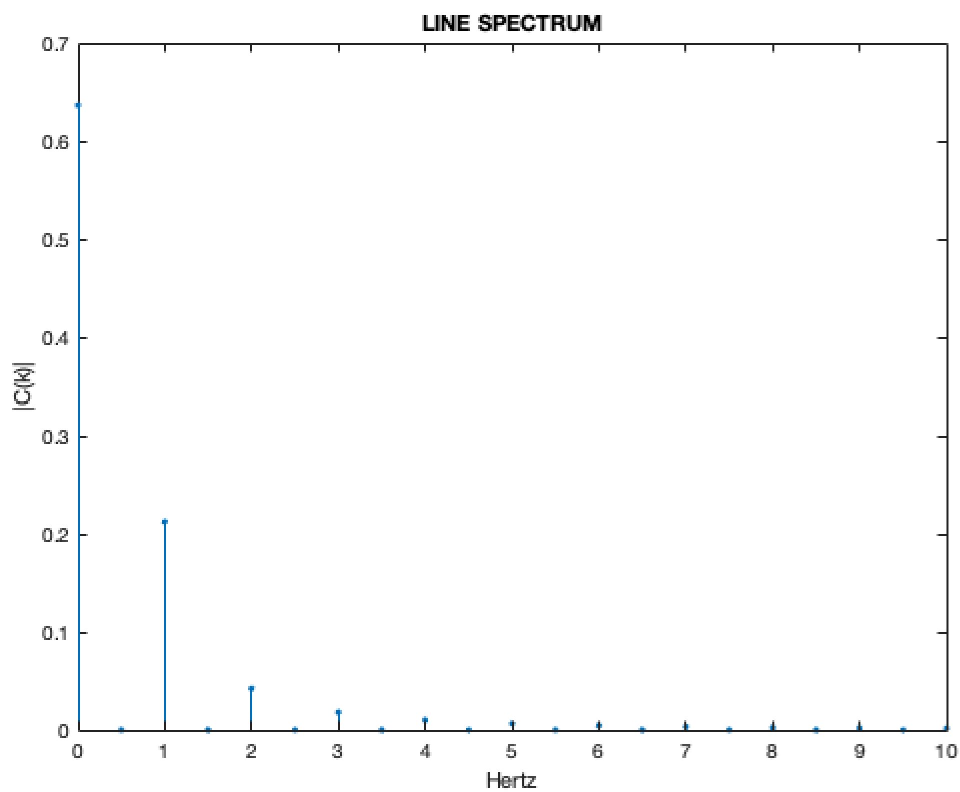


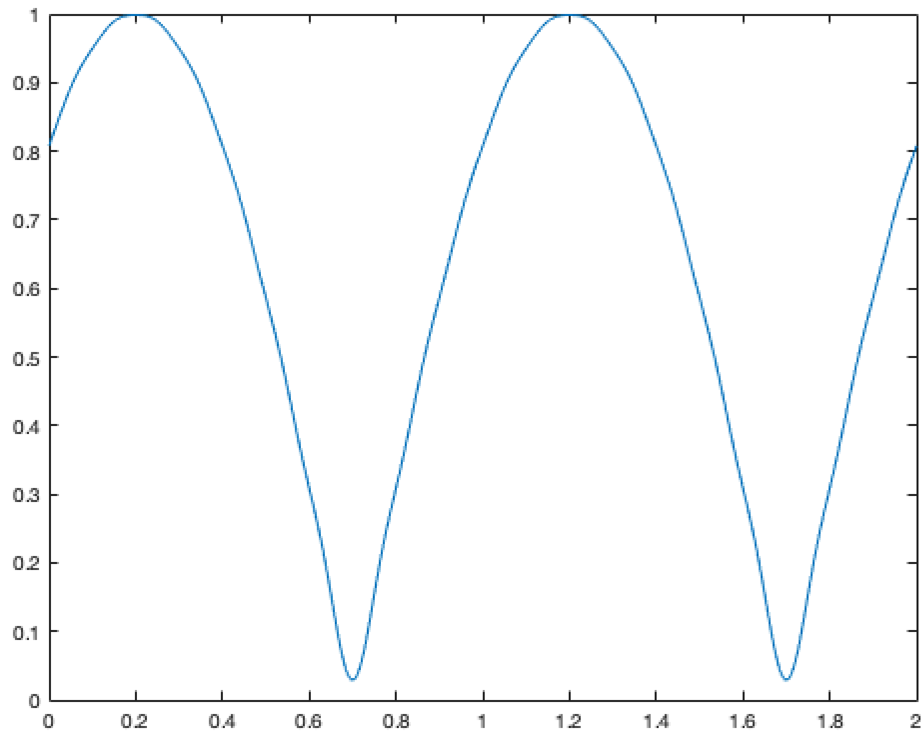
LINE SPECTRUM
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
|C(k)|
0
2
4
6
8
10
12
14
16
18
20
Hertz


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

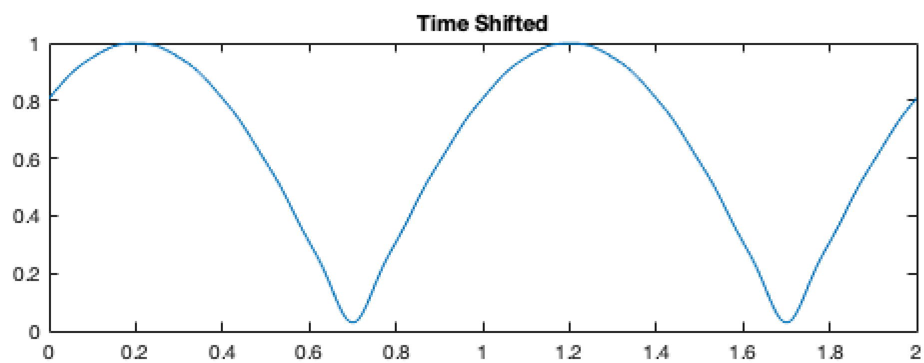

LINE SPECTRUM
|C(k)|
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
5
10
15
20
25
30
35
40
45
50
Hertz


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.5
1
1.5
2
2.5
3
3.5
4

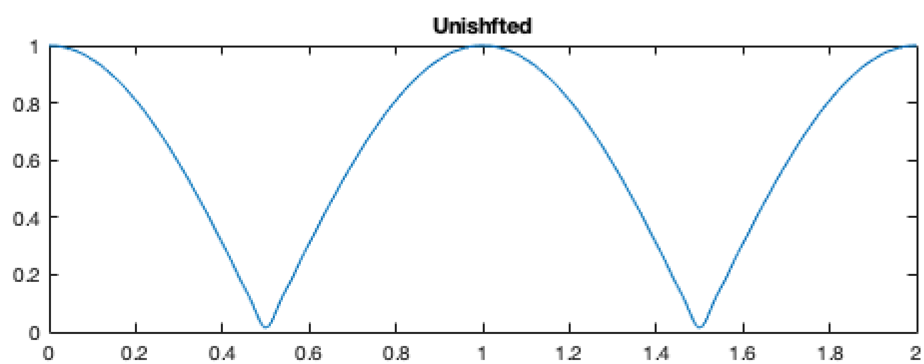

LINE SPECTRUM
|C(k)|
Hertz
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
1
2
3
4
5
6
7
8
9
10



1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2



Time Shifted
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2



Unishfted
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

Published with MATLAB® R2018a